Computing, Telecommunication and Control, 2025, Vol. 18, No. 2, Pp. 33–44.
Информатика, телекоммуникации и управление. 2025. Том 18, № 2. С. 33–44.

Research article
DOI: https://doi.org/10.18721/JCSTCS.18203
UDC 004.522,004.934

DATASET CREATION FOR COMPREHENSIVE PERFORMANCE EVALUATION OF AUTOMATIC SPEECH RECOGNITION SYSTEMS

A.Yu. Andrusenko ✉ , *P.D. Drobintsev*

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

✉ andrusenkoau@gmail.com

Abstract. The performance evaluation of Automatic Speech Recognition (ASR) systems heavily depends on the availability of diverse and representative test datasets encompassing a wide range of complexities in various domains. This work introduces a novel methodology for collecting and preparing datasets for comprehensive ASR system evaluation. The proposed dataset incorporates a modern vocabulary enriched with numerous unique terms and proper nouns, facilitating an in-depth evaluation of overall ASR performance and the effectiveness of context-biasing techniques in computer science. Additionally, the dataset retains critical text features such as Punctuation and Capitalization (P&C), enabling a rigorous evaluation of P&C prediction algorithms. We present a detailed account of the dataset creation process, along with its statistical and qualitative analysis. Furthermore, we benchmark state-of-the-art ASR models, context-biasing approaches, and P&C prediction techniques using the proposed dataset, providing valuable insights into their relative performance.

Keywords: automatic speech recognition, test dataset, large language models, punctuation and capitalization, context-biasing

Citation: Andrusenko A.Yu., Drobintsev P.D. Dataset creation for comprehensive performance evaluation of automatic speech recognition systems. Computing, Telecommunications and Control, 2025, Vol. 18, No. 2, Pp. 33–44. DOI: 10.18721/JCSTCS.18203

Научная статья
DOI: https://doi.org/10.18721/JCSTCS.18203
УДК 004.522,004.934

СОЗДАНИЕ НАБОРА ДАННЫХ ДЛЯ КОМПЛЕКСНОЙ ОЦЕНКИ ПРОИЗВОДИТЕЛЬНОСТИ СИСТЕМ АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ РЕЧИ

А.Ю. Андрусенко ✉ , *П.Д. Дробинцев*

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

✉ andrusenkoau@gmail.com

Аннотация. Оценка производительности систем автоматического распознавания речи (Automatic Speech Recognition, ASR) в значительной степени зависит от наличия разнообразных и репрезентативных тестовых наборов данных, охватывающих широкий спектр сложностей в различных доменах. В данном исследовании представлена новая методология сбора и подготовки наборов данных, предназначенных для всесторонней оценки ASR систем. Предложенный набор данных включает современный словарный запас, обогащенный многочисленными уникальными терминами и именами собственными, что позволяет проводить углубленную оценку общей производительности ASR и эффективности методов смещения контекста (context-biasing) в области компьютерных технологий. Кроме того, в наборе данных сохраняются важные текстовые характеристики, такие как пунктуация и капитализация (Punctuation & Capitalization, P&C), что делает возможной строгую оценку алгоритмов предсказания P&C. Мы подробно описываем процесс создания набора данных, включая его анализ. Более того, мы проводим тестирование передовых ASR моделей, методов смещения контекста и алгоритмов предсказания P&C на основе предложенного набора данных, предоставляя ценные сведения об их относительной производительности.

Ключевые слова: автоматическое распознавание речи, тестовый набор данных, большие языковые модели, пунктуация и капитализация, смещение контекста

Для цитирования: Andrusenko A.Yu., Drobintsev P.D. Dataset creation for comprehensive performance evaluation of automatic speech recognition systems // Computing, Telecommunications and Control. 2025. Т. 18, № 2. С. 33–44. DOI: 10.18721/JCSTCS.18203

Introduction

The rapid advancements in deep learning techniques have driven the development of numerous end-to-end Automatic Speech Recognition (ASR) systems [1]. A comprehensive evaluation of these models necessitates using test datasets that span a wide range of linguistic and acoustic conditions across diverse domains [2]. While the Word Error Rate (WER) remains the primary metric for assessing overall ASR performance, specific tasks, such as evaluating context-biasing capabilities, are attracting increasing attention. These tasks are designed to measure how effectively an ASR system recognizes domain-specific keywords and terminology [3]. Achieving robust evaluation for context-biasing requires test datasets enriched with novel, domain-specific words, and phrases that may challenge recognition accuracy due to their unfamiliarity.

