DITTO: Data-efficient and Fair Targeted Subset Selection for ASR Accent Adaptation

Suraj Kothawade[1]*, **Anmol Mekala**[2]*, **Chandra Sekhara D**[2], **Mayank Kothyari**[2], **Rishabh Iyer**[1], **Ganesh Ramakrishnan**[2], **Preethi Jyothi**[2]

[1] The University of Texas at Dallas, Dallas, USA
[2] Indian Institute of Technology Bombay, Mumbai, India

Abstract

State-of-the-art Automatic Speech Recognition (ASR) systems are known to exhibit disparate performance on varying speech accents. To improve performance on a specific target accent, a commonly adopted solution is to finetune the ASR model using accent-specific labeled speech. However, acquiring large amounts of labeled speech for specific target accents is challenging. Choosing an informative subset of speech samples that are most representative of the target accents becomes important for effective ASR finetuning. To address this problem, we propose DITTO (**D**ata-efficient and fa**I**r **T**argeted subse**T** selecti**O**n) that uses Submodular Mutual Information (SMI) functions as acquisition functions to find the most informative set of utterances matching a target accent within a fixed budget. An important feature of DITTO is that it supports fair targeting for multiple accents, *i.e.* it can automatically select representative data points from multiple accents when the ASR model needs to perform well on more than one accent. We show that DITTO is 3-5 times more label-efficient than other speech selection methods on the Indic-TTS and L2 datasets.

1 Introduction

State-of-the-art speech recognition systems have seen tremendous progress in the last few years, with end-to-end architectures becoming a default modeling choice. While end-to-end models yield impressive Word Error Rates (WERs) and work well for certain user populations (Rao et al., 2017; Chiu et al., 2018), they severely underperform when confronted with out-of-domain test utterances in target accents that are unseen or rarely seen during training (Feng et al., 2021; Koenecke et al., 2020).

A common solution (Shor et al., 2019; Sim et al., 2019) to address such mismatched settings is to adapt a well-trained, speaker-independent ASR model with a small amount of accent-specific target data to adapt models to the target setting. While these works propose different fine-tuning schedules that would be most beneficial given the limited amount of target data, the question of which utterances should be chosen in order to be transcribed and further used for fine-tuning has received far less attention. This is extremely important, since procuring and labeling accent-specific data is challenging and expensive. Awasthi *et. al.* (Awasthi et al., 2021) present a method to select sentences within a fixed budget that are most likely to induce ASR errors to record accented audio on, resulting in higher-quality personalized ASR models for target accents compared to random selection. However, they assume access to a small seed set of labeled utterances from the target speaker. We address a more realistic setting wherein we have access only to a limited number of *unlabeled* utterances from the target domain, and without access to accented speakers to read out the selected texts.

1.1 Our Contributions

In this work, we propose DITTO a data-efficient and fair targeted subset selection approach that makes use of a suite of submodular mutual information (SMI) functions (originally defined in (Iyer et al., 2021)). For a specific target accent, we are given access to a small number (*20* in our experiments) of unlabeled speech utterances, called the target (or query) set. We aim at identifying the most informative subset of speech utterances from a large unlabeled pool of diverse accents that best matches the target set. We procure the best matching subset by maximizing an SMI function instantiated using pairwise similarities between speech representations. We find DITTO to be an effective targeted subset selection technique for adapting ASR models in accents at multiple granularities - within Indian accents and accents around the world. DITTO uses a limited transcription budget, *i.e.*, just

*Equal contribution.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5810–5822
July 9-14, 2023 ©2023 Association for Computational Linguistics

around 20-35% of that of random. Furthermore, we show that DITTO can fairly select subsets that can cover multiple target accents using a facility location based SMI function.

2 Related Work

A number of works have studied subset selection for speech recognition. Wei et al. (2014a,b, 2013) use submodular function-based subset selection on generated transcripts to find a minimal set of ASR training data and Wu et al. (2007) use an entropy measure for the same. Asami et al. (2015) employ a joint Kullback-Leibler divergence-based subset selection on out-of-domain samples for ASR adaptation across acoustic characteristics such as speaker, noise and recording devices. Similarly, Liu et al. (2015) study subset selection to obtain low-vocabulary speech corpora for ASR, while Kirchhoff and Bilmes (2014) use a submodular approach for data selection in machine translation. Many recent papers (Yuan et al., 2019; Syed et al., 2017) have studied uncertainty and gradient based approaches for active learning to reduce the transcription time for ASR models, while Hamanaka et al. (2010) use a committee-based active learning method to select speech utterances.

A number of approaches have studied adaptation to atypical speech patterns like accented and dysarthic speech, such as (Shor et al., 2019) and (Tomanek et al., 2021) which fine-tune a subset of layers using labeled data from targeted accents. Sun et al. (2018) employ domain adversarial training to adapt across accents. Awasthi et al. (2021) tries addressing a problem that corresponds exactly to the reverse of our setting by trying to determine the sentences a model is most error-prone to, and recording utterances for them. While this can be effective for user-driven personalization, our method is suited to settings in which we have fixed speech utterances, and the only actionable item for us is to transcribe a subset of them. All these approaches need data specifically from the target domain to be labeled toward use for training/fine-tuning.

Finally, a number of recent works on data selection have leveraged the submodular mutual information functions used in this work for targeted subset selection. Kaushal et al. (2020) employ the SMI functions for query focused and privacy-preserving summarization, while Kothawade et al. (2022) utilize the SMI functions for improving the model performance on targeted slices. Recently, Kothawade et al. (2021) proposed an active learning approach using the SMI functions for rare classes, redundancy, and OOD data.

3 Submodular Mutual Information (SMI) Functions

Submodular Functions: We let $\mathcal{U}$ denote the *ground-set* of n data points $\mathcal{U} = \{1, 2, 3, ..., n\}$ and a set function $f : 2^{\mathcal{U}} \rightarrow \Re$. The function f is submodular (Fujishige, 2005) if it satisfies the diminishing marginal returns, namely $f(j|\mathcal{X}) = f(\mathcal{X} \cup j) - f(\mathcal{X}) \geq f(j|\mathcal{Y})$ for all $\mathcal{X} \subseteq \mathcal{Y} \subseteq \mathcal{U}, j \notin \mathcal{Y}$. Submodularity ensures that a greedy algorithm achieves bounded approximation factor when maximized (Nemhauser et al., 1978).
Submodular Mutual Information (SMI): Given a set of items $\mathcal{S}, \mathcal{T} \subseteq \mathcal{U}$, the submodular mutual information (SMI) (Gupta and Levin, 2020; Iyer et al., 2021) is defined as $I_f(\mathcal{S}; \mathcal{T}) = f(\mathcal{S}) + f(\mathcal{T}) - f(\mathcal{S} \cup \mathcal{T})$. Intuitively, this function measures the similarity between $\mathcal{T}$ and $\mathcal{S}$ and we refer to $\mathcal{T}$ as the targeted set. In the setting considered in this paper, the set $\mathcal{T}$ (target set, also called query set) consists of a small set of unlabeled utterances from an accent, and $\mathcal{U}$ is a large unlabeled set of utterances from multiple accents. To find an optimal subset given a target set $\mathcal{T}$, we can define $g_{\mathcal{T}}(\mathcal{S}) = I_f(\mathcal{S}; \mathcal{T})$, $\mathcal{S} \subseteq \mathcal{U}$ and maximize the same. Using a greedy algorithm, these submodular functions can be efficiently optimized within an approximation factor (1-1/e) of the global maximum.

