CB-Whisper: Contextual Biasing Whisper using Open-Vocabulary Keyword-Spotting

Yuang Li, Yinglu Li, Min Zhang, Chang Su, Jiawei Yu,
Mengyao Piao, Xiaosong Qiao, Miaomiao Ma, Yanqing Zhao, Hao Yang
Huawei Translation Services Center, China
{liyuang3, liyinglu, zhangmin186, suchang8, yujiawei19,
piaomengyao1, qiaoxiaosong, mamiaomiao, zhaoyanqing, yanghao30}@huawei.com

Abstract

End-to-end automatic speech recognition (ASR) systems often struggle to recognize rare named entities, such as personal names, organizations, and terminologies that are not frequently encountered in the training data. This paper presents Contextual Biasing Whisper (CB-Whisper), a novel ASR system based on OpenAI's Whisper model that can recognize user-defined named entities by performing open-vocabulary keyword-spotting (KWS) before the decoder. The KWS module leverages text-to-speech (TTS) techniques and a convolutional neural network (CNN) classifier to match the features between the entities and the utterances. To integrate the recognized entities into the Whipser decoder and avoid hallucinations, we carefully crafted multiple prompts with spoken form hints. Experiments show that the KWS module based on the Whisper encoder's features can recognize unseen user-defined keywords effectively. More importantly, the proposed CB-Whisper substantially improves the mixed-error-rate (MER) and entity recalls compared to the original Whisper model on three internal datasets and two publicly available datasets including Aishell and ACL datasets that cover English-only, Chinese-only, and code-switching scenarios.

Keywords: speech recognition, contextual biasing, keyword-spotting

1. Introduction

End-to-end (E2E) automatic speech recognition (ASR) models (Chorowski et al., 2015; Graves, 2012; Graves et al., 2006) have gained popularity in recent years for their simplicity and unified architecture. However, they suffer from low recall of proper nouns that are rare in the training data. Contextual biasing (CB) is a technique that leverages external knowledge of the expected words in the speech input to mitigate the long-tail word problem. One of the early approaches is shallow fusion (Gourav et al., 2021; Zhao et al., 2019), which builds an n-gram finite state transducer from a predefined list of hot words and boosts their scores during beam search decoding. However, this approach has some drawbacks, such as the challenge of finding the optimal fusion weight. Therefore, researchers have proposed deep fusion methods (Chang et al., 2021; Chen et al., 2019; Han et al., 2022; Munkhdalai et al., 2022; Sainath et al., 2023), which train a contextual encoder together with the ASR model from scratch. These methods embed the entity words and the speech signal into the same feature space, and the decoder uses both contextual and acoustic information to generate the transcription. To adapt existing ASR models, some methods use adaptors (Sathyendra et al., 2022; Tong et al., 2023) to modify the intermediate features of the ASR model or pointer networks (Sun et al., 2023) to modify the output distributions. Moreover, shallow fusion and deep fusion methods can be combined to further enhance the performance (Le et al., 2021; Xu et al., 2023).

reference	北京商报讯记者王晔君日前
Whisper	北京商报训记者王叶军日前
CB-Whisper	北京商报训记者王晔君日前
reference	MTDNN maintained number of classes, heads, output layers.
Whisper	EmptyDNN maintains a number of classes' heads, output layers.
CB-Whisper	MTDNN maintain number of classes heads, output layers,
reference	这个不太能用什么bp啊、梯度base的computation啊来做
Whisper	这个不太能用什么BPRT-Due-based computation来做
CB-Whisper	这个不太能用什么bp梯度base的computation来做

Table 1: Examples of the comparison between the original Whisper with the proposed CB-Whisper under Chinese-only, English-only, and code-switching scenarios.