The most widely used dataset for ASR tasks is LibriSpeech [4], a collection of English audiobook recordings. However, its utility for evaluating high-performance ASR models, such as Whisper [5], is limited due to the dataset's relatively simple data domain. Furthermore, LibriSpeech lacks a substantial number of novel or rare terms, making it unsuitable for evaluating context-biasing capabilities, especially for ASR models trained on extensive datasets.

Other datasets, such as Switchboard [6] and CallHome [7], introduce greater complexity by focusing on conversational telephone speech. Ted-Lium [8], GigaSpeech [9], and People's Speech [10] target ASR evaluation in scenarios resembling YouTube videos and online presentations. Mozilla Common Voice [11] supports other scenarios, featuring dictated, pre-prepared phrases recorded on various personal devices. For more challenging use cases, datasets like AMI [12] and CHiME-5 [13] simulate environments with significant noise, reverberation, and overlapping speakers, presenting additional difficulties for ASR systems. While these datasets allow broader evaluations of model performance, they remain incomplete in their coverage of diverse data domains. Critically, they also lack sufficient quantities of curated keywords and structured lists required for rigorous context-biasing evaluation.

To address the issue of data diversity, the Earnings21/22 [14, 15] public datasets were introduced, featuring earnings calls from nine financial sectors. Alongside the audio data, these datasets include a list of named entities (keywords) designed to facilitate the evaluation of context-biasing techniques. Despite these contributions, the keyword set has notable limitations: it contains many trivial and high-frequency words that most ASR systems already handle effectively, as well as short words (fewer than three characters) that contribute to elevated rates of false acceptance in context-biasing tasks.

Additionally, the dataset lacks segmentation, comprising lengthy audio recordings ranging from five to seventeen minutes. Processing such extended audio sequences can impose significant computational demands on ASR systems, particularly those employing self-attention mechanisms, which often experience out-of-memory issues on GPU hardware during inference.

The ConEC [16] initiative sought to enhance the Earnings21/22 benchmark by segmenting long audio recordings, refining the keyword list, and introducing a publicly available context-biasing solution based on the shallow-fusion decoding approach. However, despite these improvements, the ConEC benchmark remains constrained to a narrow domain focused exclusively on earnings presentations, limiting its applicability for broader ASR evaluation.

This work introduces a novel approach to creating an ASR evaluation dataset, collected from publicly available YouTube channels under a Creative Commons license. The dataset focuses on the modern technology domain, with a particular emphasis on computer science. It features manually annotated transcriptions that preserve Punctuation and Capitalization (P&C), enabling robust evaluation of P&C prediction tasks. Additionally, the dataset includes a diverse set of domain-specific terms, such as product names, making it highly suitable for evaluating context-biasing methods. To further support these evaluations, we also propose a method for generating keyword lists tailored to the context-biasing task.

The dataset preparation process is implemented using open-source tools within the NeMo framework[1]. The preparation pipeline incorporates several key stages: text cleaning and normalization, automated punctuation insertion using a Large Language Model (LLM), segmentation of data through the removal of non-speech segments, and additional filtering based on ASR accuracy thresholds. These steps ensure a high-quality and domain-relevant dataset for ASR evaluation.

We conducted experiments on the proposed dataset using state-of-the-art ASR models from Hugging Face. Our evaluations included assessments of overall ASR performance, the effectiveness of context-biasing techniques using the proposed keyword list, and P&C prediction accuracy. The results provide valuable insights into the capabilities and limitations of the evaluated models within this domain-specific dataset.

Data preparation pipeline

To enhance ASR evaluation benchmarks under modern conditions, we focused on data scenarios relevant to the field of computer science. A prime example of such data is keynote presentations on various technology topics from major tech companies, such as Google, Microsoft, Amazon, and others. Using

[1] GitHub – NVIDIA/NeMo: A scalable generative AI framework built for researchers and developers working on Large Language Models, Multimodal, and Speech AI (Automatic Speech Recognition and Text-to-Speech), Available: https://github.com/NVIDIA/NeMo (Accessed 21.05.2025)

the YouTube-dl library[2], we collected 15 hours of full-length recordings from recent years, capturing content directly from these events.

The collected recordings include manually created transcriptions with preserved P&C, making them valuable for tasks, such as P&C prediction. However, the raw data required extensive preprocessing to ensure its suitability for ASR evaluation benchmarks.

Text preprocessing

Even manually created transcriptions can contain numerous typos and non-standard characters, negatively impacting ASR evaluation. To address this, we applied pattern-based substitutions using regular expressions to correct common errors and remove invalid characters.