3.1 SMI functions used in DITTO

We use the SMI functions recently introduced in (Iyer et al., 2021) and their extensions introduced in (Kaushal et al., 2020; Kothawade et al., 2022). For any two data points $i \in \mathcal{U}$ and $j \in \mathcal{T}$, let s_{ij} denote the similarity between them.

Graph Cut MI: The submodular mutual information (SMI) instantiation of graph-cut (GCMI) is defined as (Kothawade et al., 2022; Iyer et al., 2021):

$$I_f(\mathcal{S}; \mathcal{T}) = 2 \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{T}} s_{ij} \tag{1}$$

Since maximizing GCMI maximizes the joint pairwise sum with the query set, it will lead to a summary similar to the query set Q. GCMI models only query-relevance and does not select based on diversity (Kothawade et al., 2022).

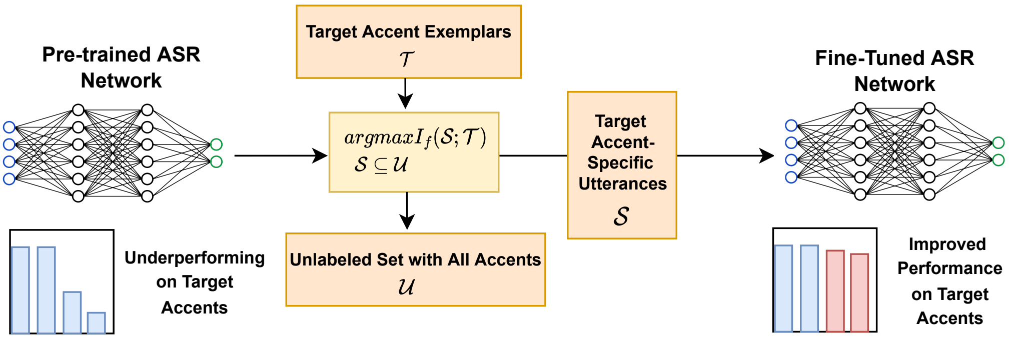
Pre-trained ASR Network
Target Accent Exemplars
$\mathcal{T}$
$argmax I_f(\mathcal{S};\mathcal{T})$
$\mathcal{S} \subseteq \mathcal{U}$
Target Accent-Specific Utterances
$\mathcal{S}$
Fine-Tuned ASR Network
Underperforming on Target Accents
Unlabeled Set with All Accents
$\mathcal{U}$
Improved Performance on Target Accents

Figure 1: ASR Accent Adaptation using DITTO.

Facility Location MI: The Facility Location Mutual Information (FLMI) function (Kothawade et al., 2022) takes the expression:

$$I_f(\mathcal{S};\mathcal{T}) = \sum_{i\in\mathcal{T}} \max_{j\in\mathcal{S}} s_{ij} + \sum_{i\in\mathcal{S}} \max_{j\in\mathcal{T}} s_{ij} \quad (2)$$

FLMI jointly models representation and query-relevance. It measures a bidirectional similarity between representation of data points that are the most relevant to the query set, and vice versa.

4 DITTO: Our Data-efficient and Fair Targeted Subset Selection Method

In this section, we discuss DITTO our data-efficient and fair targeted subset selection method for ASR accent adaptation. We show that DITTO can select fair and target-relevant subsets, which is critical for fine-tuning ASR models on one or more accents. The main idea of our method is to instantiate a submodular mutual information (SMI) function using appropriate similarity kernels in order to jointly optimize it for targeting and fairness. We summarize our method in Algorithm 1 and illustrate it in Fig. 1.

Concretely, we are provided a few unlabeled utterances from the accent (a target set $\mathcal{T}$) which we would like the ASR model M to be adapted to. The goal is to select the most informative subset $\mathcal{S}$ *with respect to* a target $\mathcal{T}$ from a large corpus $\mathcal{U}$ of unlabeled data, called the *ground set*. We are given a budget constraint, which is a constraint on the total time of the selected utterances. This corresponds to the transcription budget, since the selected utterances need to be later transcribed by a human.

We begin with extracting accent feature representations of the unlabeled set $\mathcal{U}$ and the target set $\mathcal{T}$; we subsequently discuss the feature

Algorithm 1 DITTO for Adapting ASR model M

```
Require: Target $\mathcal{T}$, Budget $B$, SMI function type $f$, large unlabeled set $\mathcal{U}$, Accent representation model $\mathcal{F}$, ASR model $M$ with parameters $\theta$
1: $\mathcal{E}_\mathcal{T} \leftarrow \mathcal{F}(\mathcal{T})$ $\{\mathcal{E}_\mathcal{T} \in \mathbb{R}^{|\mathcal{T}|\times D}\}$
2: $\mathcal{E}_\mathcal{U} \leftarrow \mathcal{F}(\mathcal{U})$ $\{\mathcal{E}_\mathcal{U} \in \mathbb{R}^{|\mathcal{U}|\times D}\}$
3: $\mathcal{X} \leftarrow$ SIMILARITY($\mathcal{E}_\mathcal{T}, \mathcal{E}_\mathcal{U}$) $\{\mathcal{X} \in \mathbb{R}^{|\mathcal{T}|\times|\mathcal{U}|}\}$
4: Define an SMI function $g_\mathcal{T}(\mathcal{S}) = I_f(\mathcal{S};\mathcal{T})$ using $\mathcal{X}$
5: $\mathcal{S} \leftarrow \underset{\mathcal{S}\subseteq\mathcal{U}, c(\mathcal{S})\leq B}{\text{argmax}}\ g_\mathcal{T}(\mathcal{S})$
   {Greedy maximization of $g$ to select a subset $\mathcal{S}$}
6: $D \leftarrow$ Transcribe utterances in $\mathcal{S}$
7: $\hat{\theta} \leftarrow$ Fine-tune ASR model $M$ on $D$
8: Return Fine-tuned model $\mathcal{M}$ with updated parameters $\hat{\theta}$.
```

