SLUE Phase-2: A Benchmark Suite of Diverse Spoken Language Understanding Tasks

Suwon Shon[1], **Siddhant Arora**[2*], **Chyi-Jiunn Lin**[3*], **Ankita Pasad**[4*], **Felix Wu**[1], **Roshan Sharma**[2], **Wei-Lun Wu**[3], **Hung-Yi Lee**[3], **Karen Livescu**[4], **Shinji Watanabe**[2]

[1]ASAPP [2]Carnegie Mellon University [3]National Taiwan University
[4]Toyota Technological Institute at Chicago

Abstract

Spoken language understanding (SLU) tasks have been studied for many decades in the speech research community, but have not received as much attention as lower-level tasks like speech and speaker recognition. In this work, we introduce several new annotated SLU benchmark tasks based on freely available speech data, which complement existing benchmarks and address gaps in the SLU evaluation landscape. We contribute four tasks: *question answering* and *summarization* involve inference over longer speech sequences; *named entity localization* addresses the speech-specific task of locating the targeted content in the signal; *dialog act classification* identifies the function of a given speech utterance. In order to facilitate the development of SLU models that leverage the success of pre-trained speech representations, we will release a new benchmark suite, including for each task (i) curated annotations for a relatively small fine-tuning set, (ii) reproducible pipeline (speech recognizer + text model) and end-to-end baseline models and evaluation metrics, (iii) baseline model performance in various types of systems for easy comparisons. We present the details of data collection and annotation and the performance of the baseline models. We also analyze the sensitivity of pipeline models' performance to the speech recognition accuracy, using more than 20 publicly available speech recognition models.

1 Introduction

Spoken language understanding (SLU) tasks involve inferring the linguistic structure or semantic meaning of a speech signal beyond its text transcript. We use this term broadly to include any natural language processing (NLP) task applied to speech, and tasks that involve linguistic understanding but also localization in the signal of relevant segments or producing speech as output. SLU has been an active area throughout the history of speech research (Hemphill et al., 1990; Calhoun et al., 2010; Busso et al., 2008; Zadeh et al., 2018; Chen et al., 2020a; Cohn et al., 2019; Yadav et al., 2020; Martinez-Lucas et al., 2020). However, compared to "lower-level" tasks like automatic speech recognition (ASR) and speaker identification, SLU has received much less attention and resources, and specifically there are much fewer benchmarks with freely available data.

SLU tasks can in principle be addressed via a pipeline approach — using ASR to map speech to text and an NLP (text) model to map text to the desired output. The alternative is an end-to-end (E2E) model, which maps directly from the input speech to the target output. While pipeline approaches can take advantage of existing strong ASR and NLP models, E2E models can be more efficient at inference time, can avoid ASR error propagation, and can directly use aspects of the speech signal beyond the text that are useful for the task (e.g., prosody) (Arora et al., 2022a; Chen et al., 2020b; Jurafsky et al., 1998; Tran et al., 2018). In addition, for tasks whose output includes speech segments or time spans, there is no direct combination of an ASR model and an NLP model that produces precisely the desired type of output. For some SLU tasks, the current state of the art is a pipeline model (Shon et al., 2022a; Lai et al., 2020), whereas for others E2E models are better (Pasad et al., 2021; Sharma et al., 2022; Wu et al., 2022b; Peng et al., 2022; Arora et al., 2022b; Shon et al., 2022b). In order to better understand the pros and cons of pipeline and E2E approaches, more public benchmarks are sorely needed.

While collecting large amounts of labeled speech data for many SLU tasks may be prohibitively costly, recent advances in pre-trained models (Baevski et al., 2020; Hsu et al., 2021; Chen et al., 2021; Wu et al., 2022a; Baevski et al., 2022; Lin et al., 2022b; Mohamed et al., 2022) make

[*] Core contributors in alphabetical order

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8906–8937

it feasible to use relatively small fine-tuning sets for each task. There have been several recent efforts to introduce new benchmark SLU tasks (Yang et al., 2021; Bastianelli et al., 2020; Feng et al., 2021; Evain et al., 2021; Arora et al., 2022b; Lugosch et al., 2021a; Shon et al., 2022a; Tomasello et al., 2022), most (but not all) using fairly small training sets of several hours to several dozens of hours of speech. Among them, the Spoken Language Understanding Evaluation (SLUE)[1] (Shon et al., 2022a) motivated us since it pursues a natural speech, rather than a short command type of speech that is populated in other benchmarks. However, there are only two SLUE tasks (sentiment analysis and named entity recognition), thus more tasks with different complexities are needed to cover the diverse application of SLU.

We introduce SLUE Phase-2, a set of SLU tasks that complement the existing SLU datasets or benchmarks. The new tasks include dialog act classification (DAC), question answering (QA), summarization (SUMM), and named entity localization (NEL) , applied to English speech data. SLUE Phase-2 has several advantages compared to other recent work introduced in section 2:

More diverse tasks: SLUE phase-2 not only include utterance or word-level classification task but also includes QA and SUMM task.

More challenging tasks: The complexity of the task is influenced by the type of input and the type of output. SLUE phase-2 uses conversational or longer discourse speech as input. The type of output is not limited to labels or text, but also includes the speech span time stamp.

New human annotation: A new annotation was collected by a human annotator. Human annotator validated an automatically-collected data if needed.

Natural speech: We do not use synthesized speech. We only include conversational or considerably long discourse speech rather than short speech commands.

CC license: Creative Common licensed dataset to give the best freedom of use.

For each task, we provide publicly available[2] datasets, annotations, models, and code. We provide both pipeline and E2E baseline models and, for pipeline models, we use multiple ASR systems to analyze the effect of the ASR error rate on the final task performance.

[1]We refer to the original SLUE as "SLUE Phase-1."

[2]To be released.

2 Related work

SUPERB (Yang et al., 2021) aggregates several existing speech tasks mainly to evaluate frozen pre-trained speech models. It focuses on low-level tasks but also contains two SLU tasks — intent classification (from Fluent Speech Commands (Lugosch et al., 2019)) and slot filling (from SNIPS (Coucke et al., 2018)). However, the former is an easy task where many models have close to 100% accuracy, and the latter uses synthesized rather than natural speech. **SLURP** (Bastianelli et al., 2020) is a spoken version of a text dataset (Liu et al., 2019) where the authors hired workers to dictate the written conversations between humans and personal robot assistants. It includes three SLU tasks — scenario prediction, action prediction, and entity prediction. These tasks cannot be generalized as the nature of the short speech command. **ASR-GLUE** (Feng et al., 2021) is based on the well-known GLUE benchmark (Wang et al., 2018) where the authors hired people to speak the GLUE text . It includes five GLUE tasks and one additional task. However, ASR-GLUE contains only a test set; researchers must rely on other datasets for training. **Timers and Such** (Lugosch et al., 2021b) is a dataset of speech commands that involve numbers, designed for intent classification and slot filling that has limited use case. **Spoken SQuAD** (Lee et al., 2018) and **Spoken CoQA** (You et al., 2022) are synthesized speech versions of the text SQuAD (Rajpurkar et al., 2016) and CoQA (Reddy et al., 2019) datasets. **NMSQA** (Lin et al., 2022a) is a multi-speaker spoken QA dataset whose test set contains natural speech but the train and validation sets are synthesized. Other well-known SLU datasets include **ATIS** (Hemphill et al., 1990) and **Switchboard NXT** (Calhoun et al., 2010), which have been used for tasks like intent and DAC, but the data is available under license constraints. Wu et al. (2020) published an open-sourced speech dataset; however, its dialog act annotation is not manually annotated but predicted using commercial API.

Speech summarization has gained interest over the past few years with tasks such as abstractive summarization of instructional **How-2** videos (Sanabria et al., 2018) and **TED Talks** (Kano et al., 2021), but the raw audio for these tasks is not publicly available. Other corpora, such as the **ICSI** (Janin et al., 2003) and **AMI** (McCowan et al., 2005) meeting summarization corpora, contain relatively less annotated data. Named entity localization (NEL) is a fairly new task. A similar task, audio de-identification (audio de-ID), has been introduced with annotations for conversational data from Switchboard and Fisher (Cohn et al., 2019; Baril et al., 2022), but these datasets are not free. Audio de-ID is a special case of NEL where the entities of interest are related to personal identifiers.

We focus on English speech-related work (most comparable with our work), but there are also ongoing efforts for other languages (Tomashenko et al., 2019; Evain et al., 2021).

3 SLUE Phase-2: Tasks and data

This section introduces the tasks and metrics in SLUE Phase-2. The SLUE phase-1 introduced the "SLUE score", a numerical summary of model performance across tasks. However, as we consider a more diverse set of tasks, using the same pre-trained model for all tasks is difficult, and evaluation via a single SLUE score may discourage building systems for individual tasks. In SLUE Phase-2, therefore, we do not adopt the single SLUE score, and evaluate each task individually.

3.1 Tasks

We explore more diverse and complex tasks compared to SLUE phase-1. As an extension of NER task in SLUE, we describe the NEL task to predict the audio time-stamps of named entities. DAC is an utterance classification task within conversation interactions to predict dialog acts given input speech. We address two longer-range context tasks: QA and SUMM where the model takes a long sequence and utilizes context across the entire scope to answer questions or summarize speech respectively.

3.1.1 Dialog Act Classification (DAC)

DAC is the task of identifying the function of a given speech utterance in a dialog, such as question, statement or backchannel. It is an utterance-level multi-label multi-class classification task; that is, an utterance can have more than one class (function). We evaluate DAC using macro-averaged (unweighted) F1 score.

3.1.2 Question Answering (QA)

The goal of QA is to find the answer span in a spoken document given a spoken question. The answer span is denoted by the start and end frames of a short phrase in the document. We use the frame-level F1 (frame-F1) score (Chuang et al., 2020) to evaluate the overlap between the predicted and the ground-truth answer spans.

3.1.3 Speech summarization (SUMM)

SUMM refers to the task of generating a text summary from a given speech input. SUMM is challenging as it requires a model to assimilate information across very long input contexts in order to identify essential information and paraphrase to obtain the abstractive summary of speech. We evaluate SUMM using ROUGE (Lin, 2004), METEOR (Denkowski and Lavie, 2014) and BERTScore (Zhang* et al., 2020).

3.1.4 Named Entity Localization (NEL)

The goal of NEL is to predict the start and end times of any named entities in a spoken utterance. NEL is related to named entity recognition (NER), but NER involves identifying entity phrases while NEL involves locating them in the audio. We evaluate performance via two F1 scores based on the overlap between the predicted and ground-truth time ranges: *frame-F1*, defined similarly to the QA *frame-F1* measure; and *word-F1*, defined similarly to the de-identification metric of Cohn et al. (2019). The *word-F1* score has a hyperparameter $\rho \in [0, 1]$, which is the fraction of overlap between a ground-truth word segment and the predicted region needed to count the word as detected; $\rho = 1$ means a perfect match is required.

3.2 Datasets and annotation

3.2.1 SLUE-HVB for DAC

For the DAC task we adapt the Harper Valley Bank (HVB) spoken dialog corpus[3] (Wu et al., 2020) of scripted consumer banking dialogs, simulated by 59 speakers. The data contains about 23 hours of audio from 1,446 conversations with transcriptions and metadata, as well as dialog act annotation. However, the original DAC annotation is automatic, without manual validation, and the set of dialog acts is simple and tailored to this corpus. We define a new set of acts and collect human annotations by professional annotators listening to the audio. Our new set of dialog acts (See Table 9 in Appendix for detail) is based on the well-known Switchboard NXT (Calhoun et al., 2010) dialog act set. Based on a pilot annotation, we remove several unneeded labels and merge others unnecessarily granular. Finally, we split the HVB data into fine-tune, dev, and test sets (Table 1). The intent of conversation is

[3] CC-BY-4.0 license

balanced along the splits. We exclude short audio clips (<210ms) and audio that contains no speech.

