IMPROVING RARE-WORD RECOGNITION OF WHISPER IN ZERO-SHOT SETTINGS

Yash Jogi[†]*, Vaibhav Aggarwal*[†]*, Shabari S Nair, Yash Verma, Aayush Kubba*

Sprinklr, India

{yash.jogi, vaibhav.aggarwal, shabari.nair, yash.verma, aayush.kubba}@sprinklr.com

ABSTRACT

Whisper, despite being trained on $680K$ hours of web-scaled audio data, faces difficulty in recognizing rare words like domain-specific terms, with a solution being contextual biasing through prompting. To improve upon this method, in this paper, we propose a supervised learning strategy to fine-tune Whisper for contextual biasing instruction. We demonstrate that by using only 670 hours of Common Voice English set for fine-tuning, our model generalizes to 11 diverse open-source English datasets, achieving a 45.6% improvement in recognition of rare words and 60.8% improvement in recognition of words unseen during fine-tuning over the baseline method. Surprisingly, our model's contextual biasing ability generalizes even to languages unseen during fine-tuning.

Index Terms— end-to-end speech recognition, contextual biasing, Whisper

1. INTRODUCTION

Recent years have witnessed a growing interest in training end-to-end speech recognition systems on large-scale audio data, leading to robust and generalized speech recognition models [1, 2]. Amongst such models, Whisper [2] stands out as the only open-source speech recognition model trained on a massive scale of $680,000$ hours of web-scraped audio data on various speech related tasks, achieving low Word-Error-Rate (WER) across diverse domains and languages. Hence, such an open-source model serves as a foundational model in the domain of speech and has been used "out of the box" in a variety of applications [3, 4, 5]. Despite being trained on such a large-scale dataset, Whisper struggles with the recognition of rare words such as proper nouns or domain specific words, which might be sparse or absent from its training data [6].

This issue of difficulty in recognizing rare words has been prevalent among other end-to-end Automatic Speech Recognition (ASR) models as well [7, 8]. To address this challenge, contextual biasing has been used extensively as one of the most popular solutions [9, 10]. Specifically, this approach involves providing relevant contextual knowledge in the form of a list of words or phrases which can be contact names, application names, a list of medical terms, or any other domain specific words to an ASR model, so as to make their recognition more accurate. For example, in the medical domain, an ASR model trained on general data might struggle to transcribe rare terms like "spirometry". However, providing relevant context, such as a list of medical terms, can significantly reduce WER and enhance the model's utility and reliability.

A key difference between Whisper and previous ASR architectures such as wav2vec2.0 [1] or Transformer-Transducer [11] is Whisper's unique prompt functionality. Specifically, Whisper differs from other ASR models in that it enables transcription control through prompting. As suggested in [2], OpenAI's official documentation for Whisper [1] mentions several ways to use the prompt feature, particularly to increase the recognition accuracy for rare words by including such a word list in the prompt and various ways to control transcription style. This prompt feature of Whisper has been used in prior work [5] for a variety of tasks in a zero-shot manner such as audio-visual speech recognition, wherein the visual context is converted to a list of bias words using CLIP [12], which is then integrated to prompt, improving transcription accuracy over using audio alone.

However, Whisper was not trained to follow any particular instruction[1]. The model has been trained to use the previous speech segment's transcription as prompt for transcribing the current speech segment, similar to how a base GPT (Generative Pre-trained Transformers) [13] model generates the next series of tokens given previously entered text. Hence, Whisper's prompt operates similar to a base GPT model[1], such as GPT-2, in that although Whisper is not explicitly trained on instructions, it can still follow an instruction given in the prompt in a zero-shot manner.

Previous works have shown that fine-tuning base GPT models with instructions significantly increases their ability to follow instructions and improves their zero-shot performance across various datasets and tasks [14]. In this paper, we leverage this concept of instruction-tuning to fine-tune Whisper for a single instruction of contextual biasing, particularly for rare words. Additionally, we investigate whether fine-tuning Whisper for biasing instruction on a single English dataset leads to generalization across other English datasets. To this end, we present a novel data-efficient supervised learn-

[†]Equal contribution.

[1]https://platform.openai.com/docs/guides/speech-to-text/prompting/

ing method to improve the performance of Whisper for contextual biasing through prompting. We propose a specific prompt selection strategy aimed at more robust training, and a weighting scheme for the Cross Entropy loss for maximizing learning for the given task. We name our fine-tuned model *Bias-Whisper* or *B-Whisper*.

