Domain Adaptation of Automatic Speech Recognition Models for Diagnostic Applications

Aman Kumar[1], Ahmed Farahat[1], Huimin Zhuge[1], and Chetan Gupta[1]

[1] *Hitachi America Ltd., Santa Clara, California, USA*

aman.kumar@hal.hitachi.com

ahmed.farahat@hal.hitachi.com

joy.zhuge@hal.hitachi.com

chetangupta@gmail.com

ABSTRACT

Automatic speech recognition (ASR), or speech-to-text (STT), is becoming an important interface for AI systems in diagnostic workflows, but general-purpose ASR models often degrade in specialized technical domains. In diagnostic applications such as fault identification, root cause analysis, and repair recommendation, general-purpose ASR systems struggle with domain-specific terminology, abbreviations, part identifiers, and measurement expressions, leading to elevated transcription errors. This work presents a domain adaptation pipeline that unifies three components: a synthetic benchmarking framework in which domain-specific technical text is converted to speech via text-to-speech (TTS) synthesis and transcribed by open-source ASR models to establish baseline performance; Low-Rank Adaptation (LoRA)-based fine-tuning of Whisper Large-v3 using those synthetic audio-text pairs; and transfer validation on curated real-world automotive YouTube recordings to assess generalization beyond synthetic conditions. Using automotive technical language as a representative diagnostic domain, a data-scaling study employing progressively larger subsets of in-domain training data evaluates performance on a held-out test set via word error rate (WER), character error rate (CER), normalized error metrics, alphanumeric error rate, semantic similarity, and Bidirectional Encoder Representations from Transformers Score (BERTScore). Results show consistent gains from lightweight domain adaptation on both held-out synthetic data and real-world recordings, confirming that synthetic data generation combined with LoRA-based fine-tuning is an effective and computationally practical strategy for improving ASR accuracy in specialized technical domains where labeled speech is scarce.

1. INTRODUCTION

Maintenance workers increasingly rely on voice-driven AI systems for hands-free support during troubleshooting and repair, making reliable automatic speech recognition (ASR) a practical requirement in industrial diagnostic workflows. Users describe symptoms, components, operating conditions, and repair actions in spoken form, and reliable transcription is therefore a prerequisite for downstream tasks such as fault retrieval, root-cause analysis, and repair recommendation.

Despite major progress in end-to-end ASR, performance often degrades in specialized technical settings. Diagnostic speech contains domain-specific terminology, abbreviations, product names, part identifiers, and measurement expressions that general-purpose ASR models frequently misrecognize (Suh, Na, & Jung, 2024; Huang, Abdel-hamid, Li, & Evermann, 2020). In automotive and industrial applications, these linguistic challenges are often compounded by acoustic variability, including background noise, recording-channel differences, and diverse speaking styles. As a result, even strong general-purpose ASR systems may produce errors that are especially harmful in technical applications, where a small mistake in an identifier, dimension, or code can alter the intended meaning.

Recent large-scale pretrained ASR models such as wav2vec 2.0 and Whisper have improved robustness across varied speech conditions (Baevski, Zhou, Mohamed, & Auli, 2020; Radford et al., 2023). However, domain mismatch remains: the long tail of technical vocabulary and structured alphanumeric expressions is still difficult to recognize accurately in specialized settings. This motivates domain adaptation methods that can specialize general ASR models without requiring full retraining. Among parameter-efficient fine-tuning approaches, low-rank adaptation methods have shown particular promise for large encoder-decoder ASR models, as they avoid the inference latency overhead of adapter-based insertion and

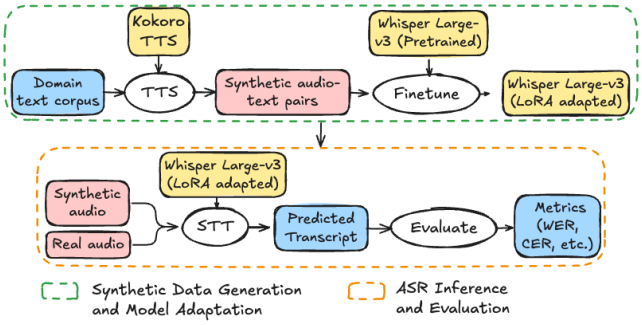

Figure 1. Overview of the proposed domain adaptation pipeline for diagnostic ASR. Domain text is converted to synthetic speech using Kokoro TTS, used to construct audio–text pairs for LoRA fine-tuning of Whisper Large-v3. The adapted model is evaluated on both synthetic and real-world speech.

the training instability associated with soft prompt methods, while achieving competitive domain-specific gains (Hu et al., 2022; Song et al., 2024; Prasad, Madikeri, Khalil, Motlicek, & Schuepbach, 2024).

A key challenge is the scarcity of labeled diagnostic speech. Collecting and transcribing domain-specific audio is expensive, particularly in industrial environments where speech may be sparse, noisy, and operationally constrained. In contrast, domain-specific technical text is often easier to obtain from manuals, parts catalogs, product descriptions, and technical documentation. This makes synthetic data generation an attractive strategy for controlled benchmarking and adaptation when real paired speech-text data are limited (Laptev et al., 2020; Zhong et al., 2022; Tran et al., 2025).