Table 1: SLUE-HVB data statistics

	utterances	duration (h)
fine-tune	11,344	6.8
dev	1,690	1.0
test	6,121	3.6

3.2.2 SLUE-SQA-5 for QA

Previous open-source English spoken QA datasets, including Spoken SQuAD (Lee et al., 2018), NMSQA (Lin et al., 2022a), Spoken-CoQA (You et al., 2022), do not have a large training set consisting of realistic human speech, so we propose a new spoken QA dataset, SLUE-SQA-5, whose fine-tune, dev, and test sets all consist of real speech data.

The text transcriptions of question-answer pairs in SLUE-SQA-5 are collected from five different text QA datasets: SQuAD[4] (Rajpurkar et al., 2016), Natural Questions[5] (NQ) (Kwiatkowski et al., 2019), TriviaQA[6] (Joshi et al., 2017), WebQuestions[7] (WQ) (Berant et al., 2013), and CuratedTREC [8] (TREC) (Baudiš and Šedivỳ, 2015). We gather the text questions from the training set of the five text QA datasets as our fine-tune set. For our dev and test sets, we first collect the questions from the dev set of SQuAD, NQ, TriviaQA, WQ and the test set of TREC, and then randomly split these questions into two subsets as our dev and test sets. To get the spoken version of the collected questions, we used Amazon Mechanical Turk (MTurk), a crowdsourcing platform with anonymous, non-expert workers, to collect spoken questions read by human speakers. The collection details are shown in Section B.1 in the Appendix.

For the documents, to avoid the enormous cost of collecting spoken versions of long text documents, we search for off-the-shelf spoken documents relevant to each question as paired documents from the Spoken Wikipedia dataset [4] (Köhn et al., 2016), which includes 1.2k spoken Wikipedia articles from about 400 different real speakers. We split the articles in Spoken Wikipedia into about 37k spoken documents with duration of 40 seconds.

We adopt a similar procedure with Joshi et al. (2017) to search for relevant documents to the questions with their transcripts automatically. The detailed search criteria and the final number of SLUE-SQA-5 questions from each source text QA dataset are in Section B.2 and Table 11 in the Appendix.

To ensure the evaluation quality, we also asked human annotators to pick 408 question-document pairs, in which the document provides enough clues to answer the question, from test data as the verified-test set. The data statistics of SLUE-SQA-5 are in Table 2.

Table 2: SLUE-SQA-5 data statistics

	questions	documents	duration (h)	question speakers
fine-tune	46,186	15,148	244	931
dev	1,939	1,624	21.2	41
test	2,382	1,969	25.8	51
verified-test	408	322	4.2	51

3.2.3 SLUE-TED for SUMM

Of the existing corpora for abstractive speech summarization, How-2 has been used in recent work (Sharma et al., 2022). However, raw audio is not publicly available for the entire corpus, and the task of summarization is relatively easy due to shorter videos and simple reference summaries. Therefore, we consider the more challenging task of generating abstracts and titles for TED Talks, whose audio is publicly available. The TEDSummary dataset was introduced by (Kano et al., 2021) and accompanied by a tool to crawl and download TED talk videos from the web [9] that may be used to recreate the TEDSummary corpus. However, the lack of information about the exact talks used in the corpus makes it difficult to reproduce their data selection. Based on their crawler, and more recent talks released on the TED website[10], we introduce SLUE-TED, a re-designed corpus of summaries for TED Talks spanning the years until 2022.

We find that, on average, nearly 66% of words in the title and 57.4% of words in the abstract are present in the transcript of a given audio, suggesting that ASR pre-training can be useful to improve speech summarization performance. For benchmark evaluation, we randomly split this corpus into 80% finetune, 10% validation, and 10% test set as shown in Table 3. A detailed description of the dataset is available in the Appendix C.2.

[4] CC BY-SA 4.0 license
[5] CC BY-SA 3.0 license
[6] Apache License 2.0
[7] CC-BY 4.0 license
[8] Public Domain

[9] https://github.com/nttcslab-sp-admin/TEDSummary
[10] CC BY–NC–ND 4.0 license

Table 3: SLUE-TED data split

	utterances	duration (h)
finetune	3384	664
dev	425	81
test	424	84

3.2.4 SLUE-VoxPopuli for NEL

SLUE-VoxPopuli was published with NER annotations in SLUE (Shon et al., 2022a). We extend SLUE-VoxPopuli to NEL by adding word-level time stamps in the dev and test sets. We use the Montreal Forced Aligner (MFA) (McAuliffe et al., 2017) to obtain word-level time stamps, using MFA's public English acoustic model (McAuliffe and Sonderegger, 2022). MFA is a standard tool that is commonly used by the community to obtain ground-truth forced alignments. We manually verify the MFA produced entity alignments for 188 utterances (20% of the utterances with entity tags) in dev set and conclude that the MFA output provides a reliable ground-truth. We share more details for the data annotation and verification procedure in Appendix D.1. Data statistics for the SLUE-NEL data are shown in Table 4. Note that we do not publish NEL annotations for the *finetune* set as we focus on re-purposing NER models for NEL, which we believe is a more realistic use-case; as is also common for the speech-to-text forced alignment models, such as MFA, to be trained without ground-truth alignments.

Table 4: SLUE-NEL data statistics

	utterances	duration (h)	# w/ entity tags (# entity phrases)
dev	1,750	5.0	943 (1857)
test	1,838	4.9	1032 (1986)

4 Experiments and results

In the SLUE Phase-1 baseline experiments, larger pre-trained models and LM shallow fusion consistently gave better performance compared to smaller pre-trained models and without LM shallow fusion. Thus, in this paper, we analyze how the ASR word error rate (WER) in pipeline models is correlated with SLU task performance, by using multiple off-the-shelf open-source ASR models, specifically NeMo models (Kuchaiev et al., 2019) and Whisper (Radford et al., 2022). Additionally, we quantify the performance gain on WER and SLU tasks achieved by fine-tuning custom ASR models compared to using off-the-shelf ASR models.

In all experiments, we use the fine-tune set of the corresponding task to fine-tune pre-trained models, the dev set to pick the best model, and the test set to evaluate both E2E and pipeline baselines. In addition, we measure the performance of an "oracle" pipeline system that uses ground-truth transcripts instead of ASR output. Below, we use the *base* sized model when there are multiple variants of the pre-trained model.

4.1 DAC

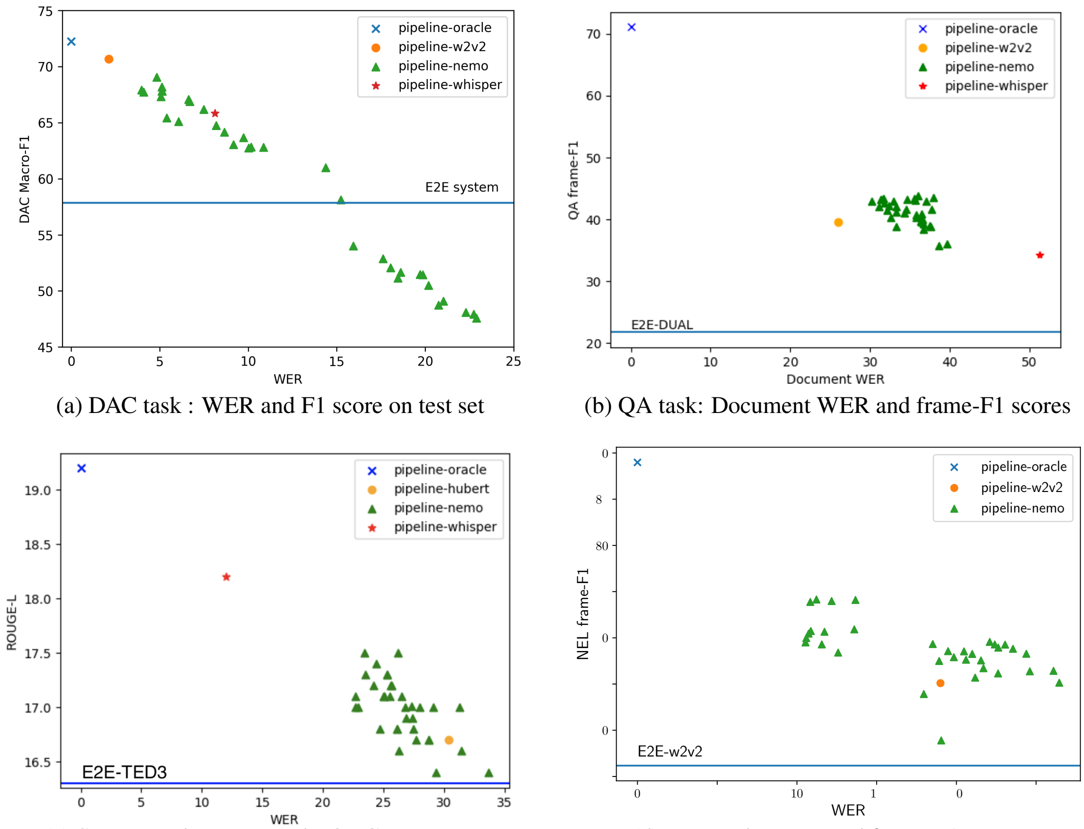
Baseline models: We follow a similar setup to the sentiment analysis baseline models in SLUE Phase-1 with some differences due to the multi-label nature of DAC. For the E2E baseline, we start with a pre-trained speech model, specifically wav2vec2 (Baevski et al., 2020), and add a self-attention pooling layer and two fully connected layers (including the output layer), with a Sigmoid output activation for each of the 18 dialog act classes. Outputs that is higher/lower than a threshold of 0.5 are classified as positive/negative for the corresponding class. For the pipeline baselines, we use either the off-the-shelf ASR models or an ASR using DAC data fine-tuned wav2vec2, and fine-tune a DeBERTa (He et al., 2020) model for the text classification.

Results: Table 5 shows the baseline results, and Figure 1a shows the relationship between WER and F1 score of pipeline models for a variety of ASR models (the ones used in Table 5 and all other NeMo models). We observe a strong correlation between the WER and DAC Macro F1 score (Pearson coorelation coefficient = -0.9). As the off-the-shelf ASR models perform well on conversational speech, fine-tuning the ASR model does not give a large improvement over the best NeMo model.

4.2 QA

Pipeline Approach: The pipeline QA system is composed of an ASR model and a text QA model predicting the start and end words of the answer span on the ASR output transcript.

We fine-tuned DeBERTa with the ground-truth transcripts of the SLUE-SQA-5 fine-tune set to get the text QA model of all pipeline systems. Note that the DeBERTa text QA models in pipeline systems and the text QA models used for searching paired



(a) DAC task : WER and F1 score on test set (b) QA task: Document WER and frame-F1 scores

(c) SUMM task : WER and ROUGE-L score (d) NEL task: WER and frame-F1 scores

Figure 1: WER sensitivity on NLP model performance

Table 5: DAC task baseline performance on test set. *the best NeMo model based on DAC F1 score is "stt-en-conformer-transducer-xxlarge"

System	Speech model	Text model	F1 score (WER)
pipeline-oracle	x	DeBERTa	72.3 (0.0)
pipeline-w2v2	wav2vec2	DeBERTa	70.7 (2.1)
pipeline-nemo	best model*	DeBERTa	69.1 (4.8)
pipeline-whisper	whisper-en	DeBERTa	65.8 (8.1)
E2E-w2v2	wav2vec2	x	57.9 (—-)

documents (please refer to Section B.2) were fine-tuned on different datasets: the former were tuned on the SLUE-SQA-5 fine-tune set while the latter were tuned on the external SQuAD dataset.

When evaluating pipeline systems on the SLUE-SQA-5 dev and test sets, we used MFA to align ground-truth transcripts and ASR output transcripts to speech. The ground-truth answer words and the answer words predicted by the text QA model are converted to the time interval of the ground-truth and predicted answer span, which were then used to calculate the frame-F1 score.

E2E Approach: We used DUAL (Lin et al., 2022a) as the QA E2E approach (denoted as E2E-DUAL). DUAL is composed of a wav2vec2-large model encoding speech waveforms, a k-means model converting wav2vec2 representations into cluster IDs, a Longformer model taking cluster IDs as input and predicting the start and end index of answer spans. We followed the training procedure in the DUAL paper except we used the k-means model of 500 clusters and fine-tuned its Longformer model for 45 epochs on the SLUE-SQA-5 fine-tune set.