Giving credence to our hypothesis, our experiments suggest that despite using only 670 hours of Common Voice English set [15] for fine-tuning, our model generalizes for biasing instruction to 11 open-source datasets such as TED-LIUM [16], SLURP [17], Vox Populi [18], etc., outperforming the baseline by a significant margin in terms of WER for rare words. Surprisingly, for contextual biasing, the model shows increased performance even on words not seen during fine-tuning. Moreover, despite being fine-tuned only on an English dataset, our model performs well on unseen languages like French, Spanish, Italian, and German for biasing instruction—further testifying to the zero-shot capabilities of our model. Notably, existing research on contextual biasing has predominantly focused on English, leaving a gap in studies addressing other languages. Our study reduces this gap by showcasing a method that generalizes to unseen languages although trained on an English dataset.

2. RELATED WORK

In this section, we discuss recent research diving into the prompting capabilities of Whisper. This work [5] is among the first to demonstrate the zero-shot capabilities of prompting in Whisper. Specifically, for various tasks such as audio-visual speech recognition, code-switched speech recognition, and speech translation, this paper presented a way contextual knowledge can be integrated into the Decoder through prompt, which significantly enhances Whisper's transcription accuracy. Another paper [19] proposed a prompt-tuning methodology that adds prompts to both the Encoder and Decoder parts of the model to transcribe the target speaker's speech from overlapped multi-talker audios.

The closest work to our approach is [20], which introduces a method for creating domain-sensitive Whisper by fine-tuning it on textual prompts that describe the audio context and genre. However, the paper's definition of 'prompt' is somewhat vague and domain-dependent, potentially resulting in subjective variations in model performance.

3. BACKGROUND

In this section, we briefly discuss Whisper and its architectural details [2]. Whisper is a family of sequence-to-sequence models that follow the encoder-decoder Transformer architecture [21]. Instead of a gold standard human-validated dataset, Whisper has been trained in a weakly-supervised fashion on $680,000$ hours of web-scaled speech dataset. One of the distinctive features of the Whisper model is its ability to handle a variety of speech-related tasks, such as transcription, translation, language identification, and voice activity detection (VAD).

Given an input speech signal, its log mel-spectogram is first calculated, denoted as $A_{T\times M}$, where M is the number of mel-bins and T is the total number of frames. These features are then passed through the encoder E, yielding speech representations h for the given speech signal.

$$h = E(A) \tag{1}$$

The speech representations h, along with the sequence of prompt tokens $c = \{c_1, c_2, , ..., c_C\}$ are then passed to the decoder D. The decoder generates probability distribution p_i for the next token y_i, conditioned on the previously decoded outputs $y_{<i}$, speech representations, and sequence of prompt tokens:

$$p_i = P(y_i|y_{<i}, h, c) \tag{2}$$

For more details regarding the multi-tasking format through prompt tokens or Whisper's training details, refer [2].

4. METHODOLOGY

In this section, we introduce a supervised learning scheme to fine-tune Whisper for a single instruction for contextual biasing, specifically for rare words. We now outline our approach for creating the training prompt for a particular utterance and provide details regarding the loss function used for supervised learning.

Prompt Selection Strategy: Prior to training, for each speech utterance, its reference transcript and hypothesis transcript generated using base Whisper model are aligned to find out incorrectly transcribed rare words in the reference transcript. A word is considered a rare word if it falls outside the most common words accounting for 90% of word occurrences in the train set. Out of these incorrectly transcribed rare words, one word is randomly selected to be the true-bias word for that utterance. By including incorrectly transcribed rare words as true-bias words in the biasing list, we can focus the fine-tuning on resolving base Whisper's mistakes, hence allowing the model to learn to better utilize the biasing list to correct its mistakes. Moreover, since we are unaware of the frequency distribution of words in Whisper's original training data, we believe selecting true-bias words based on incorrect transcription and rarity rather than rarity alone would lead to better results [6].

We maintain a global list V of such incorrectly transcribed words that fall outside the most common words accounting for 90% of word occurrences in the training dataset. For each utterance, we randomly select L false-bias words from this list V, which do not occur in the given reference transcript. Here