Modern neural text-to-speech (TTS) systems such as Tacotron 2, FastSpeech 2, VITS, and YourTTS make it practical to generate synthetic speech with high naturalness and efficient inference (Shen et al., 2017; Ren et al., 2020; Kim, Kong, & Son, 2021; Casanova et al., 2022). In addition, open and production TTS systems such as Coqui TTS (Coqui AI, 2023), Microsoft EdgeTTS (edge-tts contributors, 2023), and Kokoro (Hexgrad, 2024) enable efficient generation of large volumes of synthetic speech for benchmarking and model adaptation. Synthetic speech therefore provides controlled benchmarking and scalable supervision, but it does not fully reproduce the acoustic and prosodic variability of real diagnostic speech.

In this work, domain adaptation of ASR models for diagnostic applications is studied, using automotive technical language as a representative case. A synthetic pipeline is constructed in which domain text is converted to speech using a modern TTS system (Kokoro), enabling controlled benchmarking of ASR systems and creation of synthetic audio-text pairs for training. Whisper Large-v3 is then adapted using Low-Rank Adaptation (LoRA) (Hu et al., 2022), a parameter-efficient fine-tuning method that updates only a small subset of model parameters while leaving the rest frozen. Unlike adapter-based methods (Houlsby et al., 2019), which insert additional layers and increase inference latency, LoRA can be merged back into the original weights after training, incurring no overhead at inference time, a property that has made it the preferred PEFT approach for large-scale ASR adaptation (Song et al., 2024; Prasad et al., 2024). We evaluate the resulting models using a data-scaling study on a held-out synthetic test set and further assess generalization on curated real-world YouTube recordings from the automotive domain. An overview of the proposed pipeline is shown in Figure 1.

The main contributions of this work are as follows:

- A synthetic benchmarking pipeline for technical-domain ASR is developed using chunked automotive text and TTS generation, enabling controlled creation of domain-specific synthetic audio-text pairs.
- Parameter-efficient adaptation of Whisper Large-v3 via LoRA is performed, with a systematic data-scaling study examining how recognition quality improves with increasing amounts of in-domain training data.
- The adapted models are evaluated using complementary lexical and semantic metrics, including WER, CER, normalized error metrics, alphanumeric error rate, semantic similarity, and BERTScore.
- The best adapted model is assessed on curated real-world automotive audio to determine whether gains obtained on synthetic data transfer to more realistic speech conditions.

The remainder of this paper is organized as follows. Section 2 surveys related work on ASR domain adaptation and parameter-efficient fine-tuning. Section 3 describes the source corpus, its domain coverage, and the curation of the real-world evaluation set. Section 4 details the synthetic data generation pipeline, the LoRA adaptation framework, and all training settings. Section 5 defines the evaluation protocol, metrics, and confidence interval methodology. Section 6 presents and discusses results on both synthetic and real-world data, including a comparison with prior work. Section 7 concludes the paper.

2. RELATED WORK

Modern ASR has been shaped by large pretrained neural models that learn robust speech representations from broad and diverse data. Self-supervised approaches such as wav2vec 2.0 demonstrated that pretraining on large amounts of unlabeled audio can substantially reduce the amount of labeled speech required for downstream ASR (Baevski et al., 2020). Related self-supervised speech representation models such as HuBERT and WavLM further improved general-purpose speech embeddings and robustness across downstream tasks (Hsu et al., 2021; Chen et al., 2022). Whisper showed that large-scale weak supervision can yield strong robustness across

languages, accents, and noisy conditions, making it an attractive foundation model for zero-shot and adapted ASR (Radford et al., 2023).

Although these models perform well on many standard benchmarks, domain mismatch remains a persistent challenge. ASR systems trained on broad, general-purpose corpora often struggle in specialized environments where vocabulary, phrasing, and acoustic conditions differ from those seen during training. In such settings, transcription errors are frequently concentrated in domain-specific terminology and structured identifiers rather than common function words (Suh et al., 2024).

A broad line of work has explored domain adaptation for ASR. Earlier approaches focused on acoustic adaptation, language model adaptation, and data augmentation, including widely used perturbation and masking strategies such as speed perturbation and SpecAugment (Ko, Peddinti, Povey, & Khudanpur, 2015; Park et al., 2019). Beyond generic augmentation, several recent lines of work explore using external or text-only resources to drive adaptation (Hayashi et al., 2018; Zhu et al., 2023; Zhong et al., 2022; Tran et al., 2025). More recent work has emphasized parameter-efficient and text-driven adaptation strategies for large pretrained ASR models. Vanderreydt et al. proposed adaptive bottleneck-based parameter-efficient tuning for ASR, while Prasad et al. studied low-rank adaptation of pretrained speech models, including Whisper (Vanderreydt et al., 2023; Prasad et al., 2024). Kurian et al. further showed that domain-specific adaptation of Whisper can be performed using text-only fine-tuning of the decoder, highlighting the potential of lightweight adaptation when paired domain audio is limited (Kurian, Upadhyay, & Sengupta, 2025).

Parameter-efficient fine-tuning methods include adapter modules and low-rank updates. Adapter-based tuning reduces per-task trainable parameters by inserting lightweight modules into frozen networks (Houlsby et al., 2019). LoRA introduces trainable low-rank updates into frozen transformer layers, greatly reducing the number of trainable parameters while preserving adaptation capacity (Hu et al., 2022). Such methods are particularly relevant in industrial settings, where full-model retraining may be computationally impractical.