Text normalization was performed to convert numerical values and auxiliary symbols into their text representations. This process was implemented using the NeMo Text Processing toolkit[3], which supports both forward and inverse text normalization. The normalization process ensured that only characters from the English alphabet were retained in the processed dataset. Additionally, NeMo Text Processing supports audio-based text normalization, which leverages baseline ASR model outputs to enhance numeral normalization. While this method can improve accuracy, it has the potential to introduce errors in challenging acoustic conditions due to ASR recognition inaccuracies.

For punctuation, we standardized the dataset to include only three primary punctuation marks: periods, commas, and question marks. This simplification ensures consistency while maintaining sufficient information for P&C tasks.

Punctuation reconstruction with LLM

Certain portions of the collected data lacked P&C. To address this, we employed the Llama-3-8B[4] LLM, utilizing a carefully designed prompt. The prompt included standardized instructions: “Your task is to punctuate the input text. You can only use a period, comma, or question mark as punctuation. Add capitalization to the beginning of new sentences.”

To process entire text files, we adopted a chunk-based approach, dividing the text into segments of 250 words per iteration. However, this method introduced a potential issue: chunks could end mid-sentence, leading the LLM to erroneously assign an end-of-sentence punctuation mark (e.g., a period or question mark) to the last word in the chunk. To mitigate this, we extracted only the first n-1 complete sentences from each processed chunk, avoiding disruptions caused by mid-sentence breaks. The subsequent chunk then began at the last valid sentence boundary of the previous segment.

While the LLM effectively added punctuation and restored missing capitalization, it slightly altered the original text. In our evaluation, the WER between the input transcription and the normalized LLM output was approximately 2%. This discrepancy was primarily due to the LLM’s removal of repetitive words typical in spoken language and its attempts to correct typos introduced during manual transcription. These issues suggest potential improvements with prompt refinement.

Table 1 illustrates an example of text correction during punctuation restoration using the Llama-3-8B model.

Segmentation

The original dataset comprises full-length recordings ranging from 1 to 2 hours. Such lengthy inputs pose challenges for ASR systems utilizing global attention mechanisms, as their quadratic complexity with respect to input sequence length can lead to significant computational overhead. Additionally, the presence of prolonged musical segments in the recordings may degrade speech recognition accuracy.

Although the source data includes original timestamps associated with the corresponding text (subtitles), direct segmentation based on these timestamps often results in errors at segment boundaries,

[2] GitHub – ytdl-org/youtube-dl: Command-line program to download videos from YouTube.com and other video sites, Available: https://github.com/ytdl-org/youtube-dl (Accessed 21.05.2025)

[3] GitHub – NVIDIA/NeMo-text-processing: NeMo text processing for ASR and TTS, Available: https://github.com/NVIDIA/NeMo-text-processing (Accessed 21.05.2025)

[4] meta-llama/Meta-Llama-3-8B-Instruct · Hugging Face, Available: https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct (Accessed 21.05.2025)

Table 1

Example of original transcription and Llama-3-8B punctuation reconstruction. Reference typos, LLM corrections, and removed words are highlighted in red and green colors

before	we thought it was a perfect perfect name **black wealth**
after	We thought it was a perfect perfect name, **Blackwell**.
before	..this software and it **compress** it dimensionally **reduce** it..
after	..this software and it **compresses** it, dimensionally **reduces** it..
before	..the new **micr service** now the thing that's that's emerging here..
after	..the new **microservice**. Now, the thing that's emerging here..

thereby increasing the WER of the segmented dataset. Consequently, we utilized these timestamps solely to remove extended non-speech segments at the beginning and end of the audio files.

For more accurate data segmentation, we employed CTC-segmentation [17] using the NeMo toolkit. This method aligns ground truth transcriptions with corresponding audio files effectively. We used Citrinet [18], a convolutional neural network ASR model, for segmentation. Citrinet is particularly well-suited for handling long audio files without encountering GPU memory limitations. Furthermore, its CNN-based architecture ensures more precise alignments, avoiding the late or early prediction errors common in attention-based models.

To streamline the segmentation process, all text data was initially divided into individual sentences based on punctuation. This allowed us to determine sentence boundaries and obtain alignment confidence scores for each segment. However, short sentences often exhibited boundary errors during alignment (Fig. 1, *a*). To mitigate these issues, we merged consecutive sentences if the silence between them was less than one second and their alignment confidence score exceeded −5.0. This approach produced final audio segments with durations ranging from 2 to 40 seconds (Fig. 1, *b*).

This refined segmentation strategy reduced the WER from 12.68% to 11.78% on the processed dataset. However, it also shifted the duration distribution toward the upper limit of 40 seconds, reflecting a bias toward longer segment lengths.