Results: Table 6 shows the baseline results on the test and verified-test sets, and Figure 1b shows the relationship between document WER and frame-F1 on the test set of QA pipeline models. We observe a strong correlation (Pearson correlation coefficient=-0.89, p-value<0.01) between document WER and frame-F1. Pipeline-oracle significantly outperforms all the baseline models, and the performance gap is larger in the verified-test set, suggesting that there is room for improvement. Besides, the pipeline-w2v2 does not outperform the pipeline-nemo model, indicating that the fine-tuned ASR model does not lead to better QA performance.

Table 6: QA task baseline performance. *the best Nemo model based on frame-F1 score is "stt-en-contextnet-1024".

System	Speech model	Text model	Frame-F1	
			Test	Verified-test
pipeline-oracle	x	DeBERTa	62.3	70.3
pipeline-w2v2	wav2vec2	DeBERTa	39.6	40.1
pipeline-nemo	best model*	DeBERTa	43.3	45.9
pipeline-whisper	whisper-en	DeBERTa	32.7	35.7
E2E-DUAL	DUAL	x	21.8	23.1

4.3 SUMM

Pipeline Approach: The oracle pipeline is constructed by using the ground truth transcript to train a text summarization model, and infer the most likely summary from the ground truth transcript. Then, we use different combinations of speech recognizers and text summarization models to build different pipeline models for speech summarization. For the pipeline baseline, we train ASR models on the TEDLIUM-3 (Hernandez et al., 2018) corpus using the ESPNet (Watanabe et al., 2018) toolkit. The ASR models consist of a conformer encoder-decoder architecture with pre-trained SSL representations as features (see Appendix C.1 for more details about our models). We also experiment with state-of-the-art off-the-shelf speech recognizers, including Whisper (Radford et al., 2022) and NeMo models. The resulting talk transcripts are very long, often exceeding 2048 tokens, requiring our text summarization models to be able to handle such long input sequences. Therefore, we use the Longformer Encoder-Decoder (LED-large) model (Beltagy et al., 2020), initialized using BART-large model (Lewis et al., 2019). We investigate training our text summarisation model on both ground truth and ASR transcripts.

E2E Approach: E2E speech summarization model is trained using the ESPNet (Watanabe et al., 2018) toolkit by first pre-training for speech recognition task on the TEDLIUM-3 corpus (Hernandez et al., 2018) and then fine-tuning on our SLUE-TED data for speech summarization task as described in (Sharma et al., 2022).

Results: Table 7 shows the performance for all baseline models on the test set (see Appendix C.3 for dev set performance). We observe that the performance of the pipeline system can be improved by using a strong ASR model like Whisper. Further, we observe that the pipeline system performs slightly better when the text summarization model is fine-tuned on ASR transcripts. The pipeline models outperform the E2E system on ROUGE and METEOR, showing that the pipeline model aids in producing more accurate words. However, the end-to-end model does have a higher BERTScore, demonstrating the ability of the E2E model to produce semantically relevant summaries. All the baseline models perform worse than the pipeline-oracle model suggesting room for improvement.

To analyze the correlation between WER and the performance of the speech summarization task, we plot ROUGE-L scores in Figure 1c for various pipeline systems and a ground-truth transcript-based text summarization model. We observe a strong correlation (Pearson correlation coefficient=-0.9, p-value<0.01) between WER and ROUGE-L scores, suggesting that we can boost SUMM performance using a stronger ASR model.

To facilitate a better understanding of the performance of our E2E SUMM model, we analyze the percentage of exact matches in reference summary and predicted summaries for each POS tag. We observe that the majority of summarization errors occur because the model is not able to correctly generate the proper nouns in summary. A similar analysis on the percentage of exact matches for named entities shows that only 6.6% of entities in the reference summary were found in the predicted summary. Based on this analysis, we infer that the current speech summarization models struggle to correctly extract entities for the summary. (Full POS tags match available in Table 15 in the Appendix)

4.4 NEL

Baseline models: For NEL inference, we use the baseline NER models from Shon et al. (2022a). Both the E2E and ASR (within pipeline) models use wav2vec2 as the backbone and are trained with character-level connectionist temporal classification (CTC) (Graves et al., 2006). The text NER (within pipeline) model uses the DeBERTa as the backbone and is trained on ground-truth transcripts. Note that no dedicated model is trained for NEL. This is intentional: NER and NEL are related tasks and a realistic use case would require a single model that performs both tasks.

Inference: A CTC model produces a posterior

Table 7: SUMM task baseline performance. The ASR models are trained on the TEDLIUM-3 corpus. *the best NeMo model based on SUMM ROUGE-L score is "conformer-transducer-xxlarge". For pipeline models, we also experiment with training NLU model on ASR Transcripts (ASR) instead of ground truth transcript.

System	Speech model	Text model	ROUGE-1	ROUGE-2	ROUGE-L	METEOR	BERTScore	WER
pipeline-oracle	x	LED	30.1	7.7	19.3	13.7	83.8	0.0
pipeline-w2v2	wav2vec2-ASR	LED	26.8	5.1	16.8	12.4	82.5	34.4
pipeline-hubert	Hubert-ASR	LED	26.9	5.3	16.7	12.6	82.5	30.4
pipeline-nemo	best model*	LED	27.6	6.2	17.5	13.0	82.4	23.4
pipeline-whisper	whisper-en	LED	28.6	6.7	18.2	12.9	83.4	12.0
pipeline-whisper ASR	whisper-en	LED(ASR)	29.0	7.0	18.6	13.0	83.7	12.0
E2E-TED3	TEDLIUM-3 Conformer	x	23.8	5.1	16.3	11.7	84.0	—

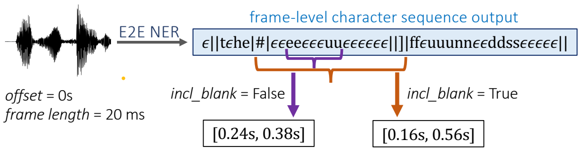


Figure 2: Example inference for an E2E NEL model using a CTC recognizer. The transcript is “the eu funds”. ‘#’ and ‘]’ are the start and end labels of an ORG entity.

probability matrix, $\mathcal{E} \in \mathbb{R}^{T \times V}$, consisting of the posterior of each character in the vocabulary of size V for each of the T frames in the input audio. For ASR, the character vocabulary consists of the English alphabet, a word separator token “|”, and a blank token“ϵ”. For the E2E model, the vocabulary also includes special characters for the start and end of an entity phrase. We obtain a frame-level character sequence output via greedy decoding on $\mathcal{E}$. The time stamps corresponding to “|” tokens in the output character sequence provide word-level start and end boundaries. As CTC is not trained with an explicit alignment signal, the word boundary tokens may not be a reliable indicator of the true time stamps, and we introduce two hyperparameters as a heuristic fix for possible mis-alignments: *offset* is a fixed duration by which we shift the time stamp predictions, and *incl_blank* $\in \{0, 1\}$ denotes whether any trailing ϵ tokens are considered a part of the predicted entity segment.

In the pipeline approach, the predicted text from ASR is passed to a text NER model, and the time stamps for detected entities are extracted from the ASR’s $\mathcal{E}$. For the E2E model, the time stamps corresponding to the entity start and end special characters are extracted directly from its $\mathcal{E}$. An example is presented in Fig. 2.

Results: Table 8 presents the baseline results. The pipeline and E2E baselines have fairly similar frame-F1, but these approaches have complementary strengths as seen from their precision and

Table 8: NEL task baseline performance on test set. The W2V2-B models are fine-tuned on slue-voxpopuli data.*the best nemo model based on NEL frame-f1 score on dev is “stt_en_conformer_ctc_small"

System	Speech model	Text model	frame-F1	word-F1 (ρ=0.8)
pipeline-oracle	x	DeBERTa	89.0	90.0
pipeline-w2v2	wav2vec2	DeBERTa	65.2	72.0
E2E-w2v2	wav2vec2	x	56.3	59.6
pipeline-nemo	best model*	DeBERTa	74.1	81.4

recall values (see Table 18, Appendix D.3). We also find that the off-the-shelf NeMo ASR model (*pipeline-nemo*) outperforms the dataset-specific ASR model (*pipeline-w2v2*).[11]

Figure 1d shows a scatter plot of NEL and WER scores for a variety of pipeline models. Although models with the lowest WER do have the best frame-F1, the overall correlation is not high. The NeMo models have different training objectives and model architectures, and we note that within each model class, the ASR and NEL metrics are much better correlated (see Figure 12, Appendix D.3). This suggests that model architecture and/or training objective play a significant role in alignment quality.[12]

5 Discussion

Among the baseline models, our pipeline models generally outperform their end-to-end counterparts. However, as shown in prior work (e.g., (Arora et al., 2022a; Pasad et al., 2021)), end-to-end models often have more room for improvement with careful and creative modeling ideas, and we hope that this new testbed helps spur such research.

In addition, the WER sensitivity analysis in Figure 1 suggests different strategies are needed for the

[11] More word-F1 results in Tab. 19 in Appendix D.4.

[12] The details of hyperparameter tuning and timestamp extraction from NeMo models are in Appendix D.2.

pipeline system depending on the SLU task. For example, fine-tuned ASR (pipeline-w2v2) plays a significant role in the DAC task while the QA task is not, and ASR model architecture is critical for the NEL task while WER is more matter for DAC and SUMM tasks.

6 Conclusion

SLUE Phase-2, with four additional SLU tasks and high-quality annotation, enables a more comprehensive analysis of diverse SLU tasks than previously possible. Besides the task definitions and annotations, this work contributes multiple baselines and performance analysis using modern off-the-shelf ASR and text models. The baseline performance on all tasks is far from perfect, and the relative performance of different models differs across tasks, indicating that these tasks are ripe for additional work and analysis to push the boundary of SLU research.

Limitations

One limitation of this work is the lack of human performance scores on the new tasks. Although the baseline performance is far from perfect, and it seems quite likely that human performance is much better, this should be measured in future work. Another limitation is that it is unknown how much each task should benefit from access to the audio in addition to text; this could be measured in principle for humans, but again we leave this to future work.

Broader Impact and Ethics

Spoken language understanding benchmarks, like the ones we propose in this work, facilitate the development of technologies that may be particularly useful for speakers who are unable to read or write text and ultimately also for unwritten languages, where speech is the only form of communication. We hope that this work also spurs more collaboration across the fields of speech and natural language processing, both of which are needed to make progress in this area.

We ensured that the SLUE-SQA speech data collection from AMT was conducted with a higher wage (on average, US$10 per hour) than the US federal minimum wage. This wage includes compensation for the time spent on re-recording and addressing technical issues on the recording platform. We further took measures to ensure that our data collection and annotation process did not introduce any potential biases in the SLUE Phase-2 benchmark. Specifically, for SLUE-SQA, we implemented an automatic check using the Google Speech-to-Text service. If the Word Error Rate (WER) exceeded 30%, workers were recommended to re-record the utterance. We chose a 30% WER threshold to identify and exclude empty or prematurely cut utterances. Our analysis showed that such violations were less than 8% of questions. Additionally, we personally listened to each recording and only discarded those where a significant portion of the content was missing. Recordings were accepted even if the WER exceeded 30%, ensuring that our dataset does not include any potential bias inherent in the automated speech-to-text service.

The DAC annotation in SLUE-HVB and verified-test set in SLUE-SQA data were done by ASAPP internal data labeling team. Everyone who participated in the annotation was an employee of ASAPP and conducted the work within the scope of their usual employment. Specifically, most of them have over 1 year of experience in speech and language-related data labeling and their education level is above a Master's degree.