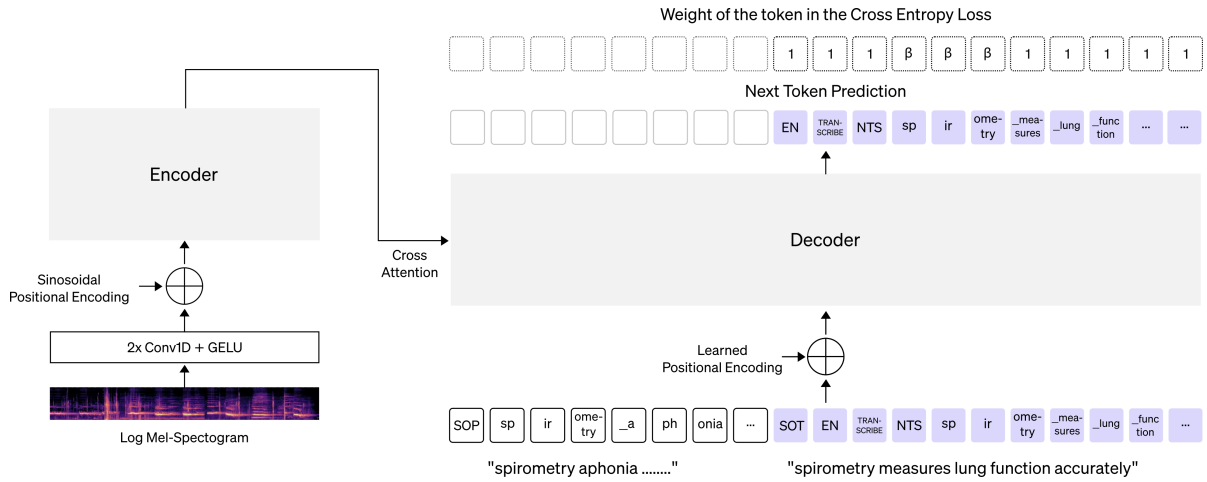

Fig. 1. **Overview of B-Whisper.** This figure depicts the train-time inputs and outputs for B-Whisper with an example. In the bottom-most sequence of squares, colored squares contain task-specifiers and transcript tokens, whereas colorless squares contain prompt tokens. In this case, the word "spirometry" is the true-bias word for the reference transcript "spirometry measures lung function accurately". As such, the tokens of "spirometry" are given weight $\beta > 1$ and the rest of the tokens are given weight 1 in Cross Entropy Loss, as shown in the top-most sequence of squares.

L is randomly chosen from $[25, 150]$ so as to ensure varying biasing list size in training. To enhance the resilience of the model in cases where the speech utterance does not contain any word present in the biasing list, we randomly drop the true-bias words from the list with probability P_{neg}, hence keeping the biasing list full of false-bias words only in such a scenario. In addition to this, to ensure the model performs well even when no biasing list is given, we completely drop the biasing list with probability P_{empty}. We combine the selected false-bias and true-bias words in a random order to create the biasing list. Then, we concatenate all the words in the biasing list using "space" characters to form the textual prompt for the given speech utterance.

Weighted Cross Entropy (CE) Loss: The number of rare words in an utterance is generally in minority compared to the total number of words present in it. Hence, to prioritize learning for such words, we introduce weight in the CE Loss. We increase the weight of the corresponding tokens for the true-bias words to $\beta > 1$ and keep the weight corresponding to the other tokens to 1. The modified loss becomes:

$$L = \sum_{i=1}^{S} w_i H(q_i, p_i) \tag{3}$$

where H is the cross entropy function, p_i is as defined in (2), q_i is the one-hot ground truth vector, w_i is $\beta > 1$ if y_i is a token of true-bias word, else it is 1, and S is the total number of tokens in the transcript. This choice of prompt selection strategy and loss makes the model learn Whisper's mistakes effectively during fine-tuning. Figure 1 shows our approach.

5. EXPERIMENTAL SETTINGS

Train Dataset: We used the Common Voice [15] English 17.02 dataset for training, which consists of 2615 hours of labeled audios from over $90k$ global speakers, providing a broad range of accents and speakers—which can help for effective model generalization. In our experiments, we used a subset of approximately 670 hours from the Common Voice training split as our train set, and the official val split for validation. We restricted to a smaller train set to prevent excessive bias towards the training distribution, thereby maintaining Whisper's generalized performance. Prior to training, we also normalized all the reference transcripts with the English normalizer function available in the official Whisper repository. [2]

Test Datasets: In addition to testing on the Common Voice test set and the entire Artie Bias [22] which is a subset of the Common Voice English dataset, we were also interested in comprehensively evaluating our model on out-of-domain datasets. Hence, we chose a set of 9 open-source out-of-domain English datasets for testing: official evaluation set of Chime6 [23], CORAAL:VLD v. 2021.07 component of CORAAL [24], test set of SLURP [17], test set of TED-LIUM [16], English test set of VoxPopuli [18], test-clean and test-other splits of LibriSpeech [25], English test set of FLEURS [26], and test set of "Medical Speech, Transcription, Intent" [27]. In addition to these, we also evaluated on the French (fr), Spanish (es), Italian (it), and German (de) test splits of Multilingual LibriSpeech (MLS) [28]. All the evaluations on these out-of-domain datasets were done in a

[2] https://github.com/openai/whisper/

zero-shot setting, without using the training set of the specific dataset during model fine-tuning, to evaluate generalization. Additionally, none of these sets were used in the training of Whisper. We normalized the reference transcripts for all 11 datasets as recommended in [2].