To enhance the diversity of segment lengths, we implemented a probabilistic sentence merging approach. Instead of enforcing mandatory sentence merging up to a predefined length threshold (based on the previously described conditions), we applied a probabilistic mechanism. Specifically, each subsequent sentence was merged with the current segment with a probability of 0.8, provided that the conditions for silence duration and confidence score were satisfied.

The outcomes of this probabilistic sentence merging approach are illustrated in fig. 1c. This method successfully increased the diversity of segment lengths while maintaining comparable ASR performance, as no significant degradation in WER was observed.

Data filtering

Manual analysis of the recognition results revealed that examples with high WER were predominantly caused by segmentation errors or inaccuracies in the reference transcriptions. To address this, we applied a filtering process based on ASR performance metrics.

As a baseline ASR model, we utilized the Fast Conformer-Transducer Large (114M parameters)[5], trained on 20,000 hours of English speech data. We filtered out examples where the WER exceeded 80%, or the Character Error Rate (CER) exceeded 30%. This process resulted in a cleaner, segmented evaluation dataset with a total duration of 12.4 hours.

[5] nvidia/stt_en_fastconformer_ctc_large · Hugging Face, Available: https://huggingface.co/nvidia/stt_en_fastconformer_ctc_large (Accessed 15.06.2024)

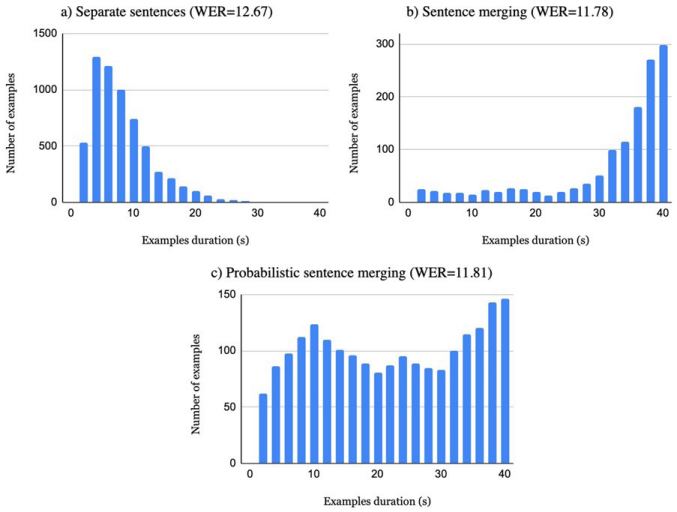

Fig. 1. Duration distribution of segmented data according to the different merge methods: separate sentences, sentence merging, and probabilistic sentence merging. WER of the data sets obtained by considered segmentation methods is 12.67%, 11.78%, and 11.81%, respectively

To prepare for the evaluation, we divided the obtained dataset into two subsets: a 4-hour development (dev) set and an 8.4-hour test set, ensuring non-overlapping talks between the two subsets. All subsequent evaluation results are reported exclusively for the test set.

Named entities (keywords)

The proposed dataset includes a substantial number of named entities suitable for context-biasing tasks. To analyze Named Entity Recognition (NER) statistics, we used SpaCy[6], following a similar methodology to previous works. SpaCy assigns entity tags to words based on predefined classes, such as ORG (organization), PERSON (person), DATE, and CARDINAL (numbers). The proposed dataset contains a significant number of examples for these tags (e.g., ORG=3,534, CARDINAL=1,457, PERSON=846, DATE=987, etc.).

However, SpaCy's tagging process introduces challenges, including overlapping classifications (e.g., the word "AI" being tagged as both ORG and PERSON) and classification errors. Additionally, most words in this entity list achieve high recognition accuracy, when evaluated using the baseline ASR model, making them less relevant for assessing context-biasing performance. For our analysis, we focused on identifying words with low recognition accuracy.

To construct a more appropriate context-biasing keyword list, we applied the following methodology:

1. ASR Evaluation: The dataset was transcribed using the baseline ASR model, and recognition accuracy was calculated for individual words (monograms) and phrases (bigrams).
2. Entity Filtering: Only words present in the named entities identified by SpaCy were retained.
3. Error Word Identification: We observed that most misrecognized phrases contained a single error word already represented in the monogram statistics. Consequently, we prioritized individual words over bigrams, selecting only a limited number of bigrams.
4. Short Word Exclusion: Words shorter than three characters were excluded, as these often contribute to high false acceptance rates during context-biasing recognition.

This process resulted in a refined list of 200 keywords with low recognition accuracy, suitable for evaluating context-biasing techniques. Additionally, we incorporated 800 distractor words – terms likely absent from the dataset – sourced from the Earnings benchmark. This combination allows for a more rigorous evaluation of context-biasing performance.