The present work is most closely related to studies on low-resource and domain-specific adaptation of Whisper-like ASR systems. Prior work has also explored the use of domain-adapted language models for industrial maintenance analysis, for example predicting maintenance actions from historical maintenance logs containing abbreviated technical descriptions (Kumar, Farahat, & Gupta, 2025). However, much of the prior work focuses on structured technical text or general adaptation settings, rather than spoken diagnostic language containing dense measurements, product names, and mixed alphanumeric strings. In contrast, the present work studies domain adaptation for diagnostic ASR in an automotive setting, evaluates performance with domain-aware metrics such as normalized error variants and alphanumeric error rate, and examines transfer from synthetic training data to curated real-world audio.

3. DATASET

3.1. Source Corpus

To support evaluation and adaptation of ASR models for diagnostic applications, a sample of a domain-specific automotive text corpus derived from the diagnostics-oriented dataset described in (Kumar, Amin, et al., 2025) is used. The sampled corpus contains 1,051 documents, each associated with a unique identifier and a technical text field. The documents cover automotive engineering topics such as engine systems, vehicle components, mechanical subsystems, control systems, and performance tuning.

The corpus contains dense domain-specific technical language relevant to diagnostic applications, including component names, process descriptions, measurements, and engineering terminology, making it well suited for both synthetic benchmarking and the construction of domain-specific speech data. Topics span engine systems, powertrain and transmission, fuel injection, braking and suspension, vehicle performance tuning, control systems, and motorsports engineering, reflecting the range of terminology found in technical manuals, repair guides, and diagnostic workflows.

3.2. Textual Data Statistics

The individual documents vary substantially in length. The full corpus contains approximately 681K tokens. Although modest in scale compared with large web corpora, it provides high-density domain knowledge that is directly relevant to technical ASR. Table 1 summarizes the length statistics of the corpus.

Table 1. Document length statistics of the automotive text corpus.

Statistic	Characters	Words
Minimum	348	59
Median	2,301	392
Mean	3,118	540
Maximum	28,229	4,863

4. METHODOLOGY

Because labeled diagnostic speech is scarce, a synthetic data generation framework is first constructed from domain-specific technical text. The text is converted into speech using a Kokoro TTS model, producing synthetic audio-text pairs. These pairs are used in two ways: first, to benchmark existing ASR models by comparing their transcriptions against the original text; and second, to fine-tune Whisper Large-v3 for in-domain speech recognition. This setup provides a controlled way to assess

how well existing ASR models handle technical vocabulary before adaptation is applied, while also supplying scalable synthetic supervision for training. Whether domain adaptation improves performance is then evaluated on a held-out synthetic test set, with generalization further assessed on curated real-world audio.

4.1. Data Curation

4.1.1. Audio-Text Pair Data Construction

For ASR fine-tuning, the text corpus was converted into 11,229 chunk-level utterances derived from 1,051 documents. Because ASR models typically operate on short audio segments, long documents are segmented into sentence-based chunks with a maximum length of 60 words. This limit was chosen for two practical reasons. First, Whisper processes audio in 30-second windows; at a natural speaking rate of approximately 130–150 words per minute, 60 words correspond to roughly 24–28 seconds of speech, fitting comfortably within that window. Second, chunks of this length contain at least one complete syntactic unit, providing sufficient local context for decoding. Sentences are treated as atomic units and never split across chunks; if adding the next sentence would exceed the limit, it is placed in the following chunk.

4.1.2. Item-based Data Construction

To avoid leakage, all chunks derived from the same source document are assigned to the same split, preventing chunks from the same document from appearing in both training and evaluation. This is important because chunks from the same document often share vocabulary and phrasing. Source documents are partitioned at the document level in an 80/10/10 ratio into a training pool, development set, and test set. From the training pool, incremental subsets from 10% to 100% are constructed for data-scaling experiments, where the 100% setting uses all utterances in the training pool and does not refer to the full corpus. Table 4 reports the resulting utterance counts.

4.1.3. Real Data Curation

Ten YouTube videos covering automotive diagnostics and education were curated for real-world evaluation, totalling approximately 11.75 hours of audio (mean duration $\approx$70 min per video, range 15–165 min).

Selection criteria. Videos were required to satisfy all of the following conditions: (i) the primary spoken language is English; (ii) content covers automotive diagnostic topics such as on-board diagnostics (OBD-II) systems, engine internals, braking and suspension systems, fuel injection, and direct injection; (iii) speech is clearly audible with no overlapping voices or dominant background music; and (iv) the speaker uses domain-specific technical vocabulary including diagnostic procedures. Videos were excluded if they contained predominantly non-technical conversational language, severe background noise that obscured speech, or content outside the automotive diagnostic domain. To reduce selection bias toward a single speaker or recording style, videos were drawn from four distinct channels representing different instructors, production qualities, and two recording environments: in-workshop, and screen-capture.

Table 2. Overall composition of the YouTube evaluation set.

Quantity	Value
Number of videos	10
Total audio duration	11.75 h
Total transcript words	144,250
Total transcript characters	642,432
Total transcript lines	17,593

Table 3. Per-video statistics of the YouTube evaluation set, summarising variation across the ten videos.

Statistic	Min	Median	Mean	Max
Audio duration (min)	15.2	54.1	70.5	164.6
Transcript words	2,661	8,951	14,425	36,067
Transcript characters	13,404	45,889	64,243	156,857
Transcript lines	278	1,264	1,759.3	4,253

Transcript verification and reliability. YouTube auto-generated transcripts served as the initial reference text. Each transcript was manually reviewed by spot-checking against the corresponding audio; randomly sampled segments were listened to and the auto-generated transcript was verified for those portions. The verified transcripts were used as ground-truth references for evaluation.