OpenAI's Whisper (Radford et al., 2022) is a state-of-the-art ASR model based on the Transformer (Vaswani et al., 2017) architecture and it was trained on a large-scale dataset of 680,000 hours of speech data. It has the ability to adaptively bias the generation of entity words by providing a prompt, a text prefix, to the decoder without any additional training. Nevertheless, long prompts can incur high computational costs and induce hallucinations. In this paper, we propose a novel method called Contextual Biasing Whisper (CB-Whisper) which incorporates a keyword-spotting (KWS) module between the encoder and the decoder of the

LREC-COLING 2024, pages 2941–2946

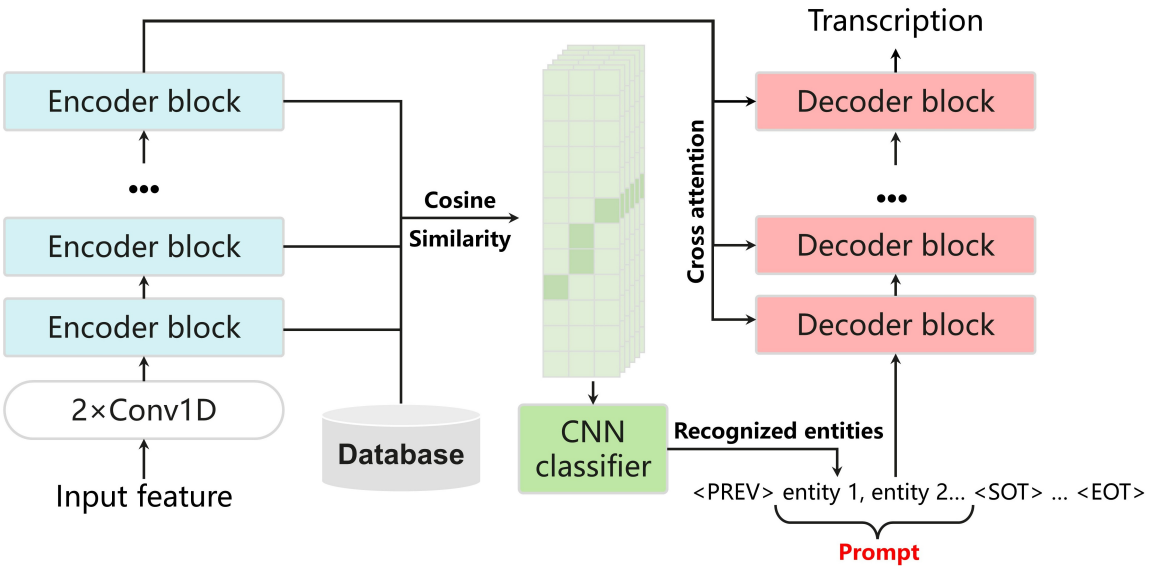

Figure 1: The flowchart for CB-Whisper. A CNN classifier recognizes entity words by using a cosine similarity matrix between the hidden states as input. The Whisper decoder takes the recognized entities as a prompt.

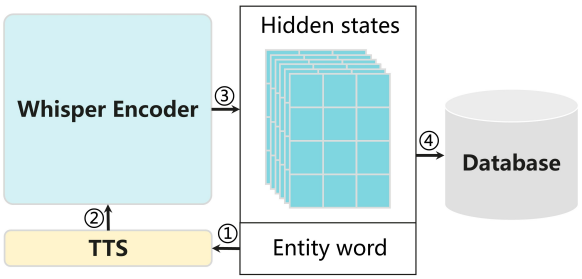

Figure 2: The preprocessing steps for entity words. The hidden states for each entity word are generated by TTS followed by the Whisper encoder.

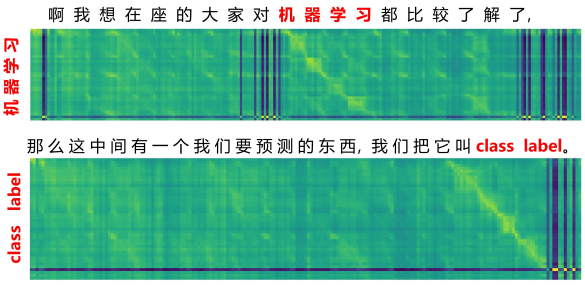

Figure 3: Visualizations of similarity matrices.