[6] spaCy · Industrial-strength Natural Language Processing in Python, Available: https://spacy.io (Accessed 21.05.2025)

Experimental setup

Speech recognition

To assess speech recognition accuracy, measured by WER, on the obtained dataset, we evaluated a selection of top-performing public models listed on the Hugging Face ASR Leaderboard[7]. This leaderboard ranks ASR models based on their average WER across multiple public test sets and includes metrics for inference speed. At the time of evaluation, the leading models included those from the NeMo toolkit (e.g., Conformer, Fast-Conformer, Parakeet, and Canary) and OpenAI (Whisper-large-v1/v2/v3).

To ensure fair comparison across models, we applied consistent normalization to all recognition outputs, following the data preparation procedure. This included expanding numerical symbols into their textual representations, removing punctuation, and converting all text to lowercase.

Punctuation and capitalization

To evaluate the capabilities of ASR models in P&C prediction, we selected public models that inherently support P&C functionality. From the NeMo toolkit, we chose the three highest-performing models with P&C capabilities at the time: Fast-Conformer Hybrid with P&C (operating in Transducer decoding mode), Parakeet-tdt_ctc-1.1b, and Canary-1b. Similarly, we selected the top three models from OpenAI's Whisper series (Whisper-large-v1/v2/v3). All these models are available on the Hugging Face platform.

To measure P&C performance, we used two key metrics:

- WER C – Word Error Rate calculated with capitalization preserved in the text.
- PER – Punctuation Error Rate, focusing exclusively on punctuation errors:

$$\mathrm{PER} = \frac{\mathrm{I}+\mathrm{D}+\mathrm{S}}{\mathrm{I}+\mathrm{D}+\mathrm{S}+\mathrm{C}}, \tag{1}$$

where I, D, S, and C are the number of insertions, deletions, substitutions, and correct punctuation predictions during a backtrace matrix calculation. More details about WER C and PER metrics can be found in [19].

Context-biasing

To assess context-biasing performance, we explored the available techniques from the NeMo toolkit using a Hybrid Transducer-CTC model[8]. Notably, this model was not trained on data from the computer science domain, ensuring a fair evaluation of context-biasing methods.

One method to enhance keyword recognition accuracy is word boosting, supported via pyctcdecode[9]. This technique employs a shallow fusion approach during the CTC beam-search decoding process. We applied the default parameters, setting the keyword boosting weight (hotword_weight = 10) and beam size (beam_size = 5).

Another approach is the fast context-biasing method using the CTC-based Word Spotter (CTC-WS) [19]. This method decouples keyword recognition from the broader recognition process by leveraging fast decoding of CTC logits based on a graph constructed exclusively from context-specific keywords. The Word Spotter identifies the desired keywords along with their time intervals and confidence scores, which are subsequently merged with the results from greedy decoding. When used with a Hybrid CTC-Transducer model, this technique enables keyword boosting within Transducer predictions. For this evaluation, we applied the default boosting parameters.

---

[7] Open ASR Leaderboard – a Hugging Face Space by hf-audio, Available: https://huggingface.co/spaces/hf-audio/open_asr_leaderboard (Accessed 21.05.2025)

[8] STT En FastConformer Hybrid Transducer-CTC Large Streaming Multi | NVIDIA NGC, Available: https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_fastconformer_hybrid_large_streaming_multi (Accessed 21.05.2025)

[9] GitHub – kensho-technologies/pyctcdecode: A fast and lightweight python-based CTC beam search decoder for speech recognition, Available: https://github.com/kensho-technologies/pyctcdecode (Accessed 21.05.2025)

As metrics for context-biasing evaluation, we used the standard WER for the entire text and F-score for words from the context-biasing list:

$$F_{score} = 2\frac{Precision \cdot Recall}{Precision + Recall}. \quad (2)$$

In addition, we measured the speed performance of the considered methods excluding the encoder work time. All runtime measurements are averaged over 10 runs.

Experimental results

Speech recognition

Table 1 presents the WER performance of the previously discussed ASR models on both the Hugging Face Leaderboard (HF LB) test sets and the proposed dataset (first two columns). The results indicate that the top-performing models on the HF LB also exhibit strong performance on the proposed test set, achieving WER values of approximately 6–7%. This high accuracy may be attributed to the extensive training data utilized by these models, which likely includes diverse sources such as YouTube content.

The best WER on the proposed dataset set was achieved by the Canary-1b model, which was also the top-ranked model on the HF LB at the time of evaluation.