4.2. Baseline ASR Model

Our primary ASR model is Whisper Large-v3, a transformer-based encoder-decoder model trained on large-scale multilingual speech data. Whisper is chosen because it provides a strong general-purpose baseline and has demonstrated robustness across a wide range of acoustic conditions.

4.3. Audio Representation and Tokenization

Whisper models operate on log-Mel spectrogram representations of audio rather than raw waveform input. During preprocessing, audio signals are first resampled to 16 kHz and converted into log-Mel spectrograms using the Whisper feature extractor.

We use the *WhisperFeatureExtractor* provided in the HuggingFace Transformers library, which computes 128-channel log-Mel filterbank features with a 25 ms window and 10 ms

Table 4. Utterance counts per split. Source documents are partitioned 80/10/10 into training pool, development, and test sets at the document level. Training percentages denote incremental subsets of the training pool only; the 100% setting uses all 9,094 training pool utterances. Development and test sets remain fixed across all experiments.

Split	Utterances	Split	Utterances
Train 10%	953	Train 60%	5,473
Train 20%	1,673	Train 70%	6,279
Train 30%	2,553	Train 80%	7,211
Train 40%	3,471	Train 90%	8,143
Train 50%	4,613	Train 100%	9,094
Dev	999	Test	1,136

hop length. The resulting spectrogram frames serve as encoder input to the Whisper model.

Text transcripts are tokenized using the Whisper tokenizer. The tokenizer produces subword tokens (byte-pair encoding style subword units) that allow efficient representation of domain-specific terminology and alphanumeric expressions frequently found in diagnostic language (Sennrich, Haddow, & Birch, 2016). The HuggingFace *WhisperProcessor* abstraction is used to combine both the feature extractor and tokenizer into a unified interface. This ensures consistent preprocessing during both training and inference.

4.4. Domain Adaptation via LoRA

To adapt Whisper Large-v3 to diagnostic speech, we apply Low-Rank Adaptation (LoRA) (Hu et al., 2022). LoRA adds trainable low-rank update matrices to selected linear layers while keeping most pretrained weights fixed, which sharply reduces the number of trainable parameters and the cost of adaptation compared to full fine-tuning.

Decoder-only adaptation. In our main setup, LoRA is applied to the *decoder* only; the speech *encoder* stays frozen. We adopt this design for three reasons: (i) Whisper's encoder already provides strong general-purpose acoustic representations after large-scale pretraining, whereas domain-specific vocabulary and phrasing are primarily realized in the text decoder; (ii) updating fewer parameters reduces overfitting risk and GPU memory when the in-domain corpus is modest relative to Whisper's pretraining data; (iii) freezing the encoder limits catastrophic drift of broadly useful acoustic features while still allowing the decoder to specialize. This approach is consistent with prior parameter-efficient adaptation practice for large sequence-to-sequence models (Kurian et al., 2025; Song et al., 2024). A controlled ablation comparing decoder-only LoRA to encoder-inclusive LoRA was not conducted in this work and is left as future work.

Adapter configuration. Table 5 summarizes the LoRA configuration used for all fine-tuning experiments. The rank and scaling (r, α) were not grid-searched on the development set; they were fixed across all runs to keep experiments comparable across training-set sizes. Sensitivity to (r, α) and to encoder-inclusive LoRA is left to future work.

Table 5. LoRA configuration used for all fine-tuning experiments.

Setting	Value
PEFT type	LoRA (`SEQ_2_SEQ_LM`)
Trainable side	Decoder only; encoder frozen
Rank r	32
Scaling α	64 (i.e. $\alpha = 2r$)
LoRA dropout	0.05
Trainable LoRA parameters	5.77×10^7 ($\approx$57.7 M)
Total model parameters	1.60×10^9 ($\approx$1.60 B)
Trainable fraction	3.60%

Table 6. Main fine-tuning hyperparameters.

Setting	Value
Compute	$1 \times$ NVIDIA RTX 6000 Ada (48 GB)
Base model	Whisper Large-v3
Adapter	Decoder-only LoRA (Table 5)
Optimizer	AdamW
Peak learning rate	5×10^{-5}
LR schedule	Cosine decay with 5% warmup
Training budget	Up to 10 epochs
Batch construction	Per-device batch 32
Regularization	Label smoothing 0.05
Precision	Mixed precision (FP16)
Model selection	Early stopping on dev loss (patience 4)

4.5. Training Procedure

The model is fine-tuned separately on increasing fractions of the training pool:

$$10\%,\ 20\%,\ 30\%,\ 40\%,\ 50\%,\ 60\%,\ 70\%,\ 80\%,\ 90\%,\ 100\%.$$

This design enables a systematic study of how recognition quality scales with the amount of available in-domain data. Full training hyperparameters are listed in Table 6.

4.6. Training Objective

Whisper is trained using a sequence-to-sequence cross-entropy objective that predicts the next text token conditioned on the encoded audio representation. During fine-tuning, the model learns to generate the correct transcript tokens given the input speech features.