representation in Sec. 5. Next, we compute a similarity matrix $\mathcal{X}$, which is an RBF kernel containing pairwise similarities $\mathcal{X}_{ij}$ between all data points in $i \in \mathcal{T}$ and $j \in \mathcal{U}$. We use $\mathcal{X}$ to instantiate one of the SMI functions $I_f(\mathcal{S};\mathcal{T})$ discussed in Sec. 3. Specifically, we optimize $g_\mathcal{T}(\mathcal{S}) = I_f(\mathcal{S};\mathcal{T})$ for $\mathcal{S} \subseteq \mathcal{U}$ subject to the cardinality constraint $c(\mathcal{S}) \leq B$, where c corresponds to the duration (in seconds) of the specific utterance and B is the time budget. We use the greedy algorithm (Mirzasoleiman et al., 2015; Nemhauser et al., 1978; Lin and Bilmes, 2010) with memoization (Iyer and Bilmes, 2019) and with a knapsack constraint on the optimization. Specifically, given the current set $\mathcal{S}$, we select the item $i = \text{argmax}_{j\in\mathcal{U}\setminus\mathcal{S}}\, g_\mathcal{T}(j|\mathcal{S})$, with the stopping criterion as $c(\mathcal{S}) \leq B$. Once, we obtain the set $\mathcal{S}$ as the solution of this optimization problem, we obtain $\mathcal{S}$'s transcriptions from a human, and fine-tune the ASR model using $\mathcal{S}$ and its labels.

Scalability of DITTO: The selection time of

DITTO is dominated by the instantiation and maximization of the SMI function. Since all SMI functions used in this work are graph based, they require the computation of a similarity kernel. Hence, the main components that contribute towards the time complexity of DITTO are the similarity kernel computation and the greedy maximization. The FLMI and GCMI functions require a $t \times u$ similarity matrix, where $t = |T|$ is the number of points in the target set and $u = |U|$ is the number of points in the unlabeled ground set. This leads to a $O(tu)$ complexity for computing the kernel. Given a selection budget of B, the time complexity of the greedy maximization for FLMI and GCMI is $O(tuB)$, which is linear in budget and ground set sizes.

5 Experimental Setup

5.1 Datasets

We experiment with adapting ASR models on two public datasets, *viz.*, *IndicTTS* and *L2-Arctic*, containing English speech in various non-native accents. IndicTTS (Vignesh et al., 2016) consists of 35K utterances from 8 Indian speakers, each with a different accent depending on their native language: Gujarati (GUJ) 9.2% of samples, Kannada (KAN) 9.4%, Tamil (TAM) 15%, Malayalam (MAL) 10.1%, Hindi (HIN) 10.1%, Rajasthani (RAJ) 9.5%, Assamese (ASM) 16.7% and Manipuri (MAN) 20.1%. L2-Arctic (Zhao et al., 2018) has 18K samples of accented English speech from 24 speakers spanning six non-native accents: Hindi (HIN), Vietnamese (VTN), Chinese (CHN), Korean (KOR), Arabic (ARB) and Spanish (ESP). The distribution among the accents is uniform for this datasets, with all represented equally.

Feature representation: Each utterance is represented as a 39-dimensional vector of MFCC coefficients averaged over the duration of the utterance.

5.2 ASR Model Description and Fine-tuning Details

Following (Awasthi et al., 2021), our pre-trained model is based on the QuartzNet-15x5 (Kriman et al., 2020) architecture. It is trained on LibriSpeech (Panayotov et al., 2015) for 400 epochs using the CTC-loss (Graves et al., 2006) and yields a Word Error Rate (WER) of 3.90 on the test-clean split of LibriSpeech. The QuartzNet-15x5 architecture is fully convolutional with residual connections. This model is fine-tuned with our selected targeted subsets $\mathcal{S}$ of accented speech to minimize CTC loss using the NovoGrad optimizer (Ginsburg et al., 2019) for 100 epochs with a batch size of 16, a linearly decaying learning rate of 10^{-5} and early stopping based on the dev set. In all our experiments, we report results averaged over three runs using three different seeds and report error bars in all plots. We used an NVIDIA GTX 1080 Ti GPU for all runs.

6 Experimental Procedure and Results

We use a transcription budget of 20.5 minutes for single-accent targeting and 41 minutes when an accent pair is targeted. The average uttterance durations are 4.92s in IndicTTS and 3.6s in L2-Arctic, thus these budgets come out to 250 and 500 samples on IndicTTS and 340 and 780 samples on L2-Arctic respectively.

In our proposed method, we use the approach outlined in Algorithm 1, with the SMI function $\mathcal{I}_f$ set as one of the FLMI or GCMI functions. We consider them since they are computationally efficient (see Sec. 4), and model different characteristics in their selections. As discussed in Sec. 3.1, GCMI models *only* query-relevance, while FLMI models *both* query-relevance and diversity. As we shall see in Sec. 6.2 and Sec. 6.3, GCMI is an apt choice for targeting in some scenarios, whereas FLMI outperforms all methods when fairness and diversity are to be jointly modeled with targeting.

6.1 Baseline selection approaches

We compare the performance of DITTO with a wide-array of standard selection techniques designed for subset selection in speech. In our results, we track the improvements in WER over the performance of the pretrained ASR model (without any finetuning), denoted as “Pre". For all single/multiple accent experiments, we compare DITTO with the following selection approaches:

Random: Selecting utterances randomly from the different accents of the ground set. The selection distributions will roughly match the ground set.

phone-diverse (“PhoneDiv"): Following an approach from (Awasthi et al., 2021), we select a phonetically rich set from the set of generated transcripts of our baseline ASR model on the utterances. We define a diminishing returns submodular set-scoring function that penalizes over-represented phones. Similar to optimizing SMI functions, greedy optimization (Mirzasoleiman et al., 2015)

