ATES: AUTOMATIC TRENDING ENTITY SELECTION FOR INCREMENTAL LEARNING OF ALL-NEURAL ASR

Rini Sharon[⋆], *Megha Sharma*[⋆], *Harish Arsikere, Valentin Mendelev,*
Debmalya Chakrabarty, Bashar Awwad Shiekh Hasan

Amazon Alexa

ABSTRACT

Accurate recognition of trending words (such as latest song names and news items) is crucial to achieve smooth and delightful interactions with voice assistants. This is a challenging task even for state-of-the-art automatic speech recognition (ASR) models due to (a) continuous distribution shifts in customer traffic and (b) sparsity of trending words, originating from recent or upcoming events. Incremental learning (IL) offers a lightweight solution to address the former challenge — thereby avoiding the need for from-scratch training, but given the large data volumes available for incremental model updates, solving the latter challenge entails a data selection mechanism tailored to trending words. To this end, we propose ATES (Automated Trending Entity Selection), a framework that analyses shifts between historical training data and recent real-world data distributions for selecting utterances with trending words. Resulting dataset is transcribed using teacher model and used in IL. On an all-neural German ASR testbed, ATES demonstrates word error rate reductions of upto 6% for 3 trending test sets over 4 weeks of incremental updates (with minimal to no degradation on general test sets), as compared to an IL baseline with randomly sampled data.

***Index Terms*—** Data Selection, Recency, Trend Mining, ASR

1. INTRODUCTION AND RELATED WORK

The integration of voice assistants into daily life has expanded the scope of queries from simple task-oriented ones, such as setting alarms and reminders, to more recent and context-dependent queries like asking about trending terms such as “Barbeinheimer”, current events, or playing specific multimedia content. To effectively handle these diverse queries, voice assistants need to stay updated with ongoing trends, newly released media, and global events. While ASR models like Recurrent Neural Network Transducer (RNN-T) [1] and Conformer [2] have demonstrated strong performance on common and frequent queries, their accuracy drops for underrepresented or infrequent queries, which is still an area of active research [3, 4, 5]. This is because deep learning models, including all-neural ASR systems, have a tendency to memorize the data patterns presented during training [6, 7, 8, 9], emphasizing the importance of continuously training them with up-to-date data that captures recent trends. However, this continuous training is demanding in terms of manual transcription efforts and computational resources, making it challenging to scale across use-cases and languages.

To address the challenges of efficiently updating ASR models, approaches like incremental learning (IL) [10, 11, 12] have been explored where the model is fine-tuned for few epochs with small volume of transcribed data, sometimes complemented with data replay [10]. Commonly, incremental data used in training is randomly sampled from given corpus which does not guarantee inclusion of difficult and trending utterances, making data selection crucial in IL.

Various data selections methods have been explored in the past for improving ASR: [13] introduced unsupervised sub-modular selection using contrastive loss ratios, [14] explored entropy-based selection to improve phonetic balance in training data, [15] discussed uniform sampling, and [16] proposed lightly supervised parametric selection of well-transcribed, high-quality segments using ASR-transcribed audio. While these methods enhance ASR performance, they are general techniques that lack dynamic targeted data selection, which is essential for identifying trending data.

Target-aware data selection has been explored in [17, 18] where the objective is to choose a subset of speech data that matches the target ASR application scenario, such as different accents, acoustic conditions, or children’s speech. These methods offer an unsupervised approach and hence are mainly used for only acoustic encoder initialization. Additionally, it is expensive to run these methods regularly with small volume of data, limiting its usage in IL.