Whisper to search entity words before constructing prompts. The KWS module is inspired by vision-based KWS methods (Momeni et al., 2020; Shin et al., 2022) and uses features from the Whisper encoder to allow user-defined keywords during inference. Furthermore, different kinds of prompts are designed to incorporate the information of the extracted entity words into the Whisper decoder. By preserving the original weights of the Whisper model, our method achieves low computational cost and avoids catastrophic forgetting. In our evaluations, CB-Whisper achieves significantly better MER and entity recalls on various test sets with an average recall improvement of over 20% compared to the original Whisper. Some examples are shown in Table 1. Whisper incorrectly recognizes "MTDNN" as "EmptyDNN", while CB-Whisper correctly identifies it. Moreover, CB-Whisper also handles Chinese names and code-switching situations more effectively.

2. Methods

2.1. System Design

Before the system can be deployed, a database is created by extracting acoustic representations for pre-defined entity words. As shown in Figure 2, the speech signal for each entity word is generated with a TTS model. Then, the Whisper encoder is used to obtain multi-layer representations that will be later matched with the features of the input utterance.

As illustrated in Figure 1, the proposed CB-Whisper identifies entity words before the decoder by utilizing the hidden states of the encoder. In detail, the cosine similarity matrix is computed between the hidden representations of the input utterance and the stored features of each entity word. Since multi-layer hidden states are considered, the similarity matrix has multiple channels. A binary CNN classifier then determines if the entity word is present in the utterance by recognizing the diagonal pattern in the similarity matrix. Finally, the predicted entity words are used as prompts to guide the Whisper

transcription	银行业金融机构执行首套房贷款政策
positive	银行业 金融机构
negative (**easy**)	第二季 吴卓羲
negative (**hard**)	银行利息 商贸业 金融系分支机构

Table 2: Examples of the synthetic KWS dataset.

decoder so that these words can be recognized more accurately.

2.2. Keyword-Spotting Module

In the proposed system, KWS is formulated as a binary classification task. A deep neural network predicts the presence of an entity word in the utterance by taking a multi-channel feature map as input. This feature map is computed as the cosine similarity matrix between the hidden states of entity words and the input utterance at the frame level. As shown in Figure 3, if an entity word exists in the utterance, a diagonal pattern can be observed. Since such a pattern is a local feature, we choose CNN as the classification model. The advantage of using the similarity matrix rather than the original hidden states as input is that it allows us to do open-vocabulary KWS, meaning that we can detect words that were not seen during training or even from a different language. A potential drawback of our approach is that the computational complexity grows proportionally to the number of entity words. Nevertheless, in our experiments, we found that the CNN network has a significantly lower computational cost than the Whisper model, and the additional computation is acceptable for hundreds of entity words. Furthermore, we can exploit the parallelism of GPU to process multiple entity words simultaneously.

To train the CNN classifier, one could create the dataset from any ASR dataset by choosing positive and negative words based on the transcription. However, this approach is too simplistic, as the randomly selected negative samples may have significant differences from the positive samples, resulting in overfitting. Thus, we propose to include confusing words as hard negative samples. We first arrange the words in lexicographic order and then select negative words that are close to the position of positive words. We also invert all the words and repeat the selection process. In this way, the positive and negative words are likely to have overlaps (Table 2), which increases the difficulty of the classification task. For example, the positive sample "银行业" ("Yin Hang Ye") and negative sample "银行利息" ("Yin Hang Li Xi") are very similar.

naive prompt	entity 1, entity 2, entity 3 实体1、实体2、实体3
prompt-1	entity 1, entity 2, entity 3, ah, 实体1、实体2、实体3，这个呃，
prompt-2	The topic of today's speech is, entity 1, entity 2, entity 3. 今天演讲的主题是，实体1、实体2、实体3。
prompt-3	The topic of today's speech is, ah, entity 1, entity 2, entity 3. Okay, then I'll continue. 今天演讲的主题是这个呃，实体1、实体2、实体3。好，那我就继续讲。

Table 3: Prompts for the Whisper decoder.

dataset	utterances	duration (min)	entities
Internal-1	99	41	150
Internal-2	112	47	346
Internal-3	64	27	130
Aishell	808	76	226
ACL	123	51	210

Table 4: Statistics of ASR test sets.