Test-Time Biasing List Creation: We test on different biasing list conditions to ensure the model's robustness across various scenarios. We adopt two scenarios for the creation of the biasing list.

In Scenario-1, we build the biasing list along the lines of [6, 10, 29], wherein for each utterance, for true-bias words, we extract words in the reference transcript that fall outside the most common words accounting for 90% of word occurrences in the train set. We sample false-bias words from a global list of rare words—words that are not present in the most common words accounting for 90% of word occurrences in the corresponding train set.

In Scenario-2, we aim to emulate the instances where none of the words in the biasing list are spoken in the given audio. To this end, following along the lines of [30], we create the biasing list which only contains false-bias words sampled randomly from the global list of rare words.

Model Details: We fine-tuned our model on top of the pre-trained Whisper Large architecture initialized with the official 'openai/whisper-large' checkpoint. The learning rate was set to 10^{-7} with Adam optimizer and a linear rate decay. The dropout rate was set to 10%. We trained our model for 1 epoch. We extended the positional embedding enough to accommodate 756 tokens. The value of β for weighted Cross Entropy Loss was set to 1.1. For all the experiments, feature extraction, pre-processing, and tokenization steps were the same as mentioned in [2]. For biasing list selection during training, we chose $P_{neg} = 0.3$ and $P_{empty} = 0.2$. For transcription generation for both B-Whisper and Whisper, we kept beam size 1.

Evaluation Metrics: In line with previous works [6, 29, 9, 31, 32] we use the following four evaluation metrics - WER: word error rate for all the words, U-WER: unbiased word error rate, or word error rate for words not part of the biasing list, R-WER: rare word error rate or word error rate for words present in the biasing list, and OOV-WER: word error rate for OOV (out-of-vocabulary) words present in the biasing list. Here, OOV words refer to the words that are not present in our training word vocabulary, whose size is around $230K$. However, these words might be present in the pre-training dataset for Whisper. Lower is better for all the mentioned evaluation metrics. We report all the mentioned metrics in $\%$.

6. RESULTS

Effect of Contextual Biasing: We have comprehensively compared the performance of Whisper and B-Whisper across 11 open-source English datasets, out of which for 9 datasets in a zero-shot setting, the results of which are given in Table 1. Here "Whisper" and "B-Whisper" denote the use of these models without providing any biasing list, whereas "Whisper+P" and "B-Whisper+P" denote the use of biasing list via prompting (biasing list size $N = 70$). We follow this convention for the rest of the paper. It should be noted that even when we did not pass any biasing list to Whisper and B-Whisper, we used the corresponding biasing list used in Whisper+P and B-Whisper+P in order to calculate their U-WER, R-WER, and OOV-WER values. The biasing list used for Table 1 was created as mentioned in Scenario-1.

Comparing the performance of Whisper+P with Whisper, Whisper+P sees a clear reduction in average R-WER from 23.7% to 18.0%, and average OOV-WER from 60% to 37.1%. This demonstrates the usefulness of using prompt 'out-of-the-box' for contextual biasing in Whisper. However, U-WER of Whisper+P increases in comparison with Whisper across all the datasets, resulting in an increased WER across 6 out of 11 datasets in comparison to Whisper. This increase in U-WER and overall WER can be attributed to the fact that Whisper's prompt expects the transcription of the previous speech segment rather than a list of biasing words. In other words, the prompt for Whisper is misaligned for the instruction of contextual biasing.

Compared to Whisper+P, R-WER reduces on average by 45.6% for B-Whisper+P. Although we only used Common Voice to train our model, the improvement is consistent across all the datasets as evident from the R-WERR values, thus proving the effectiveness of our approach in a zero-shot setting. Even in the case of OOV-WER, where the words to be biased are completely absent from the fine-tuning train set, we see that B-Whisper+P has achieved the best performance across all the datasets, with an average improvement of 60.8% over Whisper +P. This shows the strong generalization ability of B-Whisper+P. An interesting observation here is that the relative improvement in R-WER and OOV-WER is higher in datasets where Whisper already performs relatively well, such as Artie Bias and VoxPopuli. Example predictions illustrating the effectiveness of B-Whisper+P are given in Table 2.