In this study, we introduce **Automated Trending Entity Selection (ATES)**, a novel semi-supervised data selection framework designed to capture trending data which is in turn used in incremental model training setup adapted from [10]. Given two distributions, historical (used for training prior model) and target (representing current customer queries), our framework identifies gaps between them, and selects data which can be added to the historical distribution to match the target distribution. This framework is cost-effective as it automatically detects the trends given historical and current customer queries and transcribes them using a teacher model. This automated system is designed in a modular manner, allowing components to be easily added, removed, or modified based on specific use cases or requirements. Our contributions, compared to existing works, are as follows:

- We introduce Automated Trending Entity Selection (ATES) framework which identifies trends from customer traffic that are under-represented in historical training distribution and not recognized confidently by customer serving model. ATES bridges this data representation gap by generating teacher transcriptions for trending data.
- We establish evaluation metrics for measuring trendiness of utterances using language models (Section 3.1) and catalogs curated by experts (Section 4) which enlist trending words. We conduct overlap analysis of data selected by ATES with these catalogs in Section 4 to show ATES data is in accordance with what is trending as per domain experts.
- We offer a detailed explanation of the design of the ATES framework, emphasizing its modularity (allowing seamless integration/removal of modules) and effectiveness. Through ablation studies, we highlight the contribution of each component in the overall performance.

2. METHODOLOGY

Objective of this study is to enhance performance of customer traffic serving model, called student model, on trending utterances. For this we consider two approaches: training from scratch (baseline) and incremental training. We further sub-classify the latter based on type of incremental data sources, three of which are described below.

We use incremental setting described in [10] as a baseline, called HT-based IL as it uses only human transcribed (HT) data as an incremental data source and replays data from prior data sources. In our experiments, we complement incremental HT with some subset of teacher-transcribed (TT) data, decoded using 300M bidirectional RNN-T model. Straight-forward way of deriving TT subset using random sampling is referred as TT$_{\text{Rand}}$ which we compare against ATES-based IL trained on incremental HT and “recency” subset (RS) where latter is derived using our framework. Here, “recency” means speech utterances containing trending words.

Training Overview. Figure 1 highlights differences amongst from-scratch (FS), HT-based IL and ATES-based IL. We use t_z notation to refer to z^{th} week. We start with an FS model M_{t_1} which is trained using all HT data ($HT_{:t_1}$) and randomly-sampled TT data ($TT_{:t_1}$) available till time t_1. When training from scratch, both HT and TT data from interval [t_1, t_2] are added to existing data, forming ($HT_{:t_1}$ + $HT_{t_{1:2}}$ + $TT_{:t_1}$ + $TT_{t_{1:2}}$). This data is used in FS training which is run for a large number of epochs, resulting in the relatively recent $M_{t_2}^{FS}$. In HT-based IL, we start from M_{t_1} and update it with incremental HT data ($HT_{t_{1:2}}$) for a small number of epochs, resulting in $M_{t_2}^{IL}$. ATES-based IL complements $HT_{t_{1:2}}$ with $RS_{t_{1:2}}$, where RS is recency-enhanced subset of TT, to train $M_{t_2}^{ATES}$. This process is repeated for multiple time intervals. Note that all IL-variants do data replay of $HT_{:t_1} + TT_{:t_1}$ along with using latest incremental data in training.

ATES Overview. Figure 2 illustrates differences between generation of TT$_{\text{Rand}}$ and RS datasets. While former is a randomly sampled subset of TT, the latter is derived using ATES framework which progressively sub-selects from input datasets by employing a sequence of filtering modules, to ultimately yield the RS dataset. Let us assume that we want to generate $RS_{t_{1:2}}$ which captures incremental trends. ATES takes historical training data distribution, represented using HT data and current real-world data distribution captured by student transcribed (ST) data. We take HT and ST from $[t_y, t_x]$ and $[t_x, t_2]$ time intervals respectively where x and y are hyper-parameters. The relative positioning of time intervals has been depicted in form of a timeline in Figure 2. These intervals were decided based on following design choices: (a) we use non-overlapping time intervals for HT and ST as we intend to capture distribution shift relative to training data of previous student model; (b) prior student model has been trained on all HT data till t_1 but we use its subset $HT_{t_{y:x}}$ to reduce computational cost of running ATES; (c) we use $ST_{t_{x:2}}$ instead of $ST_{t_{1:2}}$ for deriving $RS_{t_{1:2}}$ because $[t1, t2]$ is a weekly interval which we empirically found to be too short to capture trends, hence motivating extension of trailing time period.