Let x denote the log-Mel spectrogram features extracted from the input audio and $y = (y_1, y_2, \ldots, y_T)$ denote the target transcript tokens. The model is trained to minimize the negative log-likelihood:

$$\mathcal{L} = -\sum_{t=1}^{T} \log P(y_t \mid y_{<t}, x) \tag{1}$$

When LoRA is applied, only the injected low-rank adapter parameters are updated during optimization, while the majority of the pre-trained model parameters remain frozen.

5. EVALUATION

5.1. Evaluation Protocol

For each training subset, we compare the base Whisper Large-v3 model against the corresponding fine-tuned model on the same held-out test set. This produces a consistent evaluation framework across all training sizes.

We also evaluate the base model and the best fine-tuned model on real data.

5.2. Inference Configuration

During inference, transcription is generated autoregressively by Whisper until an end-of-transcript token is produced. Because our dataset is English-only technical speech, decoding is performed in English transcription mode.

The same decoding configuration is used for both the base and fine-tuned models to ensure a fair comparison during evaluation.

5.3. Real-Data Evaluation Pipeline

For the YouTube evaluation set, light text preprocessing is applied before computing metrics. Each ground-truth transcript is split into sentence-like chunks with bounded word counts. Base and fine-tuned model predictions are then aligned to these chunks, and semantic similarity and BERTScore F1 are computed over aligned pairs. Chunk-level scores are aggregated into per-sample scores using chunk-length weights, so that longer chunks contribute proportionally more. We report both simple (unweighted) means and duration-weighted means across the 10 long-form videos, using audio duration as the weight so that longer recordings contribute proportionally more to the aggregate score.

5.4. Evaluation Metrics

We evaluate transcription quality using both lexical and semantic metrics.

- **Word Error Rate (WER).** WER is computed as the edit distance between reference and hypothesis word sequences normalized by the number of reference words (Levenshtein, 1966). This metric represents the standard measure of lexical transcription accuracy in ASR, where lower values indicate better performance.
- **Character Error Rate (CER).** CER is defined analogously to WER but computed over character sequences with spaces removed. This metric is sensitive to spelling errors and short token variations. Lower values indicate better transcription accuracy.
- **Normalized WER/CER.** To reduce the influence of punctuation and capitalization differences, we also report normalized error rates computed after removing punctuation and collapsing whitespace in a case-insensitive manner. This normalization emphasizes lexical content rather than formatting variations.
- **ASR-normalized WER/CER.** To account for equivalent numeric expressions, digit sequences in the range 0–99 are converted to their textual representations (e.g., "10" $\rightarrow$ "ten") prior to evaluation, followed by lowercasing and whitespace normalization. This normalization treats equivalent number forms as correct and is useful for technical-domain ASR evaluation, where numeric expressions may appear in either digit or word form.
- **Alphanumeric Error Rate.** This metric computes edit distance over alphanumeric tokens (maximal letter–digit sequences) only, divided by the number of reference alphanumeric tokens. It specifically targets domain-relevant identifiers such as diagnostic trouble codes (DTCs), VINs, and part numbers. Lower values indicate better performance.
- **Semantic Similarity.** Semantic similarity is computed as the cosine similarity between sentence embeddings of the reference and hypothesis generated using a pretrained sentence encoder. Scores are calculated for aligned utterance pairs and aggregated across the evaluation set. Values lie in $[0, 1]$, with higher values indicating stronger semantic preservation.
- **BERTScore F1.** BERTScore evaluates semantic overlap using contextual token embeddings derived from a pretrained BERT model (Zhang, Kishore, Wu, Weinberger, & Artzi, 2020). The F1 score balances precision and recall between the hypothesis and reference tokens. Scores range from $[0, 1]$, where higher values indicate better semantic alignment.

6. RESULTS AND DISCUSSION

6.1. Quantitative Results on Synthetic Held-out Dataset

Figure 2 (values in Appendix Table 8) shows performance on the held-out synthetic test set for the baseline Whisper model and LoRA fine-tuned models trained with increasing fractions of the diagnostic training data.

Across all lexical metrics, performance generally improves as more in-domain data is used for fine-tuning. Even the 10% subset reduces WER and CER relative to the baseline. Most metrics reach their best values at 80%–90% of the training pool, with marginal or slightly reversed trends at 100%, suggesting diminishing returns rather than continued monotonic improvement at the largest data fractions. In particular, WER decreases from 0.0842 to 0.0738 and CER from 0.0234 to