Acknowledgements

We would like to thank Kyle Hager, and Molly Ruhl for their helpful comments and discussion from a linguistic perspective, and the whole ASAPP MLDL team members for high quality annotation. Part of this work used PSC Bridges2 and NCSA Delta through allocations CIS210014 and IRI120015 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

References

Siddhant Arora, Siddharth Dalmia, Xuankai Chang, Brian Yan, Alan W. Black, and Shinji Watanabe. 2022a. Two-pass low latency end-to-end spoken language understanding. In *Interspeech*.

Siddhant Arora, Siddharth Dalmia, Pavel Denisov, Xuankai Chang, Yushi Ueda, Yifan Peng, Yuekai Zhang, Sujay Kumar, Karthik Ganesan, Brian Yan, et al. 2022b. ESPnet-SLU: Advancing spoken language understanding through ESPnet. In *ICASSP*.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning*.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *NeurIPS*.

Guillaume Baril, Patrick Cardinal, and Alessandro Lameiras Koerich. 2022. Named entity recognition for audio de-identification. *arXiv preprint arXiv:2204.12622*.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. SLURP: A spoken language understanding resource package. In *EMNLP*.

Petr Baudiš and Jan Šedivỳ. 2015. Modeling of the question answering task in the YodaQA system. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 222–228.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *EMNLP*.

Paul Boersma and David Weenink. 2009. Praat: doing phonetics by computer (version 5.1.13).

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. In *Language resources and evaluation*.

Sasha Calhoun, Jean Carletta, Jason M Brenier, Neil Mayo, Dan Jurafsky, Mark Steedman, and David Beaver. 2010. The NXT-format Switchboard corpus: a rich resource for investigating the syntax, semantics, pragmatics and prosody of dialogue. *Language resources and evaluation*.

Eric Y. Chen, Zhiyun Lu, Hao Xu, Liangliang Cao, Yu Zhang, and James Fan. 2020a. A large scale speech sentiment corpus. In *Language resources and evaluation*.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Micheal Zeng, and Furu Wei. 2021. WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16:1505–1518.

Xi Leslie Chen, Sarah Ita Levitan, Michelle Levine, Marko Mandic, and Julia Hirschberg. 2020b. Acoustic-prosodic and lexical cues to deception and trust: deciphering how people detect lies. *Transactions of the Association for Computational Linguistics*, 8:199–214.

Yung-Sung Chuang, Chi-Liang Liu, Hung-Yi Lee, and Lin-shan Lee. 2020. SpeechBERT: An audio-and-text jointly learned language model for end-to-end spoken question answering. In *Interspeech*.

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. In *NAACL*.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv:1805.10190*.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.

Solène Evain, Ha Nguyen, Hang Le, Marcely Zanon Boito, Salima Mdhaffar, Sina Alisamir, Ziyi Tong, Natalia Tomashenko, Marco Dinarelli, Titouan Parcollet, et al. 2021. LeBenchmark: A reproducible framework for assessing self-supervised representation learning from speech. In *Interspeech*.

Lingyun Feng, Jianwei Yu, Deng Cai, Songxiang Liu, Haitao Zheng, and Yan Wang. 2021. ASR-GLUE: A New Multi-task Benchmark for ASR-Robust Natural Language Understanding. *arXiv:2108.13048*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *ICLR*.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language*.

Franç ois Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In *Speech and Computer*, pages 198–208. Springer International Publishing.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *arXiv:2106.07447*.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The icsi meeting corpus. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, volume 1, pages I–I.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*.

Dan Jurafsky, Elizabeth Shriberg, Barbara Fox, and Traci Curl. 1998. Lexical, prosodic, and syntactic cues for dialog acts. In *Discourse Relations and Discourse Markers*.

Takatomo Kano, Atsunori Ogawa, Marc Delcroix, and Shinji Watanabe. 2021. Attention-based multi-hypothesis fusion for speech summarization. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*.

Arne Köhn, Florian Stegen, and Timo Baumann. 2016. Mining the spoken Wikipedia for speech data and beyond. In *Language Resources and Evaluation*, Paris, France. European Language Resources Association (ELRA).

Oleksii Kuchaiev, Jason Li, Huyen Nguyen, Oleksii Hrinchuk, Ryan Leary, Boris Ginsburg, Samuel Kriman, Stanislav Beliaev, Vitaly Lavrukhin, Jack Cook, et al. 2019. NeMo: a toolkit for building AI applications using neural modules. *arXiv preprint arXiv:1909.09577*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Cheng-I Lai, Yung-Sung Chuang, Hung yi Lee, Shang-Wen Li, and James R. Glass. 2020. Semi-supervised spoken language understanding via self-supervised speech and language model pretraining. *ICASSP*.

Chia-Hsuan Lee, Szu-Lin Wu, Chi-Liang Liu, and Hung-yi Lee. 2018. Spoken SQuAD: A study of mitigating the impact of speech recognition errors on listening comprehension. *Interspeech*, pages 3459–3463.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Guan-Ting Lin, Yung-Sung Chuang, Ho-Lam Chung, Shu-wen Yang, Hsuan-Jui Chen, Shang-Wen Li, Abdelrahman Mohamed, Hung-yi Lee, and Lin-shan Lee. 2022a. Dual: Textless spoken question answering with speech discrete unit adaptive learning. *arXiv preprint arXiv:2203.04911*.

Tzu-Quan Lin, Hung-yi Lee, and Hao Tang. 2022b. Melhubert: A simplified hubert on mel spectrogram. *arXiv preprint arXiv:2211.09944*.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. *arXiv preprint arXiv:1903.05566*.

Loren Lugosch, Piyush Papreja, Mirco Ravanelli, Abdelwahab Heba, and Titouan Parcollet. 2021a. Timers and such: A practical benchmark for spoken language understanding with numbers. *arXiv preprint arXiv:2104.01604*.

Loren Lugosch, Piyush Papreja, Mirco Ravanelli, Abdelwahab Heba, and Titouan Parcollet. 2021b. Timers and such: A practical benchmark for spoken language understanding with numbers. *ArXiv*, abs/2104.01604.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. 2019. Speech model pre-training for end-to-end spoken language understanding. In *INTERSPEECH*.

Luz Martinez-Lucas, Mohammed Abdelwahab, and Carlos Busso. 2020. The MSP-Conversation corpus. In *INTERSPEECH*.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. 2017. Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. In *Interspeech*, pages 498–502.

Michael McAuliffe and Morgan Sonderegger. 2022. English mfa acoustic model v2.0.0. Technical report, `https://mfa-models.readthedocs.io/acoustic/English/EnglishMFAacousticmodelv2_0_0.html`.

I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, W. Post, Dennis Reidsma, and P. Wellner. 2005. The ami meeting corpus. In *Proceedings of Measuring Behavior 2005, 5th International Conference on Methods and Techniques in Behavioral Research*, pages 137–140. Noldus Information Technology.

Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D. Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, Tara N. Sainath, and Shinji Watanabe. 2022. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1179–1210.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *ICASSP*.

Ankita Pasad, Felix Wu, Suwon Shon, Karen Livescu, and Kyu J. Han. 2021. On the use of external data for spoken named entity recognition. In *North American Chapter of the Association for Computational Linguistics*.

Yifan Peng, Siddhant Arora, Yosuke Higuchi, Yushi Ueda, Sujay Kumar, Karthik Ganesan, Siddharth Dalmia, Xuankai Chang, and Shinji Watanabe. 2022. A study on the integration of pre-trained ssl, asr, lm and slu models for spoken language understanding. *arXiv preprint arXiv:2211.05869*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. 2018. How2: a large-scale dataset for multimodal language understanding. *arXiv preprint arXiv:1811.00347*.

Roshan Sharma, Shruti Palaskar, Alan W Black, and Florian Metze. 2022. End-to-end speech summarization using restricted self-attention. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8072–8076. IEEE.

Suwon Shon, Ankita Pasad, Felix Wu, Pablo Brusco, Yoav Artzi, Karen Livescu, and Kyu J Han. 2022a. Slue: New benchmark tasks for spoken language understanding evaluation on natural speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7927–7931. IEEE.

Suwon Shon, Felix Wu, Kwangyoun Kim, Prashant Sridhar, Karen Livescu, and Shinji Watanabe. 2022b. Context-aware fine-tuning of self-supervised speech models. *arXiv preprint arXiv:2212.08542*.

Paden Tomasello, Akshat Shrivastava, Daniel Lazar, Po-Chun Hsu, Duc Le, Adithya Sagar, Ali Elkahky, Jade Copet, Wei-Ning Hsu, Yossef Mordechay, et al. 2022. Stop: A dataset for spoken task oriented semantic parsing. *arXiv preprint arXiv:2207.10643*.

Natalia Tomashenko, Antoine Caubrière, Yannick Estève, Antoine Laurent, and Emmanuel Morin. 2019. Recent advances in end-to-end spoken language understanding. In *7th International Conference on Statistical Language and Speech Processing (SLSP)*.

Trang Tran, Shubham Toshniwal, Mohit Bansal, Kevin Gimpel, Karen Livescu, and Mari Ostendorf. 2018. Parsing speech: A neural approach to integrating lexical and acoustic-prosodic information. In *Proceedings of NAACL-HLT*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation. *arXiv:2101.00390*.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. ESPNet: End-to-end speech processing toolkit. In *INTERSPEECH*.

Felix Wu, Kwangyoun Kim, Jing Pan, Kyu J Han, Kilian Q Weinberger, and Yoav Artzi. 2022a. Performance-efficiency trade-offs in unsupervised pre-training for speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7667–7671. IEEE.

Felix Wu, Kwangyoun Kim, Shinji Watanabe, Kyu Han, Ryan McDonald, Kilian Q Weinberger, and Yoav Artzi. 2022b. Wav2seq: Pre-training speech-to-text encoder-decoder models using pseudo languages. *arXiv preprint arXiv:2205.01086*.

Mike Wu, Jonathan Nafziger, Anthony Scodary, and Andrew Maas. 2020. Harpervalleybank: A domain-specific spoken dialog corpus. *arXiv preprint arXiv:2010.13929*.

Hemant Yadav, Sreyan Ghosh, Yi Yu, and Rajiv Ratn Shah. 2020. End-to-end named entity recognition from English speech. In *INTERSPEECH*.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. SUPERB: Speech processing universal performance benchmark. In *INTERSPEECH*.

Chenyu You, Nuo Chen, Fenglin Liu, Shen Ge, Xian Wu, and Yuexian Zou. 2022. End-to-end spoken conversational question answering: Task, dataset and model. *arXiv preprint arXiv:2204.14272*.

Amir Zadeh, Paul Pu Liang, Jonathan Vanbriesen, Soujanya Poria, Edmund Tong, Erik Cambria, Minghai Chen, and Louis Philippe Morency. 2018. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *ACL*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Appendix

A DAC

A.1 Dialog act list

Figure 4 shows the corelation between WER and F1 score on dev set. Table 10 shows the experiment result including dev set.