ATES Modules. The first module in ATES is frequency-bucketing filter. It computes frequency of distinct words, called tokens, found in HT and ST transcriptions and arranges them in descending order. The tail, i.e. tokens with frequency less than a threshold (10 in our setup), is discarded to reduce computations. The resulting token lists from HT and ST are compared using the following criteria where k and j are tunable hyper-parameters.

(A): The token is present in top-k percentage bucket of the ST list.

(B): The token either does not exist in the HT list or is in the bottom-j percentage bucket of the HT list.

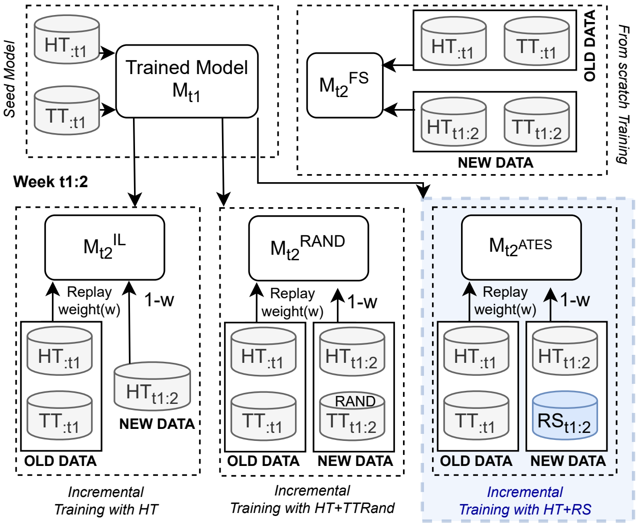

Fig. 1: *Overview of FS training, incremental training using HT and subsets of TT data like TT$_{Rand}$ and RS*

Tokens satisfying **A** and **B** are selected and passed as input to the second module. The dual conditions ensure that the token being selected is prominently present in recent ST while being scarcely present or absent in past HT data. The rationale behind this is to identify the differential data required to bring the train distribution (from HT) closer to the customer traffic distribution (using ST).

Next, in slot-filtering module tokens in transcriptions are categorized by their associated slot which is obtained using human or teacher model annotations. Similar to [19], slots in our setup are words in the transcription which refer to some entities such as “Song name”, “Artist name”, and more. In this module, we reject tokens associated with less relevant slots or no slot, ensuring the retention of tokens that hold importance with respect to recent trends. Relevance of a slot is dependent on the model’s application. For example, in a shopping chat-bot, music related slots can be considered irrelevant.

The recency tokens derived after applying first two filtering modules are mapped back to their original utterances, using utterance mapper module, which are then decoded using the teacher model. We further prune the resulting dataset using confidence filter module which discards utterances containing those recency tokens which are predicted with high confidence by both teacher and student model. Exclusion is done based on the assumption that confidence score is a proxy of model’s recognition accuracy. We rely on two scores and not just score from student to avoid false-accepts in cases where student is confident about wrong transcriptions. This module allows us to save training time and cost on tokens which model already recognises well, even if they are currently trending. For instance, widely recognized song title like “Memories” could appear in the trending subset due to its release popularity. However, retraining is unnecessary if the models already recognize it.

Once the RS dataset is obtained by applying aforementioned modules and transcribing using teacher, it is included along with $HT_{t_{1:2}}$ in the incremental training framework to periodically fine-tune models. This strategic selection aims to enhance resulting model’s ability to recognize recent speech utterances without being burdened by the time delays and extra efforts associated with manual human annotation. Note that quality of RS is heavily dependent on accuracy of teacher model. Although we use a fixed teacher model in all our experiments, we could have gotten even higher WERR gains by periodically updating the teacher model.

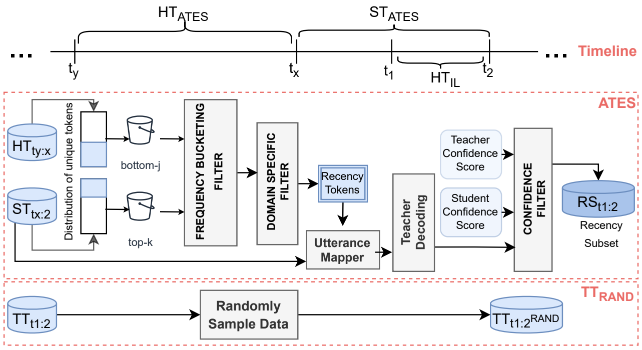

Fig. 2: *Flow diagram of ATES framework and TT_{Rand}. HT_{ATES} and ST_{ATES} in the Timeline denote the data supplied as input to the ATES framework, while HT_{IL} represents the input data for the IL framework.*

3. EXPERIMENTS

In this section, we provide details of the datasets, evaluation metrics, baselines pitted for comparison, and the experimental setup used.