Eval	Pre	Random	Entropy	PhoneDiv	FL	LogDet	GCMI	FLMI
ASM (T%)	27.1	21.5 (16.7%)	26.4 (2.2%)	22.7 (9.3%)	21.7 (16.1%)	22.2 (13.6%)	**18.1** (100%)	18.7 (100%)
GUJ (T%)	13.7	11.0 (15.6%)	11.2 (28.8%)	11.1 (1.5%)	10.9 (22.6%)	10.7 (24.6%)	9.7 (100%)	**9.4** (100%)
HIN (T%)	11.1	9.7 (9.8%)	9.7 (11.0%)	10.4 (1.7%)	9.5 (10.1%)	9.7 (12.1%)	**8.5** (100%)	9.2 (100%)
KAN (T%)	18.7	15.3 (9.8%)	15.4 (14.4%)	17.8 (2.3%)	15.7 (7.1%)	16.0 (2.5%)	**12.8** (100%)	13.1 (100%)
MAL (T%)	19.5	16.8 (12.2%)	16.9 (8.7%)	18.8 (1.7%)	18.6 (8.6%)	18.3 (4.1%)	13.9 (100%)	**13.6** (98.6%)
MAN (T%)	53.1	44.8 (13.3%)	48.4 (5.3%)	42.5 (79.9%)	44.5 (10.8%)	43.3 (20.6%)	**39.8** (100%)	39.9 (100%)
RAJ (T%)	21.9	16.9 (7.5%)	16.3 (11.5%)	18.2 (1.9%)	16.9 (9.4%)	16.4 (9.4%)	**14.3** (100%)	14.4 (100%)
TAM (T%)	12.5	11.9 (15.1%)	11.7 (18.0%)	12.2 (1.6%)	11.9 (15.3%)	11.9 (13.0%)	**11.1** (100%)	11.5 (100%)

Table 1: Single-target accent adaptation in IndicTTS on a 1230s budget.

Eval	Pre	Random	Entropy	PhoneDiv	FL	LogDet	GCMI	FLMI
ARB (T%)	24.3	23.1 (14.7%)	23.7 (16.6%)	23.3 (15.2%)	23.4 (15.1%)	23.4 (8.1%)	22.8 (58.6%)	**20.8** (98.7%)
CHN (T%)	30.7	28.3 (19.4%)	28.5 (18.1%)	28.3 (20.4%)	28.2 (16.8%)	28.3 (14.9%)	25.8 (70.4%)	**25.1** (99.6%)
HIN (T%)	18.1	17.1 (15.2%)	18.6 (7.7%)	16.4 (14.1%)	17.3 (14.2%)	16.5 (28.7%)	15.5 (48.0%)	**15.4** (92.9%)
KOR (T%)	19.1	17.7 (17.5%)	18.4 (15.7%)	18.2 (16.0%)	18.2 (17.3%)	17.6 (23.5%)	17.0 (84.3%)	**16.5** (98.9%)
ESP (T%)	23.4	22.2 (16.6%)	22.5 (23.5%)	22.4 (17.2%)	23.3 (17.9%)	22.4 (7.0%)	21.3 (89.2%)	**20.8** (99.6%)
VTN (T%)	37	33.7 (16.6%)	35.5 (18.4%)	34.5 (17.1%)	34.2 (18.6%)	34.9 (17.7%)	**31.2** (97.1%)	31.6 (100%)

Table 2: Single-target accent adaptation in L2-Arctic on a 1230s budget.

of this function gives us our selected subset. The function score($\mathcal{S}$) is defined as follows, with $\mathcal{S}$ denoting a set of sentences, $\mathcal{P}$ denoting the set of phones, τ a hyperparameter for the penalization (set to 500 in our experiments), $n_\pi(\mathcal{S})$ denoting the number of utterances of phone π in $\mathcal{S}$:

$$\text{score}(\mathcal{S}) = \sum_{\pi}(1 - \exp(-n_\pi(\mathcal{S})/\tau)) \quad (3)$$

Maximum ASR entropy ("Entropy"): This method is from Riccardi and Hakkani-Tur (2005) and focuses on selecting utterances that the baseline ASR model is most uncertain about. We score each utterance by finding the entropy across frame-level phone predictions and averaging across frames. We then pick the highest scoring utterances within our budget duration. The score of an utterance $\mathcal{C}$ is defined as follows, with $\mathcal{F}$ denoting the frames of the utterance $\mathcal{C}$, $p_f(\pi)$ denoting the ASR model's softmax value on phone-π of frame-f.

$$\text{score}(\mathcal{C}) = \frac{1}{|\mathcal{F}|}\sum_{\text{frame} f\in\mathcal{F}}\sum_{\pi} -p_f(\pi)log(p_f(\pi)) \quad (4)$$

We also use two submodular functions that are well-known for subset selection tasks. Namely, Facility Location and Log Determinant functions.

Facility Location ("FL"): The facility location function is known to select a representative subset and has been extensively used for speech data subset selection tasks (Wei et al., 2014a, 2013). Using the same notation as in Sec. 4, where $\mathcal{S}$ denotes the subset of utterances to be selected from the unlabeled set $\mathcal{U}$, the FL function is defined as:

$$f(\mathcal{S}) = \sum_{i\in\mathcal{U}}\max_{j\in\mathcal{S}}\mathcal{X}_{ij} \quad (5)$$

Log Determinant ("LogDet"): The log determinant function models diversity and is crucial for determinantal point processes (DPPs). The LogDet function is defined as follows:

$$f(\mathcal{S}) = \text{Log Det}(\mathcal{X}_{\mathcal{S}}) \quad (6)$$

where, Det(.) is the determinant, and $\mathcal{X}_{\mathcal{S}}$ denotes the rows and columns of the similarity matrix instantiated with elements in $\mathcal{S}$.

For fair evaluation, the FL and LogDet functions are optimized using the same greedy strategy (Mirzasoleiman et al., 2015) as the SMI functions used in DITTO. Note that FL and LogDet functions are computationally expensive since they require the computation of a $O(n^2)$ similarity matrix, as opposed to the SMI functions, which can be optimized in linear time.