Table 2

Performance results (%) of public ASR models from Hugging Face

ASR model	Size, B	HF LB, WER	Proposed dataset		
			WER	WER C	PER
C_transducer_L	0.12	10.2	15.4	–	–
FC_transducer_L	0.12	9.8	12.1	–	–
FC_hybrid_L_pc	0.12	–	9.4	13.3	37.5
Parakeet-tdt_ctc	1.10	8.1	6.7	10.1	25.4
Canary-1b	1.00	7.7	6.4	9.7	26.3
Whisper-large-v1	1.55	10.4	6.5	10.3	27.1
Whisper-large-v2	1.55	9.0	6.8	9.6	27.4
Whisper-large-v3	1.55	8.6	6.6	9.5	27.5

Note: HF LB WER is the average WER for other test sets in the HF LB; WER C is the WER with capitalization left in the text; PER is Punctuation Error Rate

However, smaller models (about 120M parameters) based on Conformer and Fast-Conformer architectures with only public training datasets (20k+ hours) showed WER above 10% for the proposed test set. This fact confirms the absence of the proposed computer science domain in public datasets.

Punctuation and capitalization

The results for P&C prediction are presented in the rightmost columns of Table 2, which include metrics for WER C and PER. While these metrics correlate with the standard WER, discrepancies can arise in specific cases. For instance, when comparing the Parakeet and Canary models, a model with a better WER may perform worse in terms of PER. This highlights the importance of evaluating punctuation and capitalization effectiveness independently, enabled by datasets that preserve P&C information, in addition to standard transcription accuracy.

An ablation study on punctuation prediction is illustrated in Fig. 2. This analysis examines the prediction error rates for individual punctuation marks: periods, commas, and question marks across the

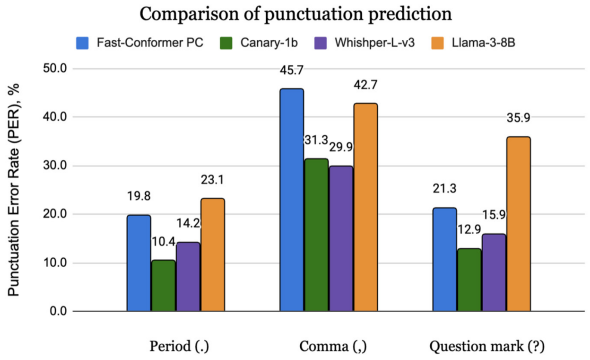

Fig. 2. Comparison of punctuation prediction (PER) for three considered punctuation classes over the proposed test set except samples with Llama-3-8B punctuation. The overall PER is presented in the legend

three evaluated ASR models. Additionally, the study includes a comparison with punctuation predictions generated by the Llama-3 LLM, which was employed during the dataset preparation process. To ensure fairness in comparison, test set examples containing punctuation generated by the LLM were excluded from the evaluation.

The results demonstrate that commas have the lowest prediction accuracy compared to periods and question marks, which are recognized with relatively high accuracy. This finding highlights the inherent difficulty of predicting commas in ASR systems. As a potential improvement, it may be preferable to omit comma prediction and focus solely on sentence-end punctuation, such as periods and question marks.

Punctuation accuracy achieved by the LLM model was lower than that of the top-performing ASR models. This discrepancy underscores the advantage of leveraging audio cues, which significantly enhance punctuation accuracy, particularly for question marks. Nevertheless, the LLM performed reasonably well when relying solely on textual input, making it a viable option for simplifying punctuation tasks in long audio files. Using ASR models for punctuation in such scenarios can be computationally intensive, as it requires chunk-wise decoding, alignment of ASR outputs with the original text, and the subsequent merging of results. The LLM offers a simpler alternative for such use cases.

An additional analysis investigated how the number of sentences within test examples affects the accuracy of end-of-sentence punctuation predictions. For single-sentence test examples, this task is relatively straightforward, as the model can predict sentence-end punctuation with high confidence at the end of the input. However, for test examples containing multiple sentences, the task becomes increasingly complex.

To quantify this effect, we measured the PER for sentence-end labels (periods and question marks) across all test examples. We then grouped the results based on the number of sentences in the reference transcriptions and averaged the PER for each group. The findings, presented in Fig. 3, confirm the hypothesis: as the number of sentences in test examples increases, the accuracy of sentence-end punctuation predictions decreases. This observation supports the utility of grouping multiple sentences into single test examples to increase the overall complexity of punctuation evaluation tasks.

Context-biasing

The results of the context-biasing evaluation for the proposed dataset are summarized in Table 3. Both context-biasing methods demonstrated improvements in recognition accuracy. However, pyctcdecode exhibited relatively poor performance compared to the CTC-WS method.