3.1. Datasets and Evaluation Metrics

We use de-identified, far-field, in-house datasets in German for all our experiments. We did not perform additional experiments on public datasets as we were not able to find any datasets which have time aspect attached with data which is essential for defining notion of trending. The volume of HT and TT data seen by both FS and teacher models are 30k and 110k hours respectively. For the incremental set-up we used 100 hours of HT and 8k hours of TT from which we derive RS and TT_{Rand} subset of 450 hours. Note that in the ATES framework, ST data plays a role in selecting and filtering utterances, but its transcriptions are not directly used for training. Instead the chosen utterances are transcribed using the teacher model.

We report all our experimental results in terms of relative word error rate (WERR) compared to FS model under consideration. We introduce a metric LMTrend which scores utterances based on their trendiness. To compute this, we train 2 trigram language models [20] (LMs) - background and target. For $LMTrend_{t_{1:2}}$, background LM is trained with $HT_{:t_x}$ and target LM is trained with $HT_{:t_2}$. The LMTrend score is the difference between the perplexities obtained using the target LM and the background LM. Higher LMTrend score indicates larger shift in data distribution. Hence, we create test sets by selecting human-transcribed utterances from different domains (like music, video) having top 5% LMTrend score and use it for experimental evaluation. Additionally, we report WERR on tail test set and two general traffic test sets - General-5W and General-10W, containing 5 weeks and 10 weeks data respectively. We define an utterance as a tail utterance if it contains word(s) which are in bottom 1% of frequency distribution of all words derived from HT. Note that all these test sets are human-transcribed. Furthermore, a manually curated catalog of trending words, for time interval under consideration, is employed to qualitatively assess the trendiness of the utterances identified by the ATES.

Table 1: *Data sources used in Baselines and ATES for IL updates*

Model Name	HT	Cutoff	Slot	Rand	ATES-IL
Incremental Data	HT	HT + $\text{TT}_{\text{Cutoff}}$	HT + TT_{Slot}	HT + TT_{Rand}	HT+ RS

3.2. Experimental Setup

3.2.1. Student and Teacher Models

The from-scratch student model, an RNN-T [1], comprises 8 encoder layers, 2 decoder layers, a joint dense layer, and an output layer with softmax non-linearity. Each encoder and decoder layer contains 1024 Long Short-Term Memory (LSTM) [21] nodes. The joint network layer has a size of 512, while the output layer has 4001 dimensions, representing 4000 word-pieces plus a blank symbol. Input consists of 192-dimensional feature vectors integrating three 64-dimensional Log-Mel-Filterbanks computed every 10ms and stacked together. Training minimizes the RNN-T loss function with the Adam optimizer [22] and warmup-hold-decay learning rate scheduler. The model is trained with approximately 140k hours of HT and TT data combined which is augmented using SpecAugment technique [23]. The 130-epoch (5k steps/epoch) training process spans 48 GPUs, processing 1536 samples per step.

The teacher model, a non-streaming RNN-T model, features an 18-layer encoder with 800 nodes each, a 2 layer LSTM decoder with 1024 nodes each, and uses bidirectional multi-head attention [24] with context embeddings and dropout (0.1). Additionally, kernel and bias regularization with strengths of 1e-6 are incorporated. Both teacher and student models have auxiliary confidence models [25] which are updated post ASR model training. It predicts probability of ASR model's transcription being correct at two levels - token and utterance. ATES uses the token level confidence in confidence filter module to reject utterances containing recency tokens where both student and teacher are confident.