6.2 Targeted Subset Selection for Single-accents

In this section, we analyze the performance of DITTO for procuring a subset that is targeted for

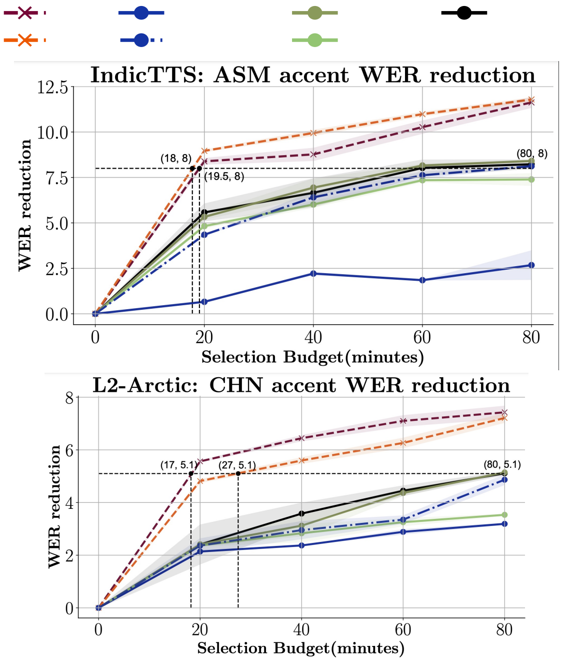

Figure 2: WER reductions across a range of budgets, targeting Assamese (from IndicTTS) and Chinese (from L2-Arctic) accents.

a single accent, followed by fine-tuning an ASR model using the selected targeted subset. For evaluation, we study the *Word Error Rate* (WER) of the targeted accent by evaluating the fine-tuned model on a held-out test set containing utterances of the targeted accent. Along with the WER, we also report a *Targeted percentage* ($\mathbf{T}\%$) that denotes the ratio of utterances selected from the targeted accent given the total budget.

We conduct extensive experiments on IndicTTS (Vignesh et al., 2016) and L2-Arctic (Zhao et al., 2018) datasets (see Sec. 5.1 for details) by targeting all accents in both datasets, one accent at a time. For each accent (around 4.3K samples in IndicTTS and 3K samples in L2-Arctic), we create data splits by partitioning 70% of the data for the unlabeled set ($\mathcal{U}$) and a small target set $\mathcal{T}$ of size 20. Of the remaining 30%, we create a test set from 27% and use 50 samples from the 3% as the finetuning dev set. In L2 Arctic, which has equal samples from each speaker of an accent, we ensure an equal split across the speakers in our accent-specific query, test and dev sets.

We present the targeted subset selection and fine-tuning results for the IndicTTS dataset in Tab. 1 and for L2-Arctic in Tab. 2. We observe that the SMI functions, GCMI and FLMI outperform other methods in terms of WER for all target accents. This is due to the fact that the SMI functions are able to identify utterances from the target accent almost to perfection in most cases. Interestingly, GCMI performs better than FLMI on IndicTTS in 6 out of 8 accents due to its high predilection towards query-relevance. On the other hand, GCMI performs worse than FLMI (although better than other methods) in terms of WER on the L2-Arctic dataset. This is because FLMI is significantly better in terms of targeting in comparison to GCMI and other methods. Note that IndicTTS is simpler since it contains data from only one speaker per accent, whereas L2-Arctic has comparatively more complex acoustics as it contains data from multiple speakers for each accent. We believe that the representation modeling capability and bidirectional similarity of FLMI allows for a better targeting ability on complex datasets like L2-Arctic. On the other hand, for datasets with lower acoustic complexity like IndicTTS: GCMI, which performs query-relevance, works well. We also present the variation in WER improvements across a range of budgets in Fig. 2 for accents picked from each dataset. The horizontal lines marked indicate the how much budget each method needed for the same WER gain. For Assamese (ASM) we see that Random needs 80 minutes to improve by 8, while FLMI and GCMI do it in 18 and 19.5 mins respectively. For Chinese (CHN) we observe: for 5.1 gain, Random needs 80 minutes, while FLMI and GCMI need only 17 and 27 minutes respectively. The SMI functions are thus 3-5 times as label efficient than random.

6.3 Fair Targeted Subset Selection for Multiple-accents

Another important setting of high practical value is that of adapting an ASR model for multiple targeted accents. In a real-world scenario, practitioners may want to improve the performance of the ASR model on accents that are under-performing. In another deployment scenario, one may need to fine-tune the ASR model on multiple accents in order to deploy it in a region where the population speaks in more than one accent. To tackle such scenarios, an ideal selection function would model fairness and select approximately equal number of utterances from each accent. To study this, we evaluate the performance of DITTO for targeting pairs of accents, followed by fine-tuning the ASR model on the selected targeted subset. For evalua-

Eval	Pre	Random	Entropy	PhoneDiv	FL	LogDet	GCMI	FLMI
ASM-WER (**T**%)	27.1	20.4 (17.8%)	24.8 (1.9%)	20.7 (17.7%)	20.1 (16.2%)	21.0 (15.1%)	21.1 (24.7%)	**19.7** (50.6%)
MAL-WER (**T**%)	19.5	16.2 (10.9%)	16.8 (8.6%)	18.4 (3.8%)	18.7 (8.7%)	17.5 (4.4%)	**14.0** (75.3%)	14.7 (49.4%)
Targeted Fairness (TF)	-	0.08	0.01	0.03	0.06	0.03	0.74	**1**
Avg. WER	23.3	18.3	20.8	19.5	19.4	19.2	17.6	**17.2**
MAL-WER (**T**%)	19.5	16.2 (10.9%)	16.8 (8.6%)	18.4 (3.8%)	18.7 (8.7%)	17.5 (4.4%)	15.7 (7.6%)	**15.0** (21.9%)
RAJ-WER (**T**%)	21.9	16.2 (8.5%)	15.8 (12.4%)	17.2 (4.0%)	15.9 (8.9%)	16.1 (7.5%)	**13.3** (92.4%)	14.3 (78.1%)
Targeted Fairness (TF)	-	0.04	0.04	0.01	0.03	0.01	0.28	**0.68**
Avg. WER	20.7	16.2	16.3	17.8	17.3	16.8	**14.5**	14.6
ASM-WER (**T**%)	27.1	20.4 (17.8%)	24.8 (1.9%)	20.7 (17.7%)	20.1 (16.2%)	21.0 (15.1%)	25.6 (0.0%)	**21.2** (25.5%)
RAJ-WER (**T**%)	21.9	16.2 (8.5%)	15.8 (12.4%)	17.2 (4.0%)	15.9 (8.9%)	16.1 (7.5%)	**13.3** (100%)	14.3 (74.5%)
Targeted Fairness (TF)	-	0.06	0.01	0.03	0.06	0.05	0	**0.76**
Avg. WER	24.5	18.3	20.3	19	18	18.6	19.5	**17.8**

Table 3: Targeting Assamese, Rajasthani and Malayalam accents in pairs on a 2460s budget (IndicTTS). For definitions of **T**% and Targeted Fairness (TF), please refer to Sec. 6.2 and Sec. 6.3.