One significant limitation of pyctcdecode is its sensitivity to the size of the context-biasing list. Due to a marked degradation in processing speed with larger lists, we were constrained to use a list of 200 words instead of the initially intended 1000 words. This limitation aligns with observations from prior

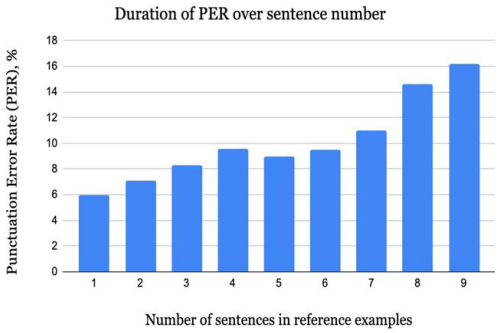

Fig. 3. PER distribution for the period and the question mark over a number of sentences in reference examples in the proposed test set. Canary-1b was used as an ASR model

work, which similarly reported performance issues when scaling the context-biasing list size in pyctcdecode.

The CTC-WS method, when applied with the proposed context-biasing list, improved recognition accuracy for both CTC and Transducer decodings by over 4%, with minimal additional decoding time overhead. These results highlight the abundance of keywords in the proposed dataset, making it highly suitable for evaluating context-biasing tasks, particularly in scenarios involving novel domains.

Table 3

CTC and Transducer decoding results for the proposed test set

Method	CB	Time, s	F-score (P/R)	WER, %
CTC				
greedy	no	4	0.36 (0.96/0.21)	16.44
pyctcdecode	no	21	0.37 (0.97/0.23)	16.57
	yes	1498	0.66 (0.87/0.54)	15.41
CTC-WS	yes	31	0.82 (0.82/0.82)	12.07
Transducer				
greedy	no	15	0.41 (0.97/0.26)	15.89
CTC-WS	yes	44	0.82 (0.82/0.82)	11.69

Note: CB is the presence of context-biasing; P is Precision; R is Recall.

Overall assessment of the proposed methodology

Based on the conducted analysis of the ASR systems evaluation, we can draw a conclusion about the effectiveness of the proposed methodology. For example, the use of LLM allows us to arrange and evaluate the accuracy of P&C recognition, which is also important when segmenting long audios by sentences. Probabilistic sentence merging allows us to simultaneously reduce the number of segmentation errors at the edges of segments and make examples with several sentences that improve P&C evaluation. The choice of keywords allows us to test various context-biasing methods, which are extremely important at this time. The proposed methodology for collecting and processing data allows us to obtain a versatile high-quality test set for broad performance evaluation of modern ASR systems in three main areas.

Conclusion

This study introduced a novel methodology for preparing evaluation datasets tailored to the computer science domain. The resulting dataset features a rich set of domain-specific terms and retains

punctuation and capitalization (P&C), enabling its use for comprehensive ASR model evaluation. It supports a broad range of tasks, including standard speech recognition (WER), context-biasing, and P&C prediction scenarios.

We provided a detailed description of the data preparation pipeline, utilizing open-source frameworks to ensure reproducibility and accessibility. Furthermore, we evaluated state-of-the-art public ASR models across the three primary use cases and conducted ablation studies on punctuation prediction using the proposed dataset. This work offers a robust resource and valuable insights for advancing ASR evaluation benchmarks.

REFERENCES

1. **Prabhavalkar R., Hori T., Sainath T.N., Schlüter R., Watanabe S.** End-to-end speech recognition: A survey. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024, Vol. 32, Pp. 325–351. DOI: 10.1109/TASLP.2023.3328283

2. **Nguyen T.S., Stüker S., Waibel A.** Toward cross-domain speech recognition with end-to-end models. *arXiv:2003.04194*, 2020. DOI: 10.48550/arXiv.2003.04194

3. **Pundak G., Sainath T.N., Prabhavalkar R., Kannan A., Zhao D.** Deep context: End-to-end contextual speech recognition. *arXiv:1808.02480*, 2018. DOI: 10.48550/arXiv.1808.02480

4. **Panayotov V., Chen G., Povey D., Khudanpur S.** Librispeech: An ASR corpus based on public domain audio books. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), 2015, Pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964

5. **Radford A., Kim J.W., Xu T., Brockman G., McLeavey C., Sutskever I.** Robust speech recognition via large-scale weak supervision. *arXiv:2212.04356*, 2022. DOI: 10.48550/arXiv.2212.04356

6. **Godfrey J.J., Holliman E., McDaniel J.** SWITCHBOARD: telephone speech corpus for research and development. *ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1992, Vol. 1, Pp. 517–520. DOI: 10.1109/ICASSP.1992.225858

7. **Cieri C., Miller D., Walker K.** The fisher corpus: A resource for the next generations of speech-to-text. *Proceedings of the Fourth International Conference on Language Resources and Evaluation* (*LREC'04*), 2004.