3.2.2. IL framework

IL framework employs data replay [10] to avoid model divergence and overfitting during consecutive updates. $r\%$ of new data is used, along with $100 - r\%$ of prior training data, in each weekly update. In our setup, replay weight $(100 - r\%)$ is 90% and the rest is split as 6% for incremental HT and 4% for subset of TT (such as TT_{Rand} and RS), maintaining a balance between new and previously learned patterns for stable model updates over time.

3.2.3. ATES

In our experimental setup, we use 4 and 8 weeks window for $[t_x, t_2]$ and $[t_y, t_x]$ respectively. We conducted experiments with 1 week window for $[t_x, t_2]$ but it did not give significant WER gains, motivating need for longer time window for capturing trends. For every IL update, we sample approximately 8K hours from $ST_{t_{x:2}}$ and take $HT_{t_{y:x}}$, both of which are then passed to ATES for sub-selection. Multiple hyper-parameters of the framework collectively influence the quality of recency sub-selected utterances, with the most pivotal one being bucketing percentage. Best combination of hyper-parameters obtained from tuning are: (a) *bucket-filtering:* top 10% of ST and bottom 30% of HT, (b) *slot-filtering:* excluding tokens with no slot, (c) *confidence-filtering:* ignoring tokens when both student and teacher confidence is greater than 800.

3.3. Baselines

We compare ATES model with other IL variants which differ in terms of incremental data sources used in training and have been listed in Table 3. HT-based IL uses just HT data for training whereas other baselines such as Cutoff, Slot and Rand use some subset of TT along with HT for training. Cutoff and Slot baselines are similar to Rand, which we discuss in detail in Section 2, except the method of obtaining subset of TT varies amongst them. Cutoff uses $\text{TT}_{\text{Cutoff}}$ which excludes teacher transcripts composed by only frequent words (i.e. words occurring in top 1% of word frequency distribution). Slot

Table 2: *Experimental results with WERR reported compared to the FS model where positive number indicates gain. T and HT refer to the teacher model and HT-based IL baseline respectively. 3 kinds of test sets have been presented - General traffic, Tail and LMTrend test sets. In LMTrend test sets, top 5% includes utterances having LMTrend score in the top 5%. Music and knowledge are subsets of the top 5% test sets and have been added separately to show domain-level metrics. The generation of these test sets involves extracting data on a rolling basis with respect to the training week under consideration.*

		Week 1			**Week 2**			**Week 3**			**Week 4**		
Test Set	**≈ No. of utts**	**T**	**HT**	**ATES**	**T**	**HT**	**ATES**	**T**	**HT**	**ATES**	**T**	**HT**	**ATES**
General-5W	15k	4.9	-0.7	0.8	4.7	-2.4	0.6	4.2	-3.6	-1.0	4.4	-4.1	-0.9
General-10W	65k	6.4	-0.9	-0.1	6.6	-2.5	-0.9	5.2	-5.2	-2.2	2.9	-6.3	-1.8
Tail	20k	5.7	-0.6	0.6	6.1	0.1	0.9	5.4	3.0	1.9	5.5	2.9	3.6
LMTrend													
Top 5%	20k	-6.7	2.9	5.4	-8.5	17.5	17.3	-7.8	2.4	4.6	-5.4	15.7	15.1
Music	3k	-22.5	7.9	11.2	-25.6	6.3	9.6	-21.7	10.5	10.9	-16.2	12.1	10.7
Knowledge	4k	11.1	4.4	5.1	9.7	6.0	7.3	8.9	0.0	6.2	7.0	-1.3	3.6

Table 3: *Comparison of baselines and ATES-based IL using WERR (higher is better) compared to FS model for week 1 of analysis.*

Test set	**HT**	**Rand**	**Cutoff**	**Slot**	**ATES**
General-5W	-0.7	-0.7	0.5	-0.7	0.8
General-10W	-0.9	-1.1	-0.3	-0.9	-0.1
Tail	-0.6	-0.3	-1.3	-1.0	0.6
LMTrend					
Top 5%	2.9	2.9	2.4	1.6	5.4
Music	7.9	4.9	4.9	7.3	11.2
Knowledge	4.4	7.1	5.0	0.6	5.1

uses TT_{Slot} which contains only those utterances which have slots. RS and TT_{Rand} had equal volume (approximately 450 hours) whereas that of TT_{Cutoff} and TT_{Slot} was 7K and 4K hours respectively.