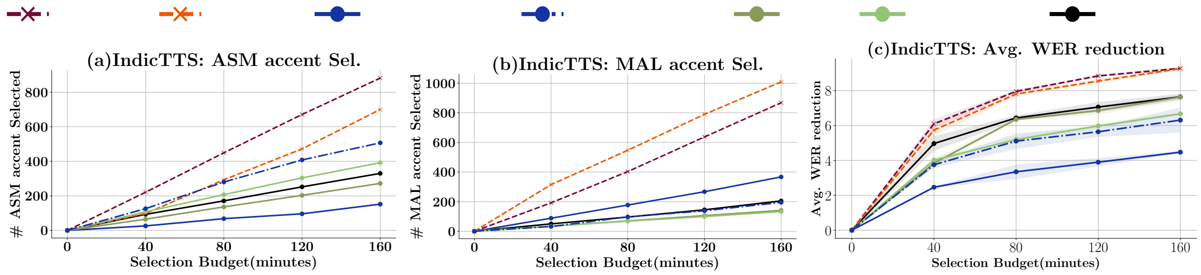

Figure 3: Variation in selections and average WER across a range of budgets, targeting ASM and MAL together.

Eval	Pre	Random	Entropy	PhoneDiv	FL	LogDet	GCMI	FLMI
ARB-WER (**T**%)	24.3	22.9 (15.4%)	23.4 (17.2%)	22.5 (14.1%)	23.4 (15.2%)	22.9 (9.9%)	21.0 (44.3%)	**20.3** (63.5%)
CHN-WER (**T**%)	30.7	27.1 (17.5%)	28.3 (19.0%)	27.7 (19.6%)	27.5 (17.2%)	27.8 (13.9%)	26.3 (31.5%)	**26.1** (32.7%)
Targeted Fairness (TF)	-	0.11	0.13	0.11	0.1	0.06	0.56	**0.83**
Avg. WER	27.5	25	25.8	25.1	25.4	25.4	23.6	**23.2**
CHN-WER (**T**%)	30.7	27.1 (17.5%)	28.3 (19.0%)	27.7 (19.6%)	27.5 (17.2%)	27.8 (13.9%)	25.9 (30.3%)	**25.3** (67.3%)
VTN-WER (**T**%)	37	32.8 (16.8%)	34.4 (16.2%)	33.6 (17.3%)	33.2 (17.5%)	33.7 (15.5%)	33.9 (15.5%)	**32.2** (31.2%)
Targeted Fairness (TF)	-	0.12	0.12	0.14	0.12	0.09	0.19	**0.84**
Avg. WER	33.8	30	31.4	30.6	30.4	30.8	29.9	**28.8**
ARB-WER (**T**%)	24.3	22.9 (15.4%)	23.4 (17.2%)	22.5 (14.1%)	23.4 (15.2%)	22.9 (9.9%)	21.3 (40.8%)	**20.3** (74.2%)
VTN-WER (**T**%)	37	32.8 (16.8%)	34.4 (16.2%)	33.6 (17.3%)	33.2 (17.5%)	33.7 (15.5%)	**33.8** (20.8%)	34.1 (19.9%)
Targeted Fairness (TF)	-	0.1	0.11	0.1	0.11	0.06	0.34	**0.59**
Avg. WER	30.6	27.8	28.9	28	28.3	28.3	27.6	**27.2**

Table 4: Targeting Vietnamese, Arabic and Chinese accents in pairs on a 2460s budget (L2-Arctic).

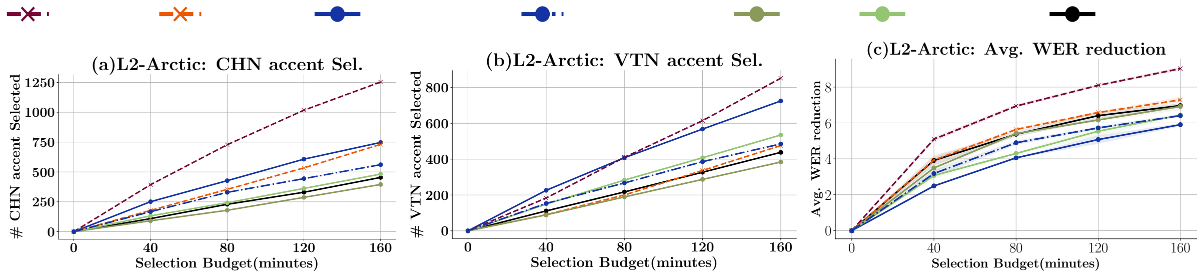

Figure 4: Variation in selections and average WER across a range of budgets, targeting CHN and VTN together.

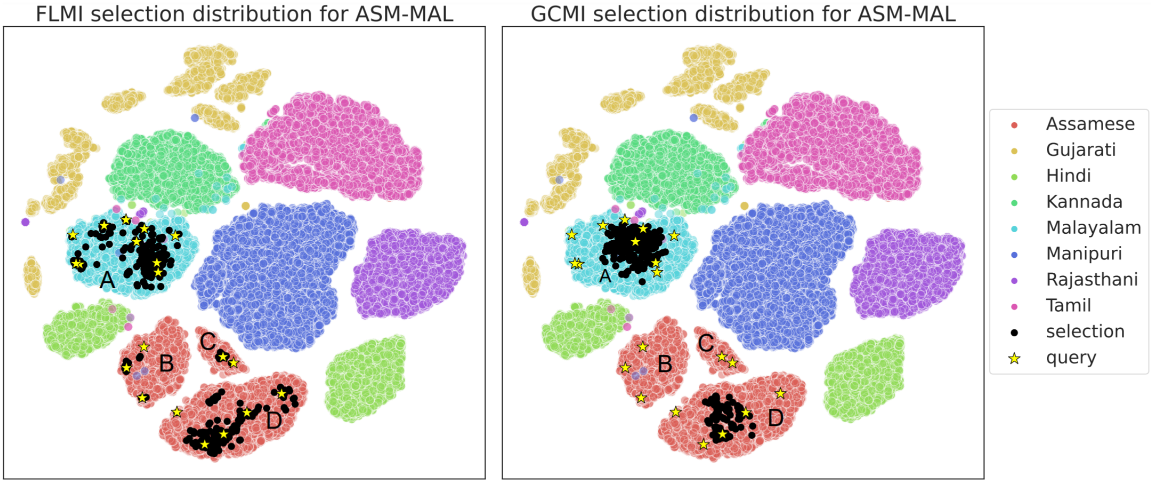

Figure 5: t-SNE visualising MFCC features of selections on the full IndicTTS dataset: comparing FLMI and GCMI. FLMI's selections are representative of the query and spread over all clusters A, B, C and D. GCMI is biased towards the bigger cluster centers: it does not select from clusters B and C at all (thus selecting a lower Assamese fraction) and selects dense, unrepresentative clusters from the centers of A and D.

tion, we study the WER and average WER for both the targeted accents by evaluating the fine-tuned ASR model on separate individual held-out test sets containing utterances from each of the targeted accents. Similar to the single accent experiments, we also report the *Targeted percentage* ($\mathbf{T}\%$) for each targeted accent. In addition, we report a *Targeted Fairness* (**TF**) score for the accent-pair, which is computed as a product of the targeted percentages of both the targeted accents. We multiply the final score by 4 to obtain a TF score of 1 when the selected subset perfectly targets both accents, *i.e.* it achieves 50% targeted percentage for both the targeted accents.