Moreover, B-Whisper+P largely fixes the poor performance of Whisper+P when it comes to U-WER and WER, achieving the best results in U-WER for 7 and WER for 9 out of 11 datasets. When used without prompting, B-Whisper achieves near identical WER as that of Whisper in most datasets, with slight deviations possibly due to distribution shift during fine-tuning. Additionally, the hyper-parameter P_{empty} can also influence these results. Overall, this shows that B-Whisper has almost retained its original behaviour in conditions where no biasing list is provided, in spite of fine-tuning.

Impact of Biasing List Size: To analyze the effect of biasing list size on contextual biasing capabilities, we have plotted Figure 2 (a) and (b), which shows the average values of WER, U-WER, and R-WER across 11 datasets of Whisper+P

		Common Voice	Artie Bias	Chime6	CORAAL	FLEURS	LS-Clean	LS-Other	Medical	SLURP	TED-LIUM	VoxPopuli	Average
WER	Whisper	11.0	6.7	25.3	**19.7**	6.3	2.6	5.4	8.4	15.8	**4.6**	7.1	10.3
	Whisper + P	10.1	6.3	32.5	28.3	6.0	2.2	4.6	8.5	16.3	8.3	9.6	12.1
	B-Whisper	10.0	6.4	23.9	20.9	6.6	2.7	5.5	8.3	16.2	4.8	7.8	10.3
	B-Whisper + P	**7.0**	**4.6**	**22.7**	20.4	**5.2**	**1.5**	**3.4**	**6.4**	**14.9**	4.7	**6.9**	**8.9**
U-WER	Whisper	8.6	5.3	24.5	**18.4**	5.4	1.7	3.6	7.3	**14.4**	**4.4**	**6.8**	9.1
	Whisper + P	9.3	5.9	32.5	27.5	5.5	1.8	3.8	7.9	15.8	8.1	9.4	11.6
	B-Whisper	7.5	5.0	22.7	19.6	5.4	1.6	3.7	7.2	14.7	4.6	7.4	9.0
	B-Whisper + P	**6.8**	**4.7**	**22.7**	19.8	**5.2**	**1.5**	**3.3**	**6.1**	14.5	4.7	6.9	**8.7**
R-WER	Whisper	35.7	22.4	32.1	44.1	16.8	10.9	21.5	21.2	34.3	10.1	12.0	23.7
	Whisper + P	18.6	10.7	32.4	44.9	11.3	6.4	11.6	15.2	22.9	11.8	12.6	18.0
	B-Whisper	35.8	22.0	34.7	47.3	21.2	11.5	21.4	20.3	35.8	9.8	14.2	24.9
	B-Whisper + P	**8.7**	**4.0**	**23.7**	**32.9**	**5.5**	**2.2**	**5.0**	**10.5**	**19.5**	**5.2**	**5.5**	**11.2**
R-WERR		53.2	62.6	26.8	26.7	51.3	65.6	56.9	30.9	14.9	55.9	56.3	45.6
OOV-WER	Whisper	75.7	62.1	62.2	73.5	58.0	51.5	64.5	72.1	68.0	23.0	49.0	60.0
	Whisper + P	33.2	24.3	46.8	64.9	38.3	24.4	30.2	54.4	37.8	23.0	30.9	37.1
	B-Whisper	71.9	56.8	58.5	76.4	63.0	52.0	65.2	71.7	68.8	29.2	50.5	60.4
	B-Whisper + P	**8.7**	**5.4**	**27.6**	**38.2**	**14.8**	**7.2**	**10.2**	**29.3**	**17.3**	**6.1**	**11.4**	**16.0**
OOV-WERR		73.8	77.8	41.0	41.1	61.4	70.5	66.2	46.1	54.2	73.5	63.1	60.8

Table 1. Overview of various WER metrics of Whisper without biasing list, Whisper with biasing list ($N = 70$), B-Whisper without biasing list, and B-Whisper with biasing list ($N = 70$). Here, R-WERR denotes the relative percentage reduction in R-WER of B-Whisper+P compared to the Whisper+P. The same applies to OOV-WERR.

and B-Whisper+P for varying biasing list size. The biasing list size is limited to 70 for Whisper since by default it allows a maximum of 224 tokens as input for prompt, which approximates to around 70 words. We can see that both models suffer from degradation in R-WER as biasing list size increases, as the number of false-bias words has increased. This change is more pronounced in B-Whisper+P. However, the values of R-WER for B-Whisper+P are around 7% points lower than Whisper+P, on account of the supervised fine-tuning. Surprisingly, in the case of U-WER for Whisper+P, there is an improvement when going from $N = 35$ to $N = 70$. On the other hand, B-Whisper shows a near-consistent U-WER with an increase in N. This could be attributed to our training regime wherein the biasing list size is randomly chosen for each training sample, making it more robust to changes in biasing list size.