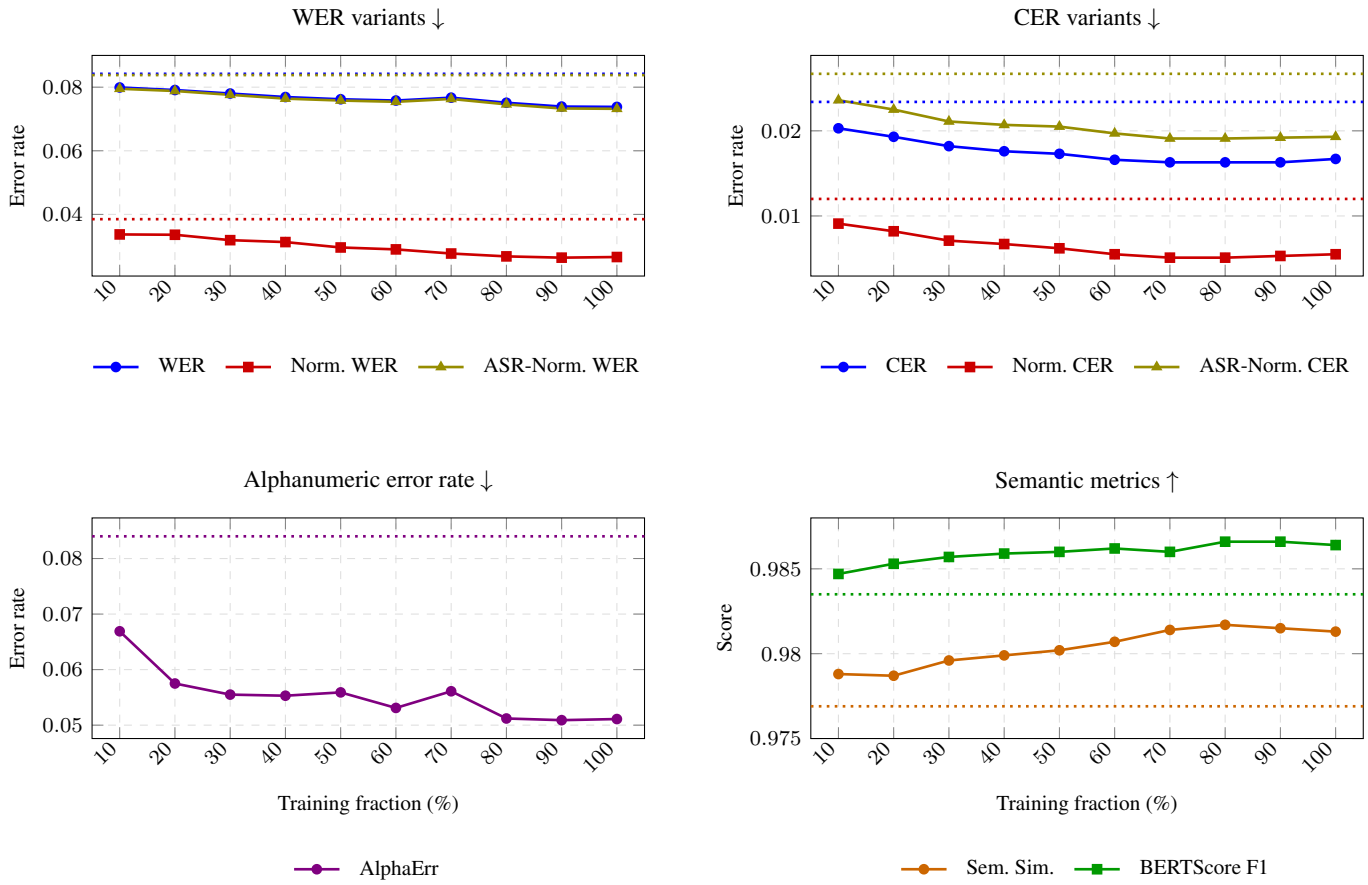

Figure 2. Performance of LoRA fine-tuned models on the held-out synthetic test set across training data fractions (10%–100% of the training pool). Solid lines show fine-tuned model performance; dotted lines of matching color indicate the unmodified Whisper Large-v3 baseline for each metric. ↓ lower is better; ↑ higher is better.

0.0163, demonstrating substantial gains from lightweight domain adaptation.

The largest relative improvements appear in the normalized metrics, where Normalized CER and Normalized WER decrease by more than 50% and 30%, respectively. This indicates that the fine-tuned model becomes significantly more robust to punctuation and formatting variations while accurately capturing domain-specific terminology.

Performance also improves for domain-relevant alphanumeric tokens, with the alphanumeric error rate decreasing from 0.0840 to 0.0509. This suggests that the adapted model better recognizes identifiers such as codes and technical strings, which are common in diagnostic transcripts.

Semantic metrics show smaller but consistent gains. Semantic similarity increases from 0.9769 to 0.9817 and BERTScore F1 from 0.9835 to 0.9866, indicating modest improvements in overall meaning preservation. Overall, these results demonstrate that parameter-efficient domain adaptation substantially improves lexical accuracy while maintaining strong semantic fidelity.

6.2. Quantitative Results on Real Dataset

Evaluation follows the pipeline described in Section 5.3. Transcription with the fine-tuned model required 3.5 hours of wall-clock time (realtime factor 3.32×), with a mean per-file processing time of 21.3 ± 17.5 minutes on a single GPU. Table 7 presents evaluation results on the curated YouTube dataset under both unweighted and duration-weighted aggregation. The fine-tuned model consistently outperforms the base model across all lexical metrics. Under the unweighted setting, WER decreases from 0.1951 to 0.1444 (25.98% error reduction) and CER decreases from 0.1148 to 0.0683 (40.48%). The largest gains appear in normalized metrics, with NormCER and NormWER improving by 48.78% and 46.41%, respectively, indicating better robustness to formatting variations. The alphanumeric error rate also decreases by 29.61%, reflecting improved recognition of domain-specific identifiers. Duration-weighted results follow similar trends but with slightly smaller gains, suggesting longer recordings remain more challenging. Semantic metrics improve modestly, with semantic similarity increasing from 0.9126 to 0.9349 and BERTScore F1 from 0.9590 to 0.9635, indicating improved preservation

Table 7. Performance on the curated YouTube evaluation set consisting of 10 diagnostic audio recordings with reference transcripts as ground truth. Improvement indicates percentage error reduction for error-based metrics and percentage increase for semantic similarity and BERTScore F1.