For our experiments, we consider pairs of the three worst-performing accents as target accent pairs from IndicTTS and L2-Arctic datasets. We present results for three target accent pairs from IndicTTS in Tab. 3: i) Assamese and Malayalam, ii) Malayalam and Rajasthani, and iii) Assamese and Rajasthani. We use data splits created in Sec. 6.2: ground and test sets remain the same, whereas query sets for the accent pairs here are made by taking 10 from each accent from the accent-specific query sets of Sec. 6.2.

We observe that the SMI functions (GCMI and FLMI) outperform other methods in terms of the Avg. WER and the TF score. Interestingly, we also observe that GCMI often favors targeting a single accent: MAL when MAL and ASM are targeted, RAJ when RAJ and MAL are targeted and RAJ from RAJ and ASM. Due to this, GCMI obtains a lower TF score than FLMI. It is worth noting that FLMI achieves a TF score of as high as 1 (see Tab. 3's Asm-Mal section) due to its ability to jointly model representation. We find that GCMI tends to favor a particular accent A due to higher pairwise similarity values $\mathcal{X}_A$ between utterances belonging to accent A in comparison to accent B. In Fig. 3, for the ASM-MAL accent pair, we illustrate the Avg. WER improvement and duration of utterances selected from both the accents across a wide range of budgets. Notably, we observe that FLMI continues to select fairly for both accents, while GCMI favors MAL accent over ASM.

To compare the Targeted Fairness between FLMI and GCMI, we visualize a t-SNE plot of the IndicTTS dataset embedded using MFCC features in Fig. 5. As shown in the legend of Fig. 5, each color represents a particular accent and the selected data points are denoted in black. The query data points from ASM and MAL accents are shown by yellow stars. We observe that the data points selected by FLMI are representative of the query, as they are spread well across MAL (cluster A) and ASM (clusters B, C and D). On the other hand, data points selected by GCMI are mainly concentrated in the bigger cluster centers of MAL (cluster A) and ASM (cluster D), while completely missing clusters B and C. This is again a consequence of the fact that FLMI jointly models representation and query-relevance whereas GCMI focuses only on query-relevance.

We conduct a similar analysis for the L2-Arctic

Eval	Pre	Random	Entropy	PhoneDiv	FL	LogDet	GCMI	FLMI
ASM-WER (**T**%)	27.1	20.4 (17.8%)	24.8 (1.9%)	20.7 (17.7%)	20.1 (16.2%)	21.0 (15.1%)	17.1 (97.8%)	**17.0** (92.6%)
MAN-WER (**T**%)	53.1	42.5 (13.6%)	47.2 (3.8%)	40.7 (59.0%)	42.4 (12.9%)	40.7 (23.9%)	44.6 (2.2%)	**42.9** (7.4%)
Targeted Fairness (TF)	-	0.1	0	0.42	0.08	0.14	0.09	**0.27**
Avg. WER	40.1	31.4	36	30.7	31.2	30.8	30.8	**30**
MAN-WER (**T**%)	53.1	42.5 (13.6%)	47.2 (3.8%)	40.7 (59.0%)	42.4 (12.9%)	40.7 (23.9%)	48.9 (0.0%)	**47.2 (2.7%)**
RAJ-WER (**T**%)	21.9	16.2 (8.5%)	15.8 (12.4%)	17.2 (4.0%)	15.9 (8.9%)	16.1 (7.5%)	**13.2 (100%)**	13.7 (97.3%)
Targeted Fairness (TF)	-	0.05	0.02	0.09	0.05	0.07	0	**0.11**
Avg. WER	37.5	29.4	31.5	29	29.2	28.4	31	**30.4**

Table 5: Targeting Assamese, Rajasthani and Manipuri accents in pairs on a 2460s budget (IndicTTS)

dataset. We present the results for pairs from three bottom performing target accents from L2-Arctic in Tab. 4: i) Arabic and Chinese, ii) Arabic and Vietnamese, and iii) Chinese and Vietnamese. Consistently, the SMI functions outperform other methods in terms of Avg. WER and TF score. Evidently, FLMI performs the best across all accent pairs. In Fig. 4, for the CHN-VTN accent pair, we demonstrate the Avg. WER improvement and duration of utterances selected from both the accents across a wide range of budgets. We observe that FLMI achieves the highest Avg. WER and selects the most number of utterances from both target accents, proving it can achieve robust targeting and fairness.

7 Conclusion

In this work, we propose DITTO, a data efficient and fair targeted subset selection method for ASR accent adaptation. DITTO utilizes submodular mutual information (SMI) functions to find representative speech utterances that belong to the target accent within a limit budget. We show that SMI functions consistently outperform other methods for targeting. We also demonstrate that DITTO is capable of targeting multiple accents fairly, which can be beneficial for deploying ASR models in regions with populations that speak in more than one accent.

8 Limitations

Similar to the limitations of existing selection methods, our method needs a reasonable feature embedding for accent representation in order to effectively target accents. MFCC features are not the best choice to represent accent information. Some accents may be more difficult to represent than others. This also lowers fairness scores for such accents. For instance, in one of our experiments where Manipuri accent was paired with Rajasthani or Assamese accents, we observe that acquiring a fair subset using any selection strategy is challenging (see Tab. 5). Although, FLMI was able to achieve a higher TF score than others, it was relatively lower than other accent pairs (see Tab. 3 and Tab. 4). This is due to the fact that the pairwise similarity scores of utterances within the Manipuri accent are lower than other accents. The lower pairwise similarity scores lead to lower marginal gains during greedy maximization and are a consequence of poor feature representations due to insufficient information being encoded about the Manipuri accent. On another note, a risk associated with the targeting ability of DITTO is that it could be misused to create models that are unfair to certain populations. For future work, evaluating the performance of DITTO on larger datasets and other diverse settings (*e.g.* out-of-distribution accents) will be interesting.