To gauge the effect of over-biasing with different biasing list sizes, Figure 2 (c) shows the average WER across 11 datasets for both Whisper+P and B-Whisper+P. The biasing lists, as mentioned in Scenario-2, contain only false-bias words. For both the models, we see a decrease in WER when going from $N = 35$ to $N = 70$. However, the WER for B-Whisper+P is around 3% points lower, indicating its effectiveness in handling purely false bias words in a biasing list. Moreover, it is interesting to note that when going from $N = 70$ to $N = 150$, there is a minimal increase in WER. This shows that B-Whisper is largely resistant to over-biasing with different biasing list sizes.

Evaluation on Unseen Languages: In order to evaluate the biasing capabilities of our model on languages unseen during fine-tuning, we performed evaluation on four European languages, the results of which have been summarized in Table 3. Although contextual biasing via prompting is effective in Whisper for English, it proves ineffective for certain non-English languages such as French and German, hence degrading both R-WER and U-WER. This might be due to the Whisper model's pre-training dataset, which primarily consists of English data, along with the misalignment in prompt definition as mentioned earlier. To our surprise, in contrast to Whisper+P, B-Whisper+P has been able to successfully translate its biasing capabilities from English to these languages, achieving better OOV-WER. Consequently, it has seen the best WER in all four languages. On the other hand, it has not seen a similar improvement in U-WER, where it lags slightly behind Whisper in 3 out of 4 languages.

Overall, our results show that B-Whisper performs well not only on its fine-tuning test set but also adeptly integrates broad knowledge acquired during pre-training of Whisper

Transcript	Whisper	B-Whisper	Whisper+P	B-Whisper+P	Biasing List
... foreign rule to the **phanariote** period	... foreign rule to the **fanaret** period	... foreign rule to the **fanaret** period	... foreign rule to the **phanaret** period	... foreign rule to the **phanariote** period	[..,mcphillips, **phanariote**, lukyamuzi,..]
i feel pain in my ears with **tinnitus**	i feel pain in my ears with **cheetahs**	i feel pain in my ears with **cheetahs**	epilpian in my ears with **cheetahs**	i feel pain in my ears with **tinnitus**	[..,kimbolton, **tinnitus**, polygynandy,..]

Table 2. Example transcripts of various models on samples from Common Voice (first) and Medical test set (second).

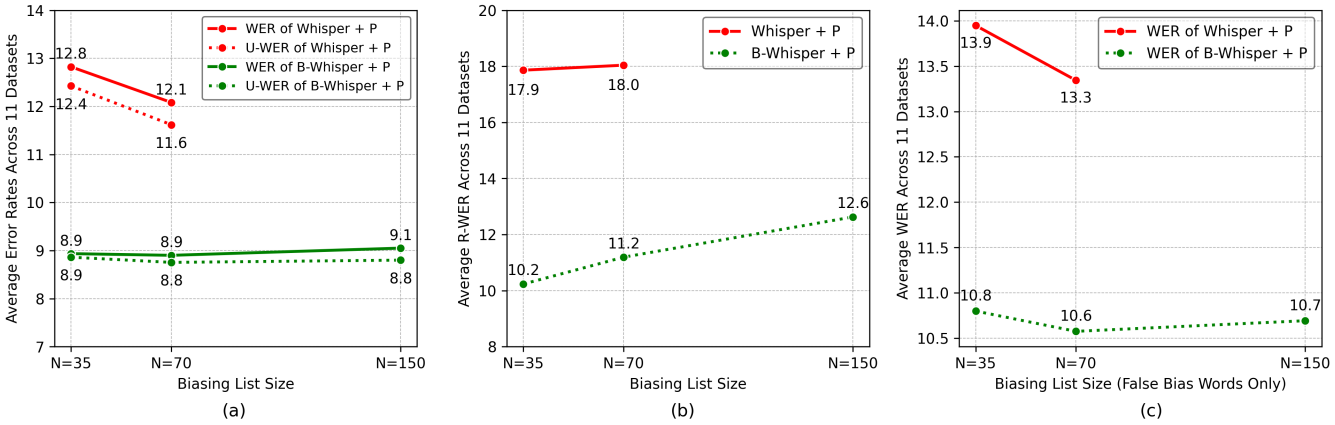

Fig. 2. Average values across 11 datasets of (a) WER and U-WER, (b) R-WER, (c) WER, for biasing list sizes 35, 70 and 150 for Whisper + P and B-Whisper + P. Here, in case of (c), the biasing list contains only false-bias words.