Table 9: Dialog acts detail

actions	sub-actions	Definition	example
question	**question_check**	Questions that check/verify information unique to a listener	What is your address?
	question_repeat	Requests for someone to repeat what they said in order to clarify/understand	Can you repeat that please?
	question_general	All other questions	How can I help you today?
answer	**answer_agree**	Answers indicating a positive response or acceptance	Yeah, let's do that
	answer_dis	Answers indicating a negative response or denial	No, that's okay
	answer_general	All other answers	
statement	**apology**	A number of often-templated utterances indicating a speaker is apologetic	I'm sorry to hear that!
	thanks	A number of often-templated utterances indicating a speaker is appreciative	Thanks for doing that
	acknowledge	A response indicating that a speaker has heard, or is empathizing with, what another speaker has said	Ok / I understand
	statement_open	Formulaic opening statements that might contain a greeting, introduction, or some other pleasantries	Hi my name is XX
	statement_close	Formulaic closing statements indicating that the conversation is coming to an end, often containing salutations	Have a great day
	statement_problem	An utterance that contains a user's primary reason for calling in (this may include questions if the question clearly indicates the call reason)	I lost my debit card / I just called in because I wanted to know what are my local branch hours?
	statement_instruct	An imperative utterance that indicates the speaker wants the listener to do something	Go to the website and log in / You'll need to upload a copy of your form
	statement_general	All other statements	
natural speech	**backchannel**	Verbal or non-verbal expressions indicating the listener's attention, agreement, or understanding, while not having much significant meaning on their own	uh-huh / is that right?
	disfluency	filler, reparandum, interregnum	Uh../ uh no... / debit uh no (credit card)
	self	Essentially rhetorical utterances, or utterances where a speaker is not expecting a response from the listener (i.e. talking to one's self)	Oh, look at me I've forgotten which button to press here / Hmm now where did I put that other number...
other	**other**	Any utterances that don't fit in any of the above categories, including noise, gibberish, or otherwise uninterpretable speech	[noise] / fjdskl / /////////

A.2 Annotation detail

Figure 3 shows the annotation interface for DAC. Annotator could choose multiple acts per utterance. The annotator could listen to the corresponding speech segment for better judgment. Utterances are provided in chronologically by combining agent and caller channels. A single conversation was annotated by a single annotator. The total conversation was divided into 40 shards with evenly distributed intent of the conversation. A total of 5 annotators completed the annotation and we did not collect personal information such as the demographic or geographic background of the annotator.[13]

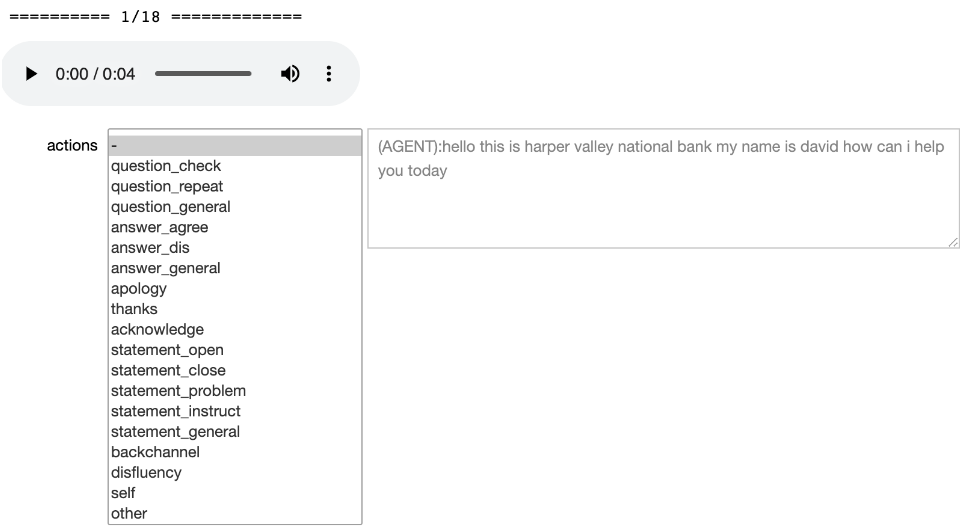


Figure 3: DAC annotation tool interface

A.3 Model training details

The E2E model fine-tuning was done with 2e-05 learning rate, 50,000 maximum update step and 2,800,000 maximum tokens for mini-batch. We use the macro-f1 score of dev set to choose the final model evaluation. We use single RTX GPU and took 2hours. Model training was done with 5 different random seed and reported median model. For pipeline system, wav2vec2 ASR model fine-tuning took 10 hours and DeBERTa NLP model took 3 hours using the same GPU. We followed the ASR and NLP fine-tuning script in SLUE-Toolkit. Reproducible baseline scripts will be released.

A.4 Additional results

Figure 4 shows the WER and F1 score on dev set and it shows the same trend compared to test set presented in Figure 1a. Table 10 shows DAC task performance evaluation including dev and test set.

Table 10: DAC task baseline performance. *the best NeMo model based on DAC F1 score is "conformer-transducer-xxlarge"

System	Speech model	Text model	F1 score (WER)	
			Dev	Test
pipeline-oracle	x	DeBERTa	76.1 (0.0)	72.3 (0.0)
pipeline-w2v2	wav2vec2	DeBERTa	72.6 (2.5)	70.7 (2.1)
pipeline-nemo	best model*	DeBERTa	72.2 (4.8)	69.1 (4.8)
pipeline-whisper	whisper-en	DeBERTa	66.1 (9.7)	65.8 (8.1)
E2E-w2v2	wav2vec2	x	57.4 (—-)	57.9 (—-)

[13]annotators generally follows the principles here: https://datapractices.org/manifesto/#principles

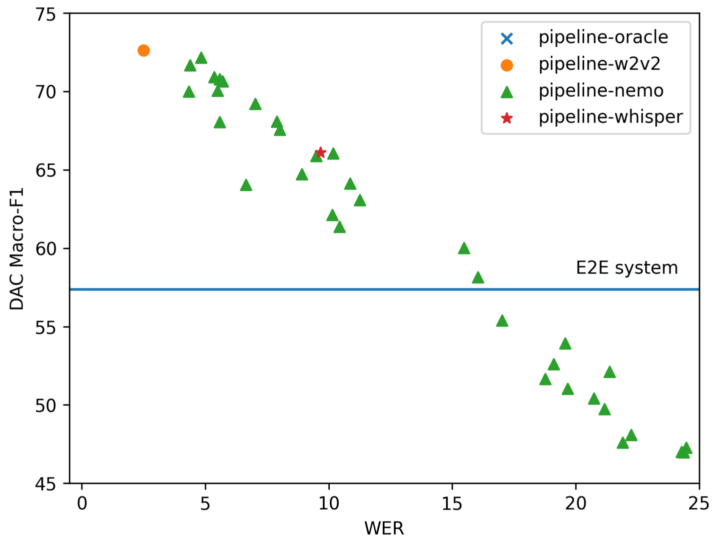


Figure 4: DAC task: WER and F1 scores on dev set

B QA

B.1 Spoken question collection details

To collection spoken questions in SLUE-SQA-5, we posted our own speech collection website to Mturk, asked each worker to read 50 questions and paid them 1 dollar, so the worker got 2 cents for reading one question. After the worker record their speech, our speech collection website uses Google Speech-to-Text service to transcribe the audio to text and calculate the WER. If the WER is higher than 30%, our website will notify the worker and suggest them recording again. In our manual check, we listened to every recording by ourselves and discarded a recording only when we found that a high portion of the content was missing; otherwise, we still accepted it even if the WER was over 30%. The interface of our speech collection website is shown in Figure 5.

B.2 Search criteria of SLUE-SQA-5 documents

When searching for the paired document to each question, we determined whether a document is relevant to a question by jointly considering (1) its rank among all documents in BM25 (Robertson et al., 2009) search, a common term-based retrieval algorithm that scores the relevance between texts via keyword matching, (2) its rank among all documents in semantic search with the sentence-transformers model[14] (Reimers and Gurevych, 2019), a neural sentence-level semantic encoder pre-trained on 215M QA pairs from multiple datasets, and (3) word-F1 derived by passing the question and the document through three different text QA models[15][16][17] fine-tuned on SQuAD dataset. We discard a question if we found no relevant document for it.

In specific, for each question, we searched for documents that meet all the criteria listed below:

- The document transcript includes the answer string to the question.
- The document has one of the top-1000 highest BM25 scores with the question among all documents.
- The document has one of the top-100 highest relevance scores with the question among all documents in semantic search with the sentence-transformers model.

[14] https://huggingface.co/sentence-transformers/multi-qa-mpnet-base-dot-v1
[15] https://huggingface.co/Palak/microsoft_deberta-large_squad
[16] https://huggingface.co/deepset/deberta-v3-large-squad2
[17] https://huggingface.co/deepset/deberta-v3-base-squad2

- When we pass the question and document through the three pre-trained text QA models mentioned in Section 3.2.2, at least one model gets a non-zero word-F1 score. (This criterion is used for dev and test set questions only.)

If there exists a document that meet all the above criteria, we combine the document, question, and the question's answer into a question-answer-document triplet. Otherwise, we consider the question unanswerable and discard it. Note that we limit the number of paired document per question to one. If we find multiple documents that meet the criteria, we will choose the one with highest relevance score in semantic search among them as the paired document.

B.3 Model training details

The E2E-DUAL model is composed of a wav2vec2-large model encoding speech waveforms, a k-means model converting wav2vec2 layer representations into cluster IDs, and a Longformer model taking cluster IDs as input and predicting the start and end index of answer spans. We extract the representations of Librispeech (Panayotov et al., 2015) train-clean-100 set from the 22nd layer of the fixed wav2vec2-large model to train the k-means model. The k-means model is then used to convert the representations of SLUE-SQA-5 fine-tune set into discrete units, which are taken as the input to the Longformer model. We fine-tune Longformer with 1e-4 learning rate, 500 warmup steps and overall 128 batch size on 4 Tesla V100 gpus. It takes around 40 hours to fine-tune the Longformer model for 45 epochs. The total number of tuned parameters in DUAL, including the k-means model and Longformer part, is reported in Table 21.

For the pipeline system, we fine-tune the wav2vec2 ASR model with 1e-4 learning rate and 16 batch size for 10 epochs, and fine-tune the DeBERTa NLP model with 4e-5 learning rate, 100 warmup steps and 64 batch size for 10 epochs. Wav2vec2 ASR model fine-tuning takes 25 hours and DeBERTa NLP model takes 6.5 hours using one V100 gpu.

B.4 Additional results

Figure 6 shows the relationship between the question WER and frame-F1 on the test set. We observe relatively weak correlation between question WER and frame-F1 compared to that between document WER and frame-F1.

Table 12 shows the QA performance on the dev set. Figure 7 shows the relationship between document WER and frame-F1 on the dev set and has the similar trend (Pearson correlation coefficient=-0.94, p-value<0.01) compared to the test set in Figure 1b. Figure 8 shows the relationship between question WER and frame-F1 on the dev set. Similar to the test set, we observe relatively weak correlation between question WER and frame-F1 compared to that between document WER and frame-F1.

Table 11: Number of SLUE-SQA-5 questions from each source text QA datasets.

	SQuAD	NQ	TriviaQA	WQ	TREC	total
fine-tune	11,900	12,383	20,358	1063	482	46,186
dev	679	85	869	212	94	1,939
test	828	125	1,051	266	112	2,382
verified-test	185	20	135	43	25	408

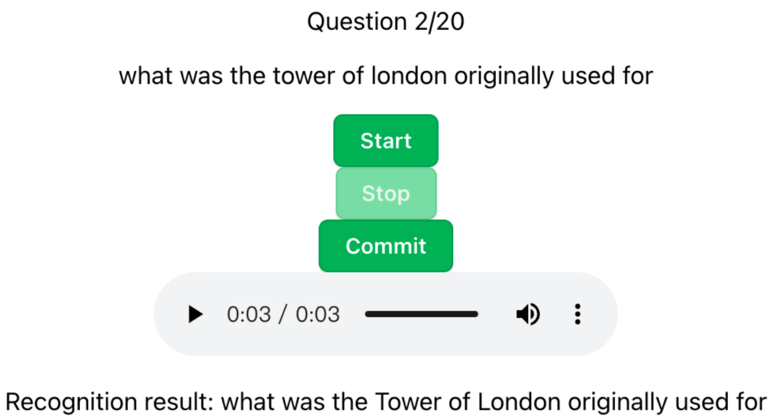


Figure 5: Interface of the website for spoken question collection in SLUE-SQA-5 dataset.

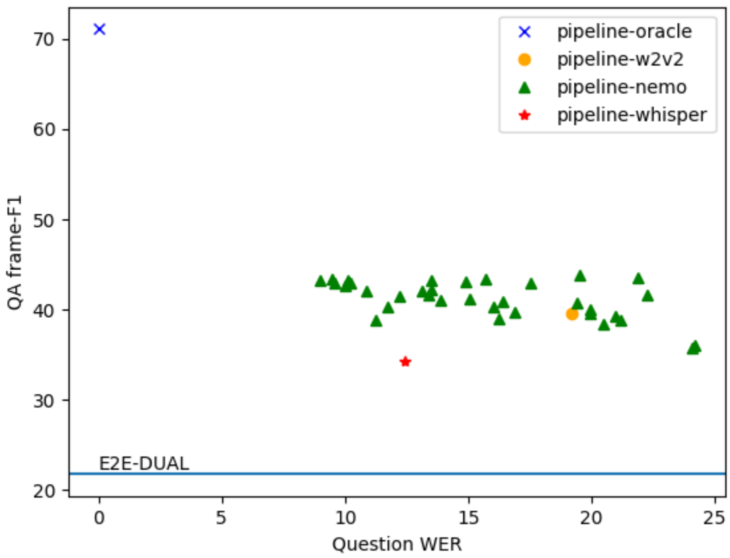


Figure 6: QA task: Question WER and frame-F1 scores

Table 12: QA task baseline performance on the dev set. *the best Nemo model based on frame-F1 score is "stt-en-contextnet-1024".