2.3. Prompting Whisper Decoder

Prompt is a term that refers to a segment of text that provides the context or the objective for the generation of large language models. Whisper, an encoder-decoder model, also supports prompting because the decoder can be viewed as an internal language model. The decoder used the transcription of the previous utterance as a prompt during training to incorporate contextual information for the recognition of the following utterance. During inference, any text-only prompt that is relevant to the input utterance can serve as historical context for decoding, instead of using history transcription. In our experiments, we constructed the prompt with the recognized entity words. We employed four different types of prompts, which are displayed in Table 3. The naive prompt simply concatenated all the entity words. However, this method could result in high deletion errors, as the decoder sometimes removes disfluencies incorrectly. Therefore, we designed prompts that include filler words such as "呃" and "ah" (prompt-1) and indicate that the following speech was a spontaneous talk (prompt-2). We also combined these two strategies (prompt-3). Compared to the naive prompt, these prompts have a more similar form to the history transcriptions used during training of the Whisper model.

3. Experimental Setups

3.1. Model Configurations

We chose the Whisper-medium model for our study and obtained the hidden states for each utterance and entity word from the 10th to the 21st layers of

dataset	no prompt	naive prompt	prompt-1	prompt-2	prompt-3	oracle
Internal-1	4.9 / 83.4	5.8 / 93.3	**3.3** / 94.3	6.4 / 92.7	3.6 / **94.6**	3.5 / 96.1
Internal-2	6.7 / 71.1	7.3 / 91.8	4.4 / 92.1	7.1 / **92.9**	**4.3** / 92.4	4.0 / 97.9
Internal-3	14.9 / 67.7	14.7 / 92.8	**6.8** / **93.0**	14.9 / **93.0**	8.3 / **93.0**	8.2 / 97.0
Aishell	14.4 / 8.4	**12.2 / 71.8**	-	-	-	11.2 / 86.9
ACL	14.6 / 83.2	13.6 / 90.8	13.9 / 90.4	12.7 / **90.8**	**12.6** / 90.5	12.1 / 94.2

Table 5: The performance of CB-Whisper measured by **MER (%) / Entity Recall (%)**. The prompts refer to those presented in Table 3. "oracle" means the upper-bound performance with ground-truth entity words.

the Whisper encoder. Consequently, the input similarity matrix of the CNN classifier was a feature map with 12 channels. For the binary CNN classifier, we used Resnet-50 (He et al., 2016) and trained it from scratch for six epochs with a batch size of 64 and a learning rate of 5e-5. During inference, we used a beam size of five for beamsearch decoding. It was found that the Whisper model sometimes produced repetitions of a small phrase when a prompt was incorporated. To fix this issue, we re-decoded the utterance without prompt if the decoded transcription has a compression ratio larger than two.

3.2. Datasets

We used Aishell-1 (Bu et al., 2017), a Chinese ASR dataset with 150 hours of speech data, to create the training dataset for the CNN classifier. We extracted 20,000 words from the transcriptions and used Microsoft's TTS[1] to generate speech for each word. We evaluated the performance of KWS and ASR on three internal datasets and two open-source datasets. The internal datasets contained technical talks and manually labeled entity words that were mainly in Chinese but also included some English words. The open-source datasets were the Aishell hot word subset (Shi et al., 2023) in Chinese and the ACL dataset (Salesky et al., 2023) in English. For the internal datasets and ACL datasets, we concatenated multiple utterances to form longer utterances as the Whisper model was trained on 30-second audio clips. The statistics of these datasets are shown in Table 4. Note that the majority of entity words in the internal datasets and all entity words in the ACL dataset are out-of-vocabulary.

3.3. Metric

To measure the ASR performance, we used mixed-error-rate (MER) and entity recall. MER is a modified version of character-error-rate (CER) that can handle code-switching situations between English and Chinese. In this metric, we treat each Chinese character and each English word as a single unit. Therefore, MER is equivalent to CER for Chinese and word-error-rate (WER) for English. Entity recall is the percentage of correctly recognized named entities in the ASR output.