4. RESULTS

4.1. Quantitative Analysis

We provide insights into WERR values (with respect to FS) reported in Table 2 by aggregating them across test sets (which are representative of the time frame being studied) and week pairs.

HT-baseline: When comparing the HT-baseline against the ATES-based IL, the latter consistently improves WERR with respect to HT 83.3% of the times by a range of 0.4% to 6.3% (2.4% on average). In scenarios where ATES doesn't outperform the HT-baseline (minimal degradation), it still shows a WERR improvement over the FS model in the range of 1.9% to 17.2%. We performed Wilcoxon test [26] on paired HT-baseline and ATES results (6 each) to test their statistical significance. The test shows significant differences in weeks 1-3 (p-values: 0.031, 0.093, 0.093), indicating notable improvement with introduction of ATES-based RS data over HT-baseline.

General Baselines: Table 3 shows that ATES-based model outperforms baselines on trending and tail test sets with average WERR improvements of 2.7%, 3.5% and 2% with respect to Cutoff, Slot and Rand respectively, while performing at par on general test sets.. We expected Cutoff and Slot to perform better than Rand but reverse observations are probably due to limited evaluation on 1 week.

Teacher: Although in few cases the teacher model exhibits WER degradation compared to the FS model, ATES still improves over the FS in most of them, probably due to the contribution of HT data. We hypothesise the degradation in teacher results is because it was not updated periodically.

Ablation Studies. We conduct ablation studies for week 2 by starting with ATES pipeline having only frequency-bucketing module and train ATEST-IL model on this data and incremental HT resulting in WERR of -19.7% compared to FS model, averaged across test sets. Next, we added slot filter module on top of frequency-bucketing module which improved WERR from -19.7% to -8.9%. Lastly, addition of confidence filter to existing two modules improved WERR to 5.8%, making it the most important module and emphasizing the importance of filtering already well-recognized tokens. The evaluation results show that bucket-filtering outputs raw recency tokens which are refined using subsequent modules to give WER improvements.

4.2. Qualitative Analysis

Domain experts prepared catalogs comprising trending words from various domains like Music, Sports and Knowledge for Week 1 shown in Table 2. To check quality of RS compared to HT dataset, we tokenized the datasets and catalog into unigrams and computed percentage of catalog covered by these datasets. For German, HT and RS covered 33.6% and 68.2% of catalog respectively, clearly indicating ATES' ability to select trending words without any human supervision. This is further validated by higher WERR gains with ATES, compared to HT, on LMTrend test sets as shown in Table 2.

We did the same analysis for datasets of different time interval in English language, to test the generalizability, and found a similar trend: HT covered 40.9% whereas RS covered 83.8% of the catalog. Furthermore, 60% of utterances in LMTrend test sets had one or more word overlaps with catalog validating trending nature of the test sets. Unlike German, we did not conduct further analysis in English by training models as data trend looked similar.

5. CONCLUSION

This paper introduces an automated trending entity selection framework - ATES, which selects trending data from customer traffic to improve ASR accuracy, amidst dynamic shifts in traffic distribution. The proposed framework identifies shifts between training and real-world traffic distribution, extracts trending utterances from the differential data which are transcribed using teacher and used in incremental model update. ATES results in upto 6% WERR of RNN-T student model as compared to IL baseline using only HT data. Additionally, we show that data selected by ATES has 70-80% overlap with trending words identified by domain experts. While ATES automates the data selection and transcription, its efficacy relies on teacher model's accuracy. To enhance performance, regular updating of teacher model will be investigated in future research. While advantages of ATES have been demonstrated in 1 high-resource language, we plan to test it in low-resource setting in future work.

6. REFERENCES

[1] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, “Sequence transduction with recurrent neural networks,” in *Proceedings of the 29th International Conference on Machine Learning (ICML'12)*, 2012.

[2] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, “Conformer: Convolution-augmented transformer for speech recognition,” 2020.