Evaluation		WER	CER	NormWER	NormCER	ASRWER	ASRCER	AlphaErr	SemSim	BERTSc
Un-weighted	Base	0.1951	0.1148	0.1258	0.0970	0.1938	0.1135	0.1568	0.9126	0.9590
	FT	0.1444	0.0683	0.0674	0.0497	0.1429	0.0673	0.1104	0.9349	0.9635
	Imp.(%)	25.98	40.50	46.42	48.76	26.26	40.70	29.59	2.44	0.47
Duration-weighted	Base	0.2008	0.1171	0.1277	0.0984	0.1998	0.1164	0.1648	0.9135	0.9569
	FT	0.1744	0.0824	0.0782	0.0591	0.1732	0.0815	0.1353	0.9263	0.9578
	Imp.(%)	13.17	29.63	38.78	39.96	13.27	29.98	17.90	1.40	0.09

of meaning in the generated transcripts.

6.3. Novelty and Comparison with Prior Work

The present work differs from closely related studies along several dimensions. Kurian et al. (Kurian et al., 2025) demonstrated decoder-only adaptation of Whisper using text-only data, without paired audio. The present work instead constructs full synthetic audio-text pairs via TTS synthesis and uses them for LoRA fine-tuning, providing richer supervision that captures both acoustic and linguistic domain characteristics. Additionally, evaluation here extends to curated real-world organic speech rather than being limited to read-speech benchmarks, providing a more realistic assessment of deployed performance.

Song et al. (Song et al., 2024) applied LoRA to Whisper for general-domain accent adaptation. The present work targets a substantially more challenging setting: alphanumeric-dense diagnostic speech containing structured part identifiers, measurement expressions, and low-frequency product names absent from general-purpose training corpora. The use of alphanumeric error rate as a domain-specific evaluation metric targets the error types most consequential in diagnostic workflows, complementing the standard WER that dominates prior Whisper adaptation studies.

The data-scaling study spanning ten training fractions from 10% to 100% provides a systematic characterization of sample efficiency in technical-domain ASR adaptation. Prior Whisper adaptation studies typically report results at a single training size; the ten-fraction scaling study here provides a more systematic view of sample efficiency in this setting.

Finally, the synthetic data pipeline, which converts domain-specific text to speech using Kokoro TTS and evaluates the resulting audio-text pairs through a structured benchmarking protocol, offers a replicable methodology that is generalizable beyond the automotive domain to any technical field for which domain text corpora are available.

6.4. Limitations

Although the adapted model improves average performance on both synthetic and real-world data, several limitations warrant acknowledgement.

Transcription errors on structured identifiers. Domain adaptation does not eliminate all technical transcription errors. Some alphanumeric identifiers remain difficult, particularly when their pronunciation is phonetically similar to common words or measurement units, as illustrated by the *4L60* $\rightarrow$ *4.0L 60* counterexample in Appendix A.

Single-TTS training bias. All synthetic training and held-out test data were generated by a single TTS system (Kokoro). This introduces a potential train-test acoustic distribution match that may inflate performance on the synthetic test set: the fine-tuned model may partially learn acoustic characteristics specific to Kokoro rather than internalizing general domain vocabulary knowledge. Cross-TTS generalization was not investigated in this study. Such experiments would clarify whether observed gains reflect genuine domain-vocabulary learning or system-specific acoustic artifacts, and constitute an important direction for future work.

Scope of real-world evaluation. The real-world evaluation is limited to 10 videos from a single language (English) and a single technical domain (automotive). Generalization to other specialized domains such as aerospace, medical devices, or industrial machinery remains to be demonstrated.

7. CONCLUSION

This work investigated domain adaptation of automatic speech recognition for automotive diagnostic speech, combining a TTS-based synthetic data generation pipeline, parameter-efficient LoRA fine-tuning of Whisper Large-v3, and transfer validation on curated real-world YouTube recordings. Experimental results demonstrate that even modest amounts of in-domain synthetic training data consistently improve transcription accuracy: WER decreases from 0.0842 to 0.0738 on the held-out synthetic test set and from 0.1951 to 0.1444 on real-world automotive recordings. Normalized error metrics show the

largest relative improvements, exceeding 45%, reflecting the model's improved robustness to domain-specific formatting and terminology. Recognition of alphanumeric identifiers, the most consequential error type in diagnostic applications, also improves substantially, with alphanumeric error rate decreasing from 0.0840 to 0.0509 on the synthetic test set and from 0.1568 to 0.1104 on real-world data. These findings demonstrate that LoRA-based fine-tuning on synthetic audio-text pairs is an effective and computationally efficient strategy for improving ASR accuracy in the automotive diagnostic domain where labeled speech is scarce. Future work should investigate cross-TTS generalization, perform systematic LoRA hyperparameter search, extend evaluation to additional technical domains, and explore domain language constraints at decoding time.