		fr	de	es	it
WER	Whisper	8.3	6.3	4.4	12.9
	Whisper + P	10.8	15.2	5.9	13.6
	B-Whisper	7.4	7.6	5.1	14.8
	B-Whisper + P	**5.5**	**5.5**	**3.7**	**11.6**
U-WER	Whisper	7.6	**6.0**	**4.0**	**13.6**
	Whisper + P	10.4	15.0	6.3	15.4
	B-Whisper	6.3	7.2	4.7	15.7
	B-Whisper + P	**5.9**	7.1	4.2	14.3
OOV-WER	Whisper	17.3	15.6	10.3	19.6
	Whisper + P	21.0	27.7	8.7	12.7
	B-Whisper	18.8	17.2	10.9	20.2
	B-Whisper + P	**6.7**	**4.2**	**3.4**	**5.2**

Table 3. Results for Whisper and B-Whisper (with and without biasing list with $N = 70$) on various test sets of MLS.

with specific skills acquired through our fine-tuning procedure, thereby generalizing on unseen languages for a new instruction. Although we keep a maximum biasing list size of 150 in our experiments, this can be extended by a retriever mechanism to filter out relevant words, similar to what is done in [33].

7. CONCLUSION

This paper explores the impact of fine-tuning Whisper for contextual biasing instruction, specifically for rare words in zero-shot settings. Our main finding is that despite using a small set of 670 hours English dataset for fine-tuning, our model B-Whisper outperforms Whisper by a large margin on 11 open-source English datasets and also on languages unseen during fine-tuning process. Due to its generalized performance despite using a relatively small amount of training data, this approach can be particularly valuable for ASR practitioners and researchers struggling with low accuracy on domain-specific terms in Whisper—especially those with limited access to extensive computational resources or large industrial-scale labeled datasets. Another exciting future work could be extending our approach by fine-tuning Whisper on multiple instructions useful for tasks such as audio-visual speech recognition, code-switching speech recognition, etc.

8. ACKNOWLEDGEMENTS

A discussion with Yoginkumar Patel in May 2023 inspired this research direction. We gratefully acknowledge his valuable comments and support throughout this work.

9. REFERENCES

[1] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12449–12460, 2020.

[2] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*. PMLR, 2023, pp. 28492–28518.

[3] Michael Moor, Oishi Banerjee, Zahra Shakeri Hossein Abad, Harlan M Krumholz, Jure Leskovec, Eric J Topol, and Pranav Rajpurkar, "Foundation models for generalist medical artificial intelligence," *Nature*, vol. 616, no. 7956, pp. 259–265, 2023.

[4] Yuan Gong, Sameer Khurana, Leonid Karlinsky, and James Glass, "Whisper-AT: Noise-robust automatic speech recognizers are also strong general audio event taggers," in *Proc. Interspeech 2023*, 2023, pp. 2798–2802.

[5] Puyuan Peng, Brian Yan, Shinji Watanabe, and David Harwath, "Prompting the hidden talent of web-scale speech models for zero-shot task generalization," in *Proc. Interspeech 2023*, 2023, pp. 396–400.

[6] Guangzhi Sun, Xianrui Zheng, Chao Zhang, and Philip C. Woodland, "Can contextual biasing remain effective with Whisper and GPT-2?," in *Proc. Interspeech*, 2023, pp. 1289–1293.

[7] Mahaveer Jain, Gil Keren, Jay Mahadeokar, Geoffrey Zweig, Florian Metze, and Yatharth Saraf, "Contextual RNN-T for open domain ASR," in *Proc. Interspeech*, 2020, pp. 11–15.

[8] Saket Dingliwal, Monica Sunkara, Srikanth Ronanki, Jeff Farris, Katrin Kirchhoff, and Sravan Bodapati, "Personalization of CTC speech recognition models," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 302–309.

[9] Kanthashree Mysore Sathyendra, Thejaswi Muniyappa, Feng-Ju Chang, Jing Liu, Jinru Su, Grant P Strimel, Athanasios Mouchtaris, and Siegfried Kunzmann, "Contextual adapters for personalized speech recognition in neural transducers," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8537–8541.

[10] Jiyang Tang, Kwangyoun Kim, Suwon Shon, Felix Wu, and Prashant Sridhar, "Improving ASR contextual biasing with guided attention," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12096–12100.