System	Speech model	Text model	Frame-F1
			Dev
pipeline-oracle	x	DeBERTa	68.5
pipeline-w2v2	wav2vec2	DeBERTa	41.8
pipeline-nemo	best model*	DeBERTa	49.2
pipeline-whisper	whisper-en	DeBERTa	35.2
E2E-DUAL	DUAL	x	24.4

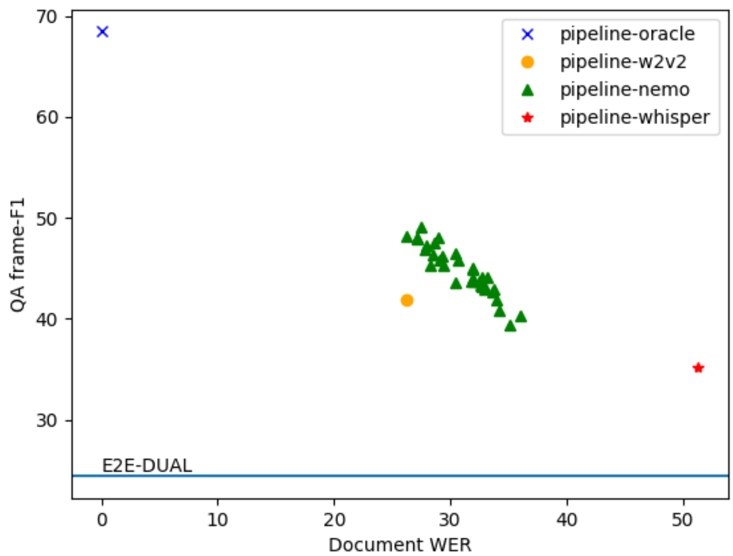


Figure 7: QA task: Document WER and frame-F1 scores on the dev set

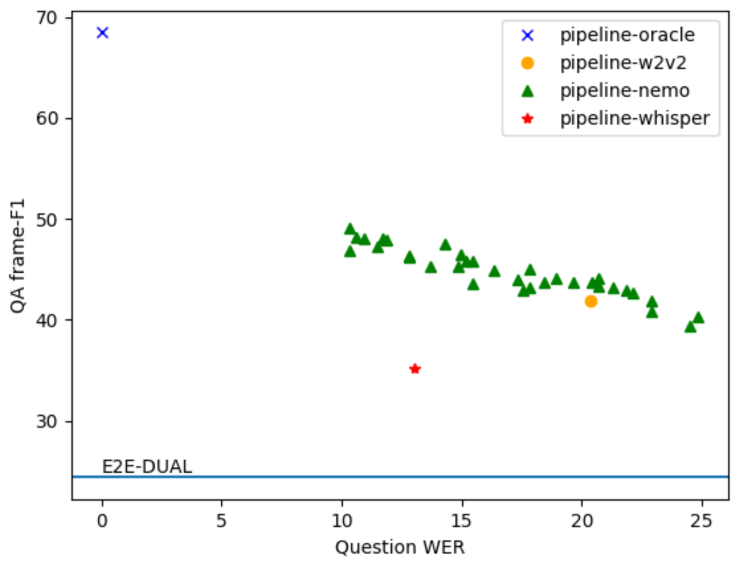


Figure 8: QA task: Question WER and frame-F1 scores on the dev set

C SUMM

C.1 Model details

The ASR models consist of a conformer encoder-decoder architecture with pre-trained SSL representations like Hubert large (Hsu et al., 2021) and wav2vec2 large (Baevski et al., 2020) representations as features. Following prior work (Peng et al., 2022), a weighted sum of multiple hidden states of SSL models is utilized. Since the TED talks are very long, we break the audio into 10 second chunks, and infer the most likely transcript for each chunk independently. Then we concatenate the resulting transcripts from each audio chunk to obtain the talk transcript. ASR models were trained for nearly 23 hours on 4 v100 gpus.

The E2E speech summarization model has similar architecture as the ASR model of the pipeline baseline. Since the TED talks were too long to fit the entire speech input on a GPU, we use only the last hidden state of SSL model and trained our E2E model using only the first 30000 speech frames (600 seconds). E2E speech summarization model was trained for nearly 16 hours on 4 v100 gpus.

For Nemo conformer and squeezeformer models, the audio is too long to perform inference using a GPU, and hence we have to break audio input into 5-minute chunks and perform inference separately on each of these chunks.

C.2 Additional dataset details

Table 13 summarizes the statistics of the dataset, and the distribution of ground truth transcript and summaries is shown in Figure 9. We observe that this dataset contains much longer audios and transcripts than prior works.

Table 13: SLUE-TED data statistics

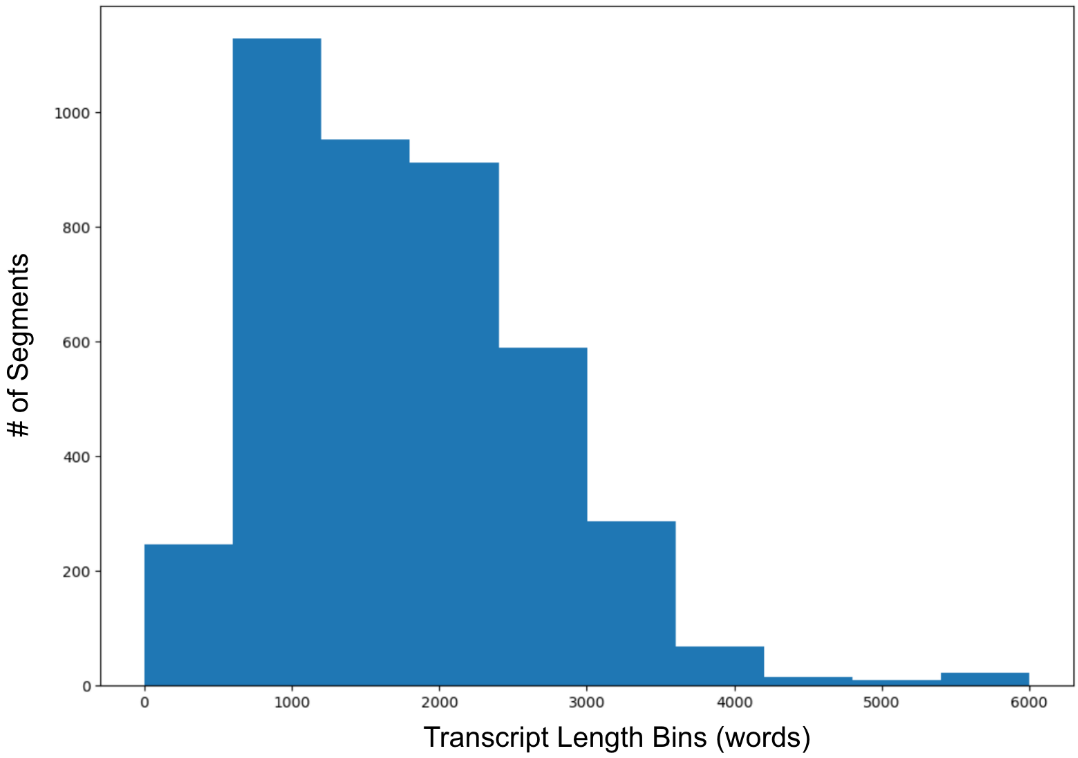
Corpus	utterances	duration (h)	duration/utt (s)	Transcript length (words)	Summary length (words)
How2	79114	1890	86	853	60
SLUE-TED	4233	829	705	1757	61

C.3 Additional results

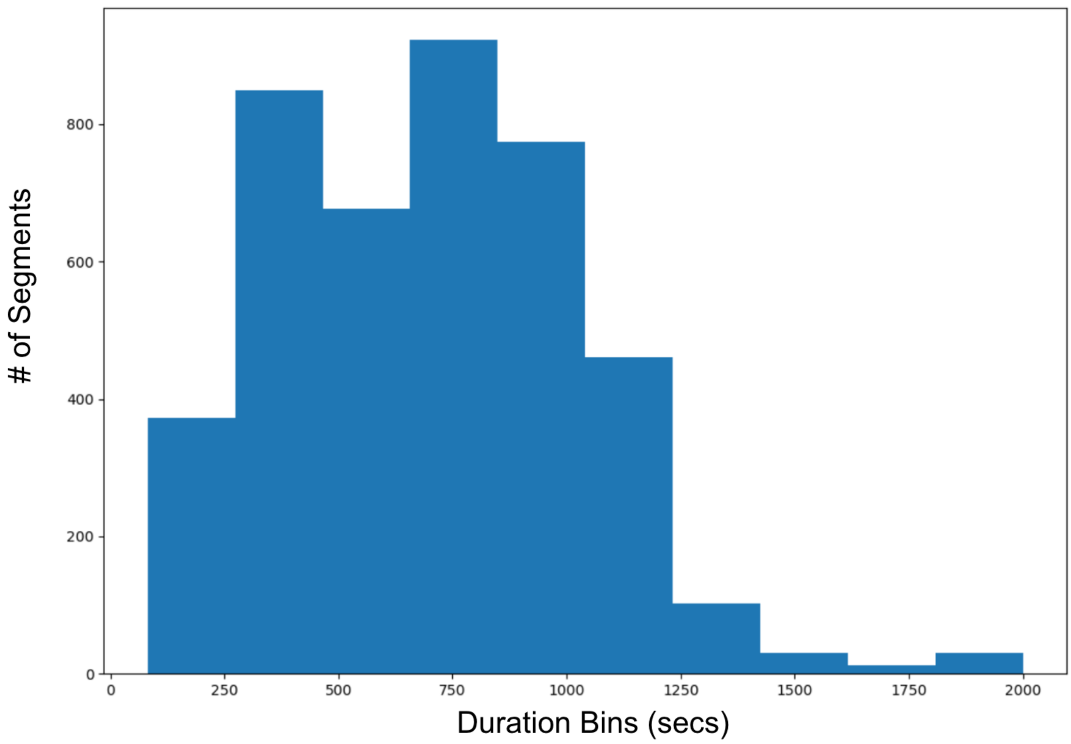
Table 14 shows the performance of all the models on the dev set. Figure 10 shows the correlation between WER and ROUGE-L scores on the dev set and has a similar trend to the one observed on test set in figure 1c. Table 15 show the percentage of exact matches in reference summary and predicted summaries for each POS tag on the test set. We further analyzed the performance of our E2E Summ model separately on abstract and title in summary and observed that the model performs slightly better at generating title (ROUGE-L:15.2, BERTScore:87.7) as compared to generating the abstract (ROUGE-L:14.4, BERTScore:83.4). Table 16 provides example summaries generated by our baseline systems. We observe that pipeline models generate more accurate words while E2E model generates more semantically similar summaries to reference. However, both these models generate summaries that differ from references suggesting significant room for improvement.

Table 14: SUMM task baseline performance on the dev set. The ASR models are trained on the TEDLIUM-3 corpus. For pipeline models, we also experiment with training NLU model on ASR Transcripts (ASR) instead of ground truth transcript. *the best nemo model based on SUMM ROUGE-L score is "conformer-transducer-xxlarge".