[3] Dhanush Bekal, Ashish Shenoy, Monica Sunkara, Sravan Bodapati, and Katrin Kirchhoff, “Remember the context! asr slot error correction through memorization,” in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 236–243.

[4] Xianrui Zheng, Yulan Liu, Deniz Gunceler, and Daniel Willett, “Using synthetic audio to improve the recognition of out-of-vocabulary words in end-to-end asr systems,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5674–5678.

[5] Cal Peyser, Sepand Mavandadi, Tara N. Sainath, James Apfel, Ruoming Pang, and Shankar Kumar, “Improving tail performance of a deliberation e2e asr model using a large text corpus,” 2020.

[6] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals, “Understanding deep learning requires rethinking generalization,” in *International Conference on Learning Representations*, 2017.

[7] Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien, “A Closer Look at Memorization in Deep Networks,” *arXiv*, June 2017.

[8] W. R. Huang, Steve Chien, Om Thakkar, and Rajiv Mathews, “Detecting Unintended Memorization in Language-Model-Fused ASR,” 2022, [Online; accessed 13. Sep. 2023].

[9] Jinyu Li, “Recent Advances in End-to-End Automatic Speech Recognition,” *ResearchGate*, Nov. 2021.

[10] Deepak Baby, Pasquale D'Alterio, and Valentin Mendelev, “Incremental learning for rnn-transducer based speech recognition models,” in *Interspeech 2022*, 2022.

[11] Li Fu, Xiaoxiao Li, Libo Zi, Zhengchen Zhang, Youzheng Wu, Xiaodong He, and Bowen Zhou, “Incremental Learning for End-to-End Automatic Speech Recognition,” *arXiv*, May 2020.

[12] Hisham Darjazini, Qi Cheng, and Ranjith Liyana-Pathirana, “Incremental Learning Algorithm for Speech Recognition,” in *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, pp. 231–235. Springer, Dordrecht, The Netherlands, 2007.

[13] Chanho Park, Rehan Ahmad, and Thomas Hain, “Unsupervised Data Selection for Speech Recognition with Contrastive Loss Ratios,” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8587–8591. IEEE, May 2022.

[14] Yusuke Yamada, Yuya Chiba, Takashi Nose, and Akinori Ito, “Effect of Training Data Selection for Speech Recognition of Emotional Speech,” *International Journal of Machine Learning and Computing*, vol. 11, no. 5, pp. 362–366, Sept. 2021.

[15] Yi Wu, Rong Zhang, and Alexander Rudnicky, “Data selection for speech recognition,” in *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pp. 562–565. IEEE, Dec. 2007.

[16] “NDLI Presents: Data Selection in the Framework of Automatic Speech Recognition,” Sept. 2023.

[17] Changfeng Gao, Gaofeng Cheng, Pengyuan Zhang, and Yonghong Yan, “Speech Corpora Divergence Based Unsupervised Data Selection for ASR,” *arXiv*, Feb. 2023.

[18] Zhiyun Lu, Yongqiang Wang, Yu Zhang, Wei Han, Zhehuai Chen, and Parisa Haghani, “Unsupervised data selection via discrete speech representation for asr,” 2022.

[19] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer, “Neural architectures for named entity recognition,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 260–270.

[20] Jean-Luc Gauvain, Lori Lamel, G. Adda, and J. Mariani, “Estimation of probabilities from sparse data for the language model component of a speech recognizer,” 02 2003.

[21] Sepp Hochreiter and Jürgen Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] Diederik P. Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” 2017.

[23] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le, “Specaugment: A simple data augmentation method for automatic speech recognition,” *arXiv preprint arXiv:1904.08779*, 2019.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, “Attention is all you need,” 2023.

[25] Prakhar Swarup, Roland Maas, Sri Garimella, Sri Harish Mallidi, and Björn Hoffmeister, “Improving asr confidence scores for alexa using acoustic and hypothesis embeddings,” 2019.

[26] Bernard Rosner, Robert J. Glynn, and Mei-Ling T. Lee, “The wilcoxon signed rank test for paired comparisons of clustered data,” *Biometrics*, vol. 62, no. 1, pp. 185–192, 2006.