9 Acknowledgments and Disclosure of Funding

This work is supported by an Amazon Research Awarded (AWA) awarded to Preethi Jyothi, Ganesh Ramakrishnan and Rishabh Iyer, and by the National Science Foundation under Grant No. IIS-2106937 awarded to Rishabh Iyer. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of Amazon or the National Science Foundation.

References

Taichi Asami, Ryo Masumura, Hirokazu Masataki, Manabu Okamoto, and Sumitaka Sakauchi. 2015. Training data selection for acoustic modeling via submodular optimization of joint kullback-leibler divergence. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Abhijeet Awasthi, Aman Kansal, Sunita Sarawagi, and

Preethi Jyothi. 2021. Error-driven fixed-budget asr personalization for accented speakers. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7033–7037. IEEE.

Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al. 2018. State-of-the-art speech recognition with sequence-to-sequence models. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4774–4778. IEEE.

Siyuan Feng, Olya Kudina, Bence Mark Halpern, and Odette Scharenborg. 2021. Quantifying bias in automatic speech recognition. *arXiv preprint arXiv:2103.15122*.

Satoru Fujishige. 2005. *Submodular functions and optimization*. Elsevier.

Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, Yang Zhang, and Jonathan M Cohen. 2019. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks. *arXiv preprint arXiv:1905.11286*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376.

Anupam Gupta and Roie Levin. 2020. The online submodular cover problem. In *ACM-SIAM Symposium on Discrete Algorithms*.

Yuzo Hamanaka, Koichi Shinoda, Sadaoki Furui, Tadashi Emori, and Takafumi Koshinaka. 2010. Speech modeling based on committee-based active learning. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4350–4353. IEEE.

Rishabh Iyer and Jeffrey Bilmes. 2019. A memoization framework for scaling submodular optimization to large scale problems. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2340–2349. PMLR.

Rishabh Iyer, Ninad Khargoankar, Jeff Bilmes, and Himanshu Asanani. 2021. Submodular combinatorial information measures with applications in machine learning. In *Algorithmic Learning Theory*, pages 722–754. PMLR.

Vishal Kaushal, Suraj Kothawade, Ganesh Ramakrishnan, Jeff Bilmes, Himanshu Asnani, and Rishabh Iyer. 2020. A unified framework for generic, query-focused, privacy preserving and update summarization using submodular information measures. *arXiv preprint arXiv:2010.05631*.

Katrin Kirchhoff and Jeff Bilmes. 2014. Submodularity for data selection in machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 131–141.

Allison Koenecke, Andrew Nam, Emily Lake, Joe Nudell, Minnie Quartey, Zion Mengesha, Connor Toups, John R Rickford, Dan Jurafsky, and Sharad Goel. 2020. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14):7684–7689.

Suraj Kothawade, Nathan Beck, Krishnateja Killamsetty, and Rishabh Iyer. 2021. Similar: Submodular information measures based active learning in realistic scenarios. *Advances in Neural Information Processing Systems*, 34.

Suraj Kothawade, Vishal Kaushal, Ganesh Ramakrishnan, Jeff Bilmes, and Rishabh Iyer. 2022. Prism: A rich class of parameterized submodular information measures for guided data subset selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10238–10246.

Samuel Kriman, Stanislav Beliaev, Boris Ginsburg, Jocelyn Huang, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, and Yang Zhang. 2020. Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6124–6128. IEEE.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *NAACL-HLT*, pages 912–920.

Yuzong Liu, Rishabh Iyer, Katrin Kirchhoff, and Jeff Bilmes. 2015. Svitchboard ii and fisver i: High-quality limited-complexity corpora of conversational english speech. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. 2015. Lazier than lazy greedy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE.

Kanishka Rao, Haşim Sak, and Rohit Prabhavalkar. 2017. Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 193–199. IEEE.

Giuseppe Riccardi and Dilek Hakkani-Tur. 2005. Active learning: Theory and applications to automatic speech recognition. *IEEE transactions on speech and audio processing*, 13(4):504–511.

Joel Shor, Dotan Emanuel, Oran Lang, Omry Tuval, Michael Brenner, Julie Cattiau, Fernando Vieira, Maeve McNally, Taylor Charbonneau, Melissa Nollstadt, et al. 2019. Personalizing asr for dysarthric and accented speech with limited data. *arXiv preprint arXiv:1907.13511*.

Khe Chai Sim, Petr Zadrazil, and Françoise Beaufays. 2019. An investigation into on-device personalization of end-to-end automatic speech recognition models. *arXiv preprint arXiv:1909.06678*.

Sining Sun, Ching-Feng Yeh, Mei-Yuh Hwang, Mari Ostendorf, and Lei Xie. 2018. Domain adversarial training for accented speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4854–4858. IEEE.

Ali Raza Syed, Andrew Rosenberg, and Michael Mandel. 2017. Active learning for low-resource speech recognition: Impact of selection size and language modeling data. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5315–5319. IEEE.

Katrin Tomanek, Françoise Beaufays, Julie Cattiau, Angad Chandorkar, and Khe Chai Sim. 2021. On-device personalization of automatic speech recognition models for disordered speech. *arXiv preprint arXiv:2106.10259*.

S Rupak Vignesh, S Aswin Shanmugam, and Hema A Murthy. 2016. Significance of pseudo-syllables in building better acoustic models for indian english tts. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5620–5624. IEEE.

Kai Wei, Yuzong Liu, Katrin Kirchhoff, Chris Bartels, and Jeff Bilmes. 2014a. Submodular subset selection for large-scale speech training data. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3311–3315. IEEE.

Kai Wei, Yuzong Liu, Katrin Kirchhoff, and Jeff Bilmes. 2013. Using document summarization techniques for speech data subset selection. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 721–726.

Kai Wei, Yuzong Liu, Katrin Kirchhoff, and Jeff Bilmes. 2014b. Unsupervised submodular subset selection for speech data. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4107–4111. IEEE.

Yi Wu, Rong Zhang, and Alexander Rudnicky. 2007. Data selection for speech recognition. In *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pages 562–565. IEEE.

Yang Yuan, Soo-Whan Chung, and Hong-Goo Kang. 2019. Gradient-based active learning query strategy for end-to-end speech recognition. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2832–2836. IEEE.

Guanlong Zhao, Sinem Sonsaat, Alif Silpachai, Ivana Lucic, Evgeny Chukharev-Hudilainen, John Levis, and Ricardo Gutierrez-Osuna. 2018. L2-arctic: A non-native english speech corpus. In *INTERSPEECH*, pages 2783–2787.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 8

☑ A2. Did you discuss any potential risks of your work?
Section 8

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

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

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

Section 6

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 5.2

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

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5 and 6

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

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 5

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

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.