REFERENCES

Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 33, pp. 12449–12460). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html

Casanova, E., Weber, J., Shulby, C. D., Candido Junior, A., Gölge, E., & Ponti, M. A. (2022, 17–23 Jul). YourTTS: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, & S. Sabato (Eds.), *Proceedings of the 39th international conference on machine learning* (Vol. 162, pp. 2709–2720). PMLR. Retrieved from https://proceedings.mlr.press/v162/casanova22a.html

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., ... Wei, F. (2022, October). WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, *16*(6), 1505–1518. Retrieved from https://doi.org/10.1109/JSTSP.2022.3188113 doi: 10.1109/JSTSP.2022.3188113

Coqui AI. (2023). *Coqui TTS: A deep learning toolkit for text-to-speech.* https://github.com/coqui-ai/TTS. (Accessed: 2026-03-12)

edge-tts contributors. (2023). *edge-tts: Python client for microsoft edge text-to-speech.* https://github.com/rany2/edge-tts. (Accessed: 2026-03-12)

Hayashi, T., Watanabe, S., Zhang, Y., Toda, T., Hori, T., & Astudillo, R. (2018). Back-translation-style data augmentation for end-to-end ASR. In *2018 IEEE spoken language technology workshop (SLT)* (pp. 426–433). IEEE. doi: 10.1109/SLT.2018.8639619

Hexgrad. (2024). *Kokoro-82m: Open-weight neural text-to-speech model.* https://huggingface.co/hexgrad/Kokoro-82M. (Accessed: 2026-03-12)

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... Gelly, S. (2019). Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th international conference on machine learning* (Vol. 97, pp. 2790–2799). PMLR. Retrieved from https://proceedings.mlr.press/v97/houlsby19a.html

Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., & Mohamed, A. (2021). HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 3451–3460. Retrieved from https://doi.org/10.1109/TASLP.2021.3122291 doi: 10.1109/TASLP.2021.3122291

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... Chen, W. (2022). LoRA: Low-rank adaptation of large language models. In *International conference on learning representations.* Retrieved from https://openreview.net/forum?id=nZeVKeeFYf9 (arXiv:2106.09685)

Huang, R., Abdel-hamid, O., Li, X., & Evermann, G. (2020). Class LM and word mapping for contextual biasing in end-to-end ASR. In *Interspeech 2020* (pp. 4348–4351). Retrieved from https://www.isca-speech.org/archive/interspeech_2020/huang20f_interspeech.html doi: 10.21437/Interspeech.2020-1787

Kim, J., Kong, J., & Son, J. (2021, 18–24 Jul). Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th international conference on machine learning* (Vol. 139, pp. 5530–5540). PMLR. Retrieved from https://proceedings.mlr.press/v139/kim21f.html

Ko, T., Peddinti, V., Povey, D., & Khudanpur, S. (2015). Audio augmentation for speech recognition. In *Interspeech 2015* (pp. 3586–3589). Retrieved from https://www.isca-speech.org/archive/interspeech_2015/ko15_interspeech.html doi: 10.21437/Interspeech.2015-711

Kumar, A., Amin, E. M., Lee, X. Y., Vidyaratne, L., Farahat, A. K., Ghosh, D. D., ... Gupta, C. (2025). Building domain-specific small language models via guided data generation. *arXiv preprint arXiv:2511.21748*. Retrieved from https://arxiv.org/abs/2511.21748

Kumar, A., Farahat, A., & Gupta, C. (2025). Predicting maintenance actions from historical logs using domain-

specific LLMs. In *Proceedings of the PHM society asia-pacific conference* (Vol. 5). Retrieved from https://papers.phmsociety.org/index.php/phmap/article/view/4652 doi: 10.36001/phmap.2025.v5i1.4652

Kurian, B., Upadhyay, A., & Sengupta, A. (2025). Domain-specific adaptation for ASR through text-only fine-tuning. In *Proceedings of the 1st workshop on multimodal models for low-resource contexts and social impact (mmloso 2025)* (pp. 78–85). Mumbai, India: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2025.mmloso-1.7/

Laptev, A., Korostik, V., Svischev, A., Andrusenko, A., Medennikov, I., & Rybin, S. (2020). You do not need more data: Improving end-to-end speech recognition by text-to-speech data augmentation. In *2020 13th international congress on image and signal processing, biomedical engineering and informatics (cisp-bmei)* (pp. 439–444). IEEE. doi: 10.1109/CISPBMEI51763.2020.9263564

Levenshtein, V. I. (1966, February). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, *10*(8), 707–710.

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019* (pp. 2613–2617). Retrieved from https://www.isca-speech.org/archive/interspeech_2019/park19e_interspeech.html doi: 10.21437/Interspeech.2019-2680

Prasad, A., Madikeri, S., Khalil, D., Motlicek, P., & Schuepbach, C. (2024). Speech and language recognition with low-rank adaptation of pretrained models. In *Interspeech 2024* (pp. 2825–2829). Retrieved from https://www.isca-speech.org/archive/interspeech_2024/prasad24_interspeech.html doi: 10.21437/Interspeech.2024-2187

Radford, A., Kim, J. W., Xu, T., Brockman, G., Mcleavey, C., & Sutskever, I. (2023, 23–29 Jul). Robust speech recognition via large-scale weak supervision. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th international conference on machine learning* (Vol. 202, pp. 28492–28518). PMLR. Retrieved from https://proceedings.mlr.press/v202/radford23a.html

Ren, Y., Hu, C., Qin, T., Zhao, S., Zhao, Z., & Liu, T.-Y. (2020). FastSpeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558*. Retrieved from https://arxiv.org/abs/2006.04558

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1715–1725). Berlin, Germany: Association for Computational Linguistics. Retrieved from https://aclanthology.org/P16-1162/ doi: 10.18653/v1/P16-1162

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., ... Wu, Y. (2017). Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