System	Speech model	Text model	ROUGE-1	ROUGE-2	ROUGE-L	METEOR	BERTScore	WER
pipeline-oracle	x	LED	29.4	7.2	18.9	13.3	83.7	0.0
pipeline-wv2v2	W2V2-ASR	LED	26.7	5.5	17.0	12.2	82.6	34.5
pipeline-hubert	Hubert-ASR	LED	26.6	5.3	16.6	12.3	82.5	30.2
pipeline-nemo	best model*	LED	27.4	5.8	17.3	12.7	82.6	25.5
pipeline-whisper	whisper-en	LED	29.1	7.2	18.8	13.1	83.7	11.0
pipeline-whisper ASR	whisper-en	LED(ASR)	29.1	7.3	18.9	13.3	83.7	11.0
E2E-TED3	TEDLIUM3-Conformer	x	23.9	5.2	16.3	10.4	84.3	—



(a) Transcript length distribution



(b) Audio duration distribution

Figure 9: Figure showing transcript length and audio duration distribution in TED Summary dataset

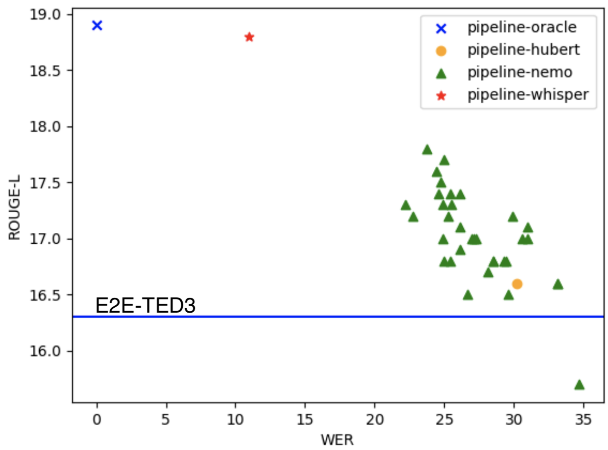


Figure 10: SUMM task : WER and ROUGE-L score on dev set

Table 15: Matches in predicted summary and reference summary for different POS tags

POS Tag	Matches(%)
PROPN	6.1
AUX	42.5
ADJ	10.8
CCONJ	55.1
ADV	9.7
VERB	11.3
PRON	34.3
NOUN	19.7
DET	82.5

Table 16: SLUE-TED Summarization examples.
.

Method	Example
Reference	The work that makes all other work possible [SEP] Domestic workers are entrusted with the most precious aspects of people's lives – they're the nannies, the elder-care workers and the house cleaners who do the work that makes all other work possible. Too often, they're invisible, taken for granted or dismissed as "help"; yet they continue to do their wholehearted best for the families and homes in their charge. In this sensational talk, activist Ai-Jen Poo shares her efforts to secure equal rights and fair wages for domestic workers and explains how we can all be inspired by them. "Think like a domestic worker who shows up and cares no matter what" she says.
pipeline-hubert	The domestic workers' rights movement [SEP] In the US, domestic workers are often characterized as unskilled, unskilled and largely uneducated – a legacy that's often cast aside for more humane work. But in this bold, human talk, Ameera Al-Sabouni advocates for a new kind of work, one that includes days of rest, paid time off and other protections for domestic workers – and shares how the movement for domestic workers' rights is gaining legislative momentum.
E2E-TED3	The work that makes all other work possible? [SEP] What makes all other work possible? In this world, it's possible, says important immorality domestic workers are so fundamental to the very basics of our lives, says lawyer and lawyer and TED Fellow Juan Enriquez. She tells the story of how workplaces that makes all other work possible.
Reference	The link between fishing cats and mangrove forest conservation [SEP] Mangrove forests are crucial to the health of the planet, gobbling up CO2 from the atmosphere and providing a home for a diverse array of species. But these rich habitats are under continual threat from deforestation and industry. In an empowering talk, conservationist and TED Fellow Ashwin Naidu shares how community-driven efforts in South and Southeast Asia are working to protect mangroves – all with a little help from the mysterious and endangered fishing cat.
pipeline-hubert	Why protecting forests is the best thing for the environment [SEP] protecting one acre of rainforests in south east asia may well be like protecting five or more acres of tropical forests in the future. But would you like to eliminate your entire life's carbon footprint? Eco-entrepreneur and TED fellow Sophia Kianni considers that action is being taken to protect these precious ecosystems – and the millions of people who live next to them. "Mangroves are more than just their home to a fast-growing ecosystem" she says. "They can be the first line of defense between storm surges, tsunamis and the millions of people who live next to these forests for their survival."
E2E-TED3	The tigers of the Mangroves [SEP] We can all be part of a future where fishing cats are threatened by habitat loss, loves to fish and lives in some of the most unique and valuable ecosystems on earth, mainly because of our international deforestations, local people and the global community. So what's learned that we can all be part of a future where fishing cats are threatened by habitat loss, local people and the global community. In this eye-opening talk, she shares how these restored Mangroves may be lost.

D Named entity localization

D.1 Annotation details

As described in Sec. 3.2.4, we use MFA to obtain ground-truth word-level alignments. When we run MFA, it fails to align twenty-six files across dev and test splits. On manual inspection we identify differences in audio utterance and the corresponding text transcript due to incorrect end-pointing for twenty-two of these files. These cases have contiguous words at the end of the transcript that are not a part of the audio utterance. Running MFA after removing these extra words from the transcripts fixes these cases. But, for seven of these files, at least one entity word is a part of the missing words and so, the time alignments don't have all the entity phrases that are a part of the published SLUE-NER annotations. In the interest of utterance-level consistency between SLUE-NER and SLUE-NEL, we skip these files. For the remainder four of the twenty-six files that MFA fails to align, we manually add the word alignments using Praat software (Boersma and Weenink, 2009).

In order to check the validity of MFA produced alignments, we manually verify the entity alignments for 372 entity phrases across randomly chosen 188 utterances in dev split. This constitutes 20% of all entity phrases in the dev split and thus our analysis should be representative for the complete split. Our manual pass exposed 51 of 372 phrases to be misaligned and the nature of misalignment varied from a minor offset to being completely off. In order to quantify the effect of the identified misalignments on our evaluation metrics, we manually rectify the alignments for these 51 phrases and report the following scores for this representative set of 188 utterances:

1. The frame-F1 between rectified and original timestamps is 96%,
2. The relative difference in baseline model scores (evaluating models listed in Table 8) using these two versions as ground-truths is <3%,
3. The general trend in baseline model scores is similar across models for the results using these two versions as ground-truths.

Thus, we conclude that the alignments produced by MFA are reliable for robustly comparing between different modeling approaches and can be used as ground-truth despite minor issues in the generated time-stamps. Additionally, we find that the faulty timestamps are a result of imperfect transcripts in VoxPopuli and not an issue with MFA. The imperfections in these transcripts are expected, since the data is originally curated with 20% character error rate threshold (Wang et al., 2021).

D.2 Hyperparameter details

NEL evaluation has two hyperparameters,*o*ffset and *incl_blank*. We evaluate the dev set on a range of offset values between -0.3 seconds and 0.3 seconds with an increment of 20 milliseconds. The *incl_blank* is a Boolean hyperparameter. The best hyperparameter values based on dev set performance are listed in Table 17.

The 34 NeMo models have one of the three types of decoding strategies – (i) character-level CTC, (ii) subword-level CTC, and (iii) subword-level RNN transducer (RNNT). The character-level CTC models are processed in the same way as the *pipeline-w2v2* models, where the *incl_blank* denotes whether or not the ϵ tokens before and after the entity phrase, between the word separator tokens, are included in the entity time stamp. The subword-level CTC model vocabulary in the NeMo toolkit does not have a word separator token, and instead, the start of the word is characterized by an "_" prepended to a subword. So, the *incl_blank* denotes whether the trailing ϵ tokens, before the start of the next word, are included in the entity time stamp. The RNNT model class in the NeMo toolkit directly gives subword-level start times, so *offset* was the only relevant hyperparameter here.

D.3 Error analysis

Table 18 shows precision and recall values for the NEL models. The E2E model outperforms in *precision* (i.e, more predicted regions are named entities), whereas the pipeline model outperforms in *recall*. The mismatch in text NER's training (ground-truth text) and inference (ASR output) could lead to higher false positives in the pipeline model.

Table 17: Best hyperparameters for NEL models

System	Speech model	Training objective	offset (s)	incl_blank
E2E-w2v2	wav2vec2	char-CTC	0.00	True
pipeline-w2v2	wav2vec2	char-CTC	-0.08	True
pipeline-nemo	QuartzNet15x5Base-En	char-CTC	-0.22	True
	stt_en_jasper10x5dr		-0.26	True
	stt_en_quartznet15x5		-0.26	True
pipeline-nemo	stt_en_citrinet_1024	subword-CTC	-0.10	True
	stt_en_citrinet_1024_gamma_0_25		-0.10	True
	stt_en_citrinet_256		-0.10	True
	stt_en_citrinet_256_gamma_0_25		0.00	True
	stt_en_citrinet_512		-0.12	True
	stt_en_citrinet_512_gamma_0_25		-0.16	True
pipeline-nemo	stt_en_conformer_ctc_large	subword-CTC	-0.12	True
	stt_en_conformer_ctc_large_ls		-0.02	False
	stt_en_conformer_ctc_medium		-0.12	True
	stt_en_conformer_ctc_medium_ls		-0.02	False
	stt_en_conformer_ctc_small		-0.08	True
	stt_en_conformer_ctc_small_ls		0.00	False
	stt_en_conformer_ctc_xlarge		-0.08	True
pipeline-nemo	stt_en_squeezeformer_ctc_large_ls	subword-CTC	-0.02	False
	stt_en_squeezeformer_ctc_medium_large_ls		-0.02	False
	stt_en_squeezeformer_ctc_medium_ls		-0.02	False
	stt_en_squeezeformer_ctc_small_ls		-0.02	False
	stt_en_squeezeformer_ctc_small_medium_ls		-0.02	False
	stt_en_squeezeformer_ctc_xsmall_ls		-0.02	False
pipeline-nemo	stt_en_conformer_transducer_large	subword-RNNT	0.16	n/a
	stt_en_conformer_transducer_large_ls		0.14	n/a
	stt_en_conformer_transducer_medium		0.20	n/a
	stt_en_conformer_transducer_small		0.20	n/a
	stt_en_conformer_transducer_xlarge		0.18	n/a
	stt_en_conformer_transducer_xxlarge		0.18	n/a
pipeline-nemo	stt_en_contextnet_1024	subword-RNNT	0.22	n/a
	stt_en_contextnet_1024_mls		0.30	n/a
	stt_en_contextnet_256		0.14	n/a
	stt_en_contextnet_256_mls		0.20	n/a
	stt_en_contextnet_512		0.22	n/a
	stt_en_contextnet_512_mls		0.30	n/a

Figure 12 shows the scatter plot between WER and F1 scores for NeMo, where the points are color-coded for different base model types. We see that the NEL and ASR performance are correlated within a single model category.