[1] https://github.com/rany2/edge-tts

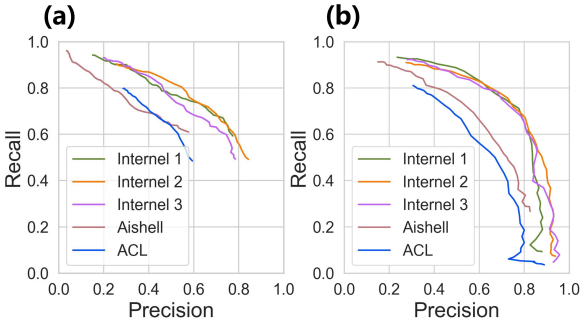

Figure 4: Precision-Recall curves for the proposed KWS method. The CNN classifiers were trained (a) without confusing words and (b) with confusing words.

4. Results

4.1. Results for TTS-based Keyword-Spotting

We first evaluated the KWS performance of the CNN classifier. The precision-recall (PR) curves are depicted in Figure 4. The PR curves were closer to the upper right corner when the training data included confusing words (Figure 4 (b)), compared to the model trained without confusing words (Figure 4 (a)). This suggests that the data creation method that incorporates confusing words can enhance the CNN classifier's robustness.

It is worth noting that the KWS module achieved outstanding open-vocabulary KWS performance. The internal datasets contain mostly technical terms that are not present in the training data, yet the system can achieve a recall of more than 0.8 and a precision of over 0.5 on all three datasets. The ACL dataset is in English, which is a different language from the Aishell training set. However, our system can still achieve a recall of more than 0.75 and a precision of more than 0.4.

4.2. Results for CB-Whisper

ASR performance of our proposed CB-Whisper is shown in Table 5. It can be observed that when naive prompts were incorporated, the entity recall on all datasets improved significantly. The most notable improvement was observed on the Aishell hot word subset, where the entity recall increased from 8.4% to 71.8%. This was mainly attributed to the

fact that Whisper almost completely failed to recognize Chinese names. For other datasets, the entity recall improved by 10% to 20% absolutely. However, the MER was less improved or even degraded on the Internal-1 and Internal-2 datasets. The main reason was that the Whisper model tended to omit filler words and disfluencies if the prompt was in a well-formatted text form. This problem can be alleviated by adding filler words to the prompt. Using prompt-1 can significantly reduce the MER on internal datasets, with the highest absolute MER improvement of 8.1% on the Internal-3 subset. On the contrary, indicating the style of the speech is less effective (prompt-2). However, on the English ACL talk dataset, prompt-2 is slightly better than prompt-1. The main reason is that the technical talks in ACL conferences tend to be more fluent than the internal data. Prompt-3 is the combination of prompt-1 and prompt-2, which are closer to a spoken form and resemble a presentation transcription. Although prompt-3 contributes to notable improvements across all datasets and avoids the degradation of MER compared to no prompt and naive prompt, it is only notably better than prompt-1 on the ACL dataset. These results indicate that the prompt should include the entity words in a style that matches the usage scenario.

5. Conclusions

In this paper, we propose CB-Whisper, a novel ASR framework that incorporates the prior knowledge of entity words to improve their recognition accuracy. The main component of our model is a KWS module that measures the similarity between the encoder hidden states of the synthetic speech of entity words and the input utterance. The identified entities are then used as prompts for the Whisper decoder. We conduct experiments on five test sets and demonstrate that our method achieves significant improvement in both MER and entity recall. Furthermore, our method does not require any fine-tuning of the Whisper model or any training of the CNN classifier in the target domain. For future works, we plan to further improve the efficiency and accuracy of the CNN classifier and explore automatic ways to optimize the form of prompts. Additionally, we will formulate more realistic scenarios where the entity words are not obtained from the ground-truth transcriptions but rather extracted from domain-specific text corpora using named entity recognition (NER) techniques or from slides using optical character recognition (OCR) methods.

6. Bibliographical References

Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. 2017. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In *Proc. O-COCOSDA*, pages 1–5.

Feng-Ju Chang, Jing Liu, Martin Radfar, Athanasios Mouchtaris, Maurizio Omologo, Ariya Rastrow, and Siegfried Kunzmann. 2021. Context-aware transformer transducer for speech recognition. In *Proc. ASRU*, pages 503–510.

Zhehuai Chen, Mahaveer Jain, Yongqiang Wang, Michael L. Seltzer, and Christian Fuegen. 2019. Joint grapheme and phoneme embeddings for contextual end-to-end ASR. In *Proc. Interspeech*. ISCA.

Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In *Proc. NeurIPS*, pages 577–585.

Aditya Gourav, Linda Liu, Ankur Gandhe, Yile Gu, Guitang Lan, Xiangyang Huang, Shashank Kalmane, Gautam Tiwari, Denis Filimonov, Ariya Rastrow, Andreas Stolcke, and Ivan Bulyko. 2021. Personalization strategies for End-to-End speech recognition systems. In *Proc. ICASSP*, pages 7348–7352.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. In *Proc. ICML*, Edinburgh, Scotland.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. ICML*, pages 369–376.

Minglun Han, Linhao Dong, Zhenlin Liang, Meng Cai, Shiyu Zhou, Zejun Ma, and Bo Xu. 2022. Improving End-to-End contextual speech recognition with fine-grained contextual knowledge selection. In *Proc. ICASSP*, pages 8532–8536.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proc. CVPR*.

Duc Le, Gil Keren, Julian Chan, Jay Mahadeokar, Christian Fuegen, and Michael L. Seltzer. 2021. Deep shallow fusion for RNN-T personalization. In *Proc. SLT*, pages 251–257.

Liliane Momeni, Triantafyllos Afouras, Themos Stafylakis, Samuel Albanie, and Andrew Zisserman. 2020. Seeing wake words: Audio-visual keyword spotting. *arXiv preprint arXiv:2009.01225*.

Tsendsuren Munkhdalai, Khe Chai Sim, Angad Chandorkar, Fan Gao, Mason Chua, Trevor Strohman, and Françoise Beaufays. 2022. Fast contextual adaptation with neural associative memory for on-device personalized speech recognition. In *Proc. ICASSP*, pages 6632–6636.

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*.

Tara N. Sainath, Rohit Prabhavalkar, Diamantino Caseiro, Pat Rondon, and Cyril Allauzen. 2023. Improving contextual biasing with text injection. In *Proc. ICASSP*, pages 1–5.

Elizabeth Salesky, Kareem Darwish, Mohamed Al-Badrashiny, Mona Diab, and Jan Niehues. 2023. Evaluating multilingual speech translation under realistic conditions with resegmentation and terminology. In *Proc. IWSLT*, pages 62–78, Toronto, Canada. ACL.

Kanthashree Mysore Sathyendra, Thejaswi Muniyappa, Feng-Ju Chang, Jing Liu, Jinru Su, Grant P. Strimel, Athanasios Mouchtaris, and Siegfried Kunzmann. 2022. Contextual adapters for personalized speech recognition in neural transducers. In *Proc. ICASSP*, pages 8537–8541.

Xian Shi, Yexin Yang, Zerui Li, Yanni Chen, Zhifu Gao, and Shiliang Zhang. 2023. SeACo-Paraformer: A non-autoregressive ASR system with flexible and effective hotword customization ability. *arXiv preprint arXiv:2308.03266*.

Hyeon Kyeong Shin, Hyewon Han, Doyeon Kim, Soo Whan Chung, and Hong Goo Kang. 2022. Learning audio-text agreement for open-vocabulary keyword spotting. *Proc. InterSpeech*.

Guangzhi Sun, Chao Zhang, and Philip C. Woodland. 2023. Minimising biasing word errors for contextual ASR with the tree-constrained pointer generator. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:345–354.

Sibo Tong, Philip Harding, and Simon Wiesler. 2023. Slot-triggered contextual biasing for personalized speech recognition using neural transducers. In *Proc. ICASSP*, pages 1–5.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. NeurIPS*, volume 30.

Yaoxun Xu, Baiji Liu, Qiaochu Huang, Xingchen Song, Zhiyong Wu, Shiyin Kang, and Helen Meng. 2023. CB-Conformer: Contextual biasing conformer for biased word recognition. In *Proc. ICASSP*, pages 1–5.

Ding Zhao, Tara N. Sainath, David Rybach, Pat Rondon, Deepti Bhatia, Bo Li, and Ruoming Pang. 2019. Shallow-fusion end-to-end contextual biasing. In *Proc. Interspeech*. ISCA.