[11] Qian Zhang, Han Lu, Hasim Sak, Anshuman Tripathi, Erik McDermott, Stephen Koo, and Shankar Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and RNN-T loss," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7829–7833, 2020.

[12] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al., "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[13] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al., "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, pp. 9, 2019.

[14] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al., "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27730–27744, 2022.

[15] Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020, pp. 4218–4222.

[16] François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Esteve, "TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation," in *Speech and Computer: 20th International Conference, SPECOM 2018, Leipzig, Germany, September 18–22, 2018, Proceedings 20*. Springer, 2018, pp. 198–208.

[17] Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser, "SLURP: A spoken language understanding resource package," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 7252–7262.

[18] Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proceedings of the 59th Annual Meeting of the*

Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 2021, pp. 993–1003, Association for Computational Linguistics.

[19] Hao Ma, Zhiyuan Peng, Mingjie Shao, Jing Li, and Ju Liu, "Extending Whisper with prompt tuning to target-speaker ASR," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12516–12520.

[20] Feng-Ting Liao, Yung-Chieh Chan, Yi-Chang Chen, Chan-Jan Hsu, and Da-shan Shiu, "Zero-shot domain-sensitive speech recognition with prompt-conditioning fine-tuning," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2023, pp. 1–8.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[22] Josh Meyer, Lindy Rauchenstein, Joshua D Eisenberg, and Nicholas Howell, "Artie bias corpus: An open dataset for detecting demographic bias in speech applications," in *Proceedings of the twelfth language resources and evaluation conference*, 2020, pp. 6462–6468.

[23] Shinji Watanabe, Michael Mandel, Jon Barker, Emmanuel Vincent, Ashish Arora, Xuankai Chang, Sanjeev Khudanpur, Vimal Manohar, Daniel Povey, Desh Raj, David Snyder, Aswin Shanmugam Subramanian, Jan Trmal, Bar Ben Yair, Christoph Boeddeker, Zhaoheng Ni, Yusuke Fujita, Shota Horiguchi, Naoyuki Kanda, Takuya Yoshioka, and Neville Ryant, "The CHiME-6 challenge: tackling multispeaker speech recognition for unsegmented recordings," in *Proceedings of the 6th International Workshop on Speech Processing in Everyday Environments (CHiME 2020)*, Barcelona, Spain (Virtual), May 2020.

[24] Minnie Quartey, Charlie Farrington, Tyler Kendall, Lucas Jenson, Chloe Tacata, and Jaidan McLean, "The corpus of regional african american language: VLD (valdosta, ga 2017)," 2020, The Online Resources for African American Language Project.

[25] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[26] Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna, "Fleurs: Few-shot learning evaluation of universal representations of speech," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 798–805.

[27] Figure Eight Inc., "Medical speech, transcription, and intent," `https://www.kaggle.com/datasets/paultimothymooney/medical-speech-transcription-and-intent/data`, 2019.

[28] Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, and Ronan Collobert, "MLS: A large-scale multilingual dataset for speech research," in *Proc. Interspeech 2020*, 2020, pp. 2757–2761.

[29] Duc Le, Mahaveer Jain, Gil Keren, Suyoun Kim, Yangyang Shi, Jay Mahadeokar, Julian Chan, Yuan Shangguan, Christian Fuegen, Ozlem Kalinli, Yatharth Saraf, and Michael L. Seltzer, "Contextualized streaming end-to-end speech recognition with trie-based deep biasing and shallow fusion," in *Proc. Interspeech 2021*, 2021, pp. 1772–1776.

[30] Zhong Meng, Zelin Wu, Rohit Prabhavalkar, Cal Peyser, Weiran Wang, Nanxin Chen, Tara N. Sainath, and Bhuvana Ramabhadran, "Text injection for neural contextual biasing," in *Proc. Interspeech 2024*, 2024, pp. 2985–2989.

[31] Muhammad Shakeel, Yui Sudo, Yifan Peng, and Shinji Watanabe, "Contextualized end-to-end automatic speech recognition with intermediate biasing loss," in *Proc. Interspeech 2024*, 2024, pp. 3909–3913.

[32] Guangzhi Sun, Chao Zhang, and Philip C Woodland, "Minimising biasing word errors for contextual ASR with the tree-constrained pointer generator," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 345–354, 2022.

[33] Sai Muralidhar Jayanthi, Devang Kulshreshtha, Saket Dingliwal, Srikanth Ronanki, and Sravan Bodapati, "Retrieve and copy: Scaling ASR personalization to large catalogs," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 2023, pp. 631–639.