Table 18: NEL task baseline precision and recall performance on dev set. *the best nemo model based on NEL frame-f1 score on dev is "stt_en_conformer_ctc_small"

System	Speech model	Text model	frame-F1		word-F1 (ρ=1)		word-F1 (ρ=0.8)		word-F1 (ρ=0.5)	
			Prec.	Recall	Prec.	Recall	Prec.	Recall	Prec.	Recall
pipeline-oracle	x	DeBERTa	91.7	92.8	92.4	94.7	92.4	94.7	92.4	94.7
pipeline-w2v2	wav2vec2	DeBERTa	57.8	78.8	70.4	46.4	71.1	74.1	68.5	84.9
E2E-w2v2	wav2vec2	x	81.0	51.7	71.8	19.5	83.8	55.0	83.2	63.2
pipeline-nemo	best model*	DeBERTa	69.2	83.2	82.4	56.4	83.7	83.1	79.7	88.1

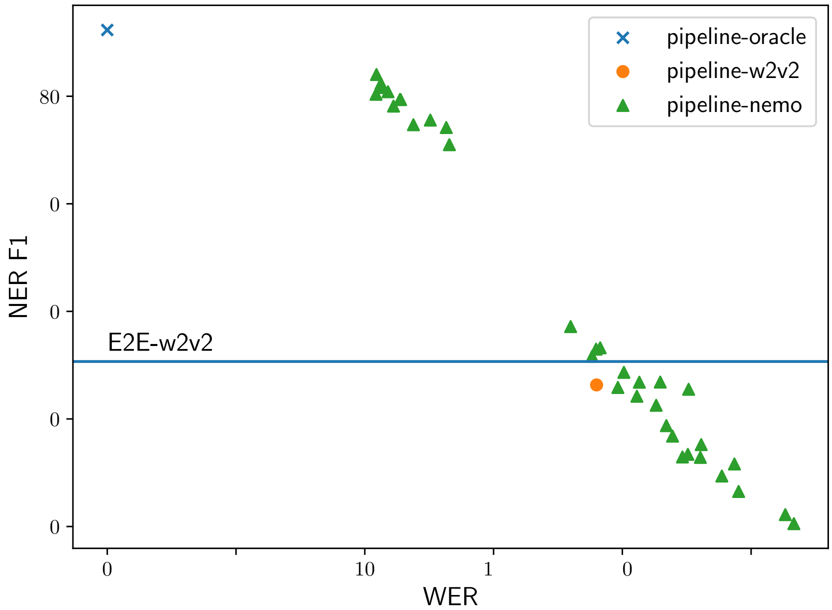


Figure 11: NER task: WER and frame-F1 scores on dev set

D.4 Additional results

Table 19 shows performance of NEL for dev and test sets across different thresholds for word-F1. For word-F1, relaxing the tolerance from $\rho = 1$ to $\rho = 0.8$ gives a major performance boost – up to 30% and 116% relative for pipeline and E2E models respectively.

Table 19: NEL task baseline performance. The wav2vec2 models are fine-tuned on slue-voxpopuli data.*the best NeMo model based on NEL frame-f1 score on dev is "stt_en_conformer_ctc_small"

System	Speech model	Text model	frame-F1		word-F1 (ρ=1)		word-F1 (ρ=0.8)		word-F1 (ρ=0.5)	
			Dev	Test	Dev	Test	Dev	Test	Dev	Test
pipeline-oracle	x	DeBERTa	92.3	89.0	93.6	90.0	93.6	90.0	93.6	90.0
pipeline-w2v2	wav2vec2	DeBERTa	66.9	65.1	56.0	53.6	72.7	72.1	75.9	74.1
E2E-w2v2	wav2vec2	x	63.2	56.2	30.8	25.7	66.5	59.4	71.8	64.6
pipeline-nemo	best model*	DeBERTa	75.5	74.1	66.9	64.0	83.4	81.4	83.7	81.0

Figure 13 shows the correlation between WER and frame-F1 on dev set. It follows a similar trend to test set (see Figure 1d).

E Experiment detail

Table 20 shows NeMo model name list used in the experiment. Table 21 shows the number of parameters for model used in the experiment.

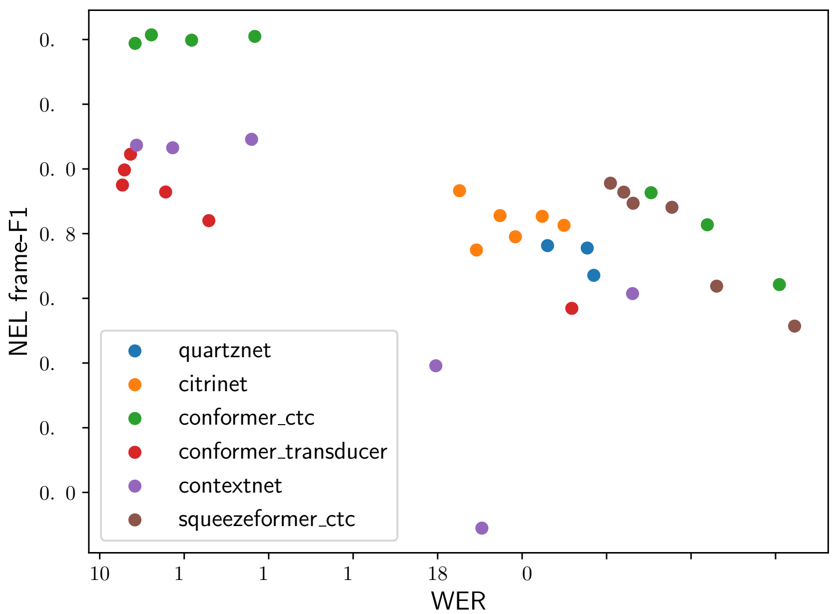


Figure 12: WER and frame-F1 scores on test set for different NeMo models

Table 20: NeMo model list used in the experiment

NeMo model	DAC	QA	SUMM	NEL
QuartzNet15x5Base-En	o	o	o	o
stt_en_citrinet_1024	o	o	o	o
stt_en_citrinet_1024_gamma_0_25	o	o	o	o
stt_en_citrinet_256	o	o	o	o
stt_en_citrinet_256_gamma_0_25	o	o	o	o
stt_en_citrinet_512	o	o	o	o
stt_en_citrinet_512_gamma_0_25	o	o	o	o
stt_en_conformer_ctc_large	o	o	o	o
stt_en_conformer_ctc_large_ls	o	o	o	o
stt_en_conformer_ctc_medium	o	o	o	o
stt_en_conformer_ctc_medium_ls	o	o	o	o
stt_en_conformer_ctc_small	o	o	o	o
stt_en_conformer_ctc_small_ls	o	o	o	o
stt_en_conformer_ctc_xlarge	o	o	o	o
stt_en_conformer_transducer_large	o	o	o	o
stt_en_conformer_transducer_large_ls	o	o	o	o
stt_en_conformer_transducer_medium	o	o	o	o
stt_en_conformer_transducer_small	o	o	o	o
stt_en_conformer_transducer_xlarge	o	o	o	o
stt_en_conformer_transducer_xxlarge	o	o	o	o
stt_en_contextnet_1024	o	o	o	o
stt_en_contextnet_1024_mls	o	o	o	o
stt_en_contextnet_256	o	o	o	o
stt_en_contextnet_256_mls	o	o	o	o
stt_en_contextnet_512	o	o	o	o
stt_en_contextnet_512_mls	o	o	o	o
stt_en_jasper10x5dr	o	o	o	o
stt_en_quartznet15x5	o	o	o	o
stt_en_squeezeformer_ctc_large_ls	o	o	o	o
stt_en_squeezeformer_ctc_medium_large_ls	o	o	o	o
stt_en_squeezeformer_ctc_medium_ls	o	o	o	o
stt_en_squeezeformer_ctc_small_ls	o	o	o	o
stt_en_squeezeformer_ctc_small_medium_ls	o	o	o	o
stt_en_squeezeformer_ctc_xsmall_ls	o	o	o	o

Table 21: Model parameter size used in experiment. We use *base* sized model when there are multiple variants of the pre-trained model except off-the-shelf ASR model

Type	model name	parameter size
Speech model	wav2vec2	95M
	DUAL (k-means model and Longformer part)	149M
	TEDLIUM3-Conformer	48.8M
	Hubert-ASR (Conformer part excluding Hubert)	49.1M
	W2V2-ASR (Conformer part excluding wav2vec2)	49.1M
Text model	DeBERTa	139M
off-the-shelf ASR model	Whisper-en	71M
	QuartzNet15x5Base-En	18M
	stt_en_citrinet_1024	143M
	stt_en_citrinet_1024_gamma_0_25	141M
	stt_en_citrinet_256	10M
	stt_en_citrinet_256_gamma_0_25	9M
	stt_en_citrinet_512	36M
	stt_en_citrinet_512_gamma_0_25	36M
	stt_en_conformer_ctc_large	121M
	stt_en_conformer_ctc_large_ls	121M
	stt_en_conformer_ctc_medium	30M
	stt_en_conformer_ctc_medium_ls	30M
	stt_en_conformer_ctc_small	13M
	stt_en_conformer_ctc_small_ls	12M
	stt_en_conformer_ctc_xlarge	635M
	stt_en_conformer_transducer_large	120M
	stt_en_conformer_transducer_large_ls	120M
	stt_en_conformer_transducer_medium	32M
	stt_en_conformer_transducer_small	14M
	stt_en_conformer_transducer_xlarge	644M
	stt_en_conformer_transducer_xxlarge	998M
	stt_en_contextnet_1024	144M
	stt_en_contextnet_1024_mls	144M
	stt_en_contextnet_256	14M
	stt_en_contextnet_256_mls	14M
	stt_en_contextnet_512	40M
	stt_en_contextnet_512_mls	40M
	stt_en_jasper10x5dr	332M
	stt_en_quartznet15x5	18M
	stt_en_squeezeformer_ctc_large_ls	236M
	stt_en_squeezeformer_ctc_medium_large_ls	125M
	stt_en_squeezeformer_ctc_medium_ls	77M
	stt_en_squeezeformer_ctc_small_ls	18M
	stt_en_squeezeformer_ctc_small_medium_ls	28M
	stt_en_squeezeformer_ctc_xsmall_ls	9M

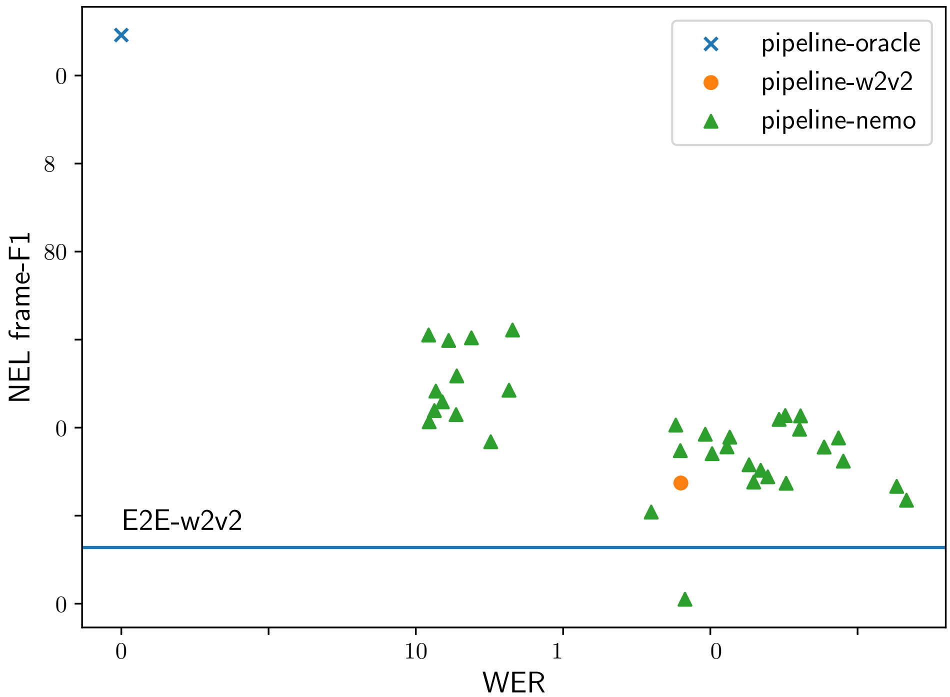


Figure 13: NEL task: WER and frame-F1 scores on dev set

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
limitations section in page 9.

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
in the abstract and section 1.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

section 4

☑ B1. Did you cite the creators of artifacts you used?
section 4.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. We don't create any artifacts for distribution in this submission since a few of authors are prohibited to distribute dataset and source code anonymously. We will publish the dataset and reproducible code upon paper acceptance.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 4

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. We use the open-sourced dataset that already published and added additional annotation. Added annotation only is related to dialog act class and Document-Question pair validation. No new content is added in the original audio or original text dataset.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
We don't create any artifacts for distribution at this point.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
section 3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

in section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
in section 4 and appendix.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
in section 4 and appendix

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
section 4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
section 4.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

described in section 3 and appendix for detail.

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
described at section 3 and more details at appendix.

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
at section 3 and appendix

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
section 3

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Added detail in Broader Impact and Ethics section

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
In appendix.