8. **Rousseau A., Deléglise P., Estève Y.** TED-LIUM: an automatic speech recognition dedicated corpus. *Proceedings of the Eighth International Conference on Language Resources and Evaluation* (*LREC'12*), 2012, Pp. 125–129.

9. **Chen G., Chai S., Wang G., Du J., Zhang W.-Q., Weng C., Su D., Povey D., Trmal J., Zhang J., Jin M., Khudanpur S., Watanabe S., Zhao S., Zou W., Li X., Yao X., Wang Y., You Z., Yan Z.** GigaSpeech: An evolving, multi-domain ASR corpus with 10,000 hours of transcribed audio. *INTERSPEECH 2021*, 2021, Pp. 3670–3674. DOI: 10.21437/Interspeech.2021-1965

10. **Galvez D., Diamos G., Ciro J., Cerón J.F., Achorn K., Gopi A., Kanter D., Lam M., Mazumder M., Reddi V.J.** The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. *35th Conference on Neural Information Processing Systems* (*NeurIPS 2021*), 2021, Pp. 1–12.

11. **Ardila R., Branson M., Davis K., Henretty M., Kohler M., Meyer J., Morais R., Saunders L., Tyers F.M., Weber G.** Common voice: A massively-multilingual speech corpus. *arXiv:1912.06670*, 2019. DOI: 10.48550/arXiv.1912.06670

12. **Carletta J., Ashby S., Bourban S. et al.** The AMI meeting corpus: A pre-announcement. *Machine Learning for Multimodal Interaction* (*MLMI 2005*), 2005, Vol. 3869, Pp. 28–39. DOI: 10.1007/11677482_3

13. **Barker J., Watanabe S., Vincent E., Trmal J.** The fifth 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines. *INTERSPEECH 2018 – 19th Annual Conference of the International Speech Communication Association*, 2018.

14. **Del Rio M., Delworth N., Westerman R., Huang M., Bhandari N., Palakapilly J., McNamara Q., Dong J., Żelasko P., Jetté M.** Earnings-21: A practical benchmark for ASR in the wild. *INTERSPEECH 2021*, Pp. 3465–3469. DOI: 10.21437/Interspeech.2021-1915

15. **Del Rio M., Ha P., McNamara Q., Miller C., Chandra S.** Earnings-22: A practical benchmark for accents in the wild. *arXiv:2203.15591*, 2022. DOI: 10.48550/arXiv.2203.15591

16. **Huang R., Yarmohammadi M., Trmal J., Liu J., Raj D., Garcia L.P., Ivanov A.V., Ehlen P., Yu M., Povey D., Khudanpur S.** ConEC: Earnings call dataset with real-world contexts for benchmarking contextual speech recognition. *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation* (*LREC-COLING 2024*), 2024, Pp. 3700–3706.

17. **Kürzinger L., Winkelbauer D., Li L., Watzel T., Rigoll G.** CTC-segmentation of large corpora for German end-to-end speech recognition. *Speech and Computer*, 2020, Pp. 267–278. DOI: 10.1007/978-3-030-60276-5_27

18. **Majumdar S., Balam J., Hrinchuk O., Lavrukhin V., Noroozi V., Ginsburg B.** Citrinet: Closing the gap between non-autoregressive and autoregressive end-to-end models for automatic speech recognition. *arXiv:2104.01721*, 2021. DOI: 10.48550/arXiv.2104.01721

19. **Meister A., Novikov M., Karpov N., Bakhturina E., Lavrukhin V., Ginsburg B.** LibriSpeech-PC: Benchmark for evaluation of punctuation and capitalization capabilities of end-to-end ASR models. *arXiv:2310.02943*, 2023. DOI: 10.48550/arXiv.2310.02943

20. **Andrusenko A., Laptev A., Bataev V., Lavrukhin V., Ginsburg B.** Fast context-biasing for CTC and transducer ASR models with CTC-based word spotter. *INTERSPEECH 2024*, 2024, Pp. 757–761. DOI: 10.21437/Interspeech.2024-1002

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Andrusenko Andrei Yu.
Андрусенко Андрей Юрьевич
E-mail: andrusenkoau@gmail.com

Drobintsev Pavel D.
Дробинцев Павел Дмитриевич
E-mail: drob@ics2.ecd.spbstu.ru
ORCID: https://orcid.org/0000-0003-1116-7765

Submitted: 18.12.2024; Approved: 17.04.2025; Accepted: 30.04.2025.
Поступила: 18.12.2024; Одобрена: 17.04.2025; Принята: 30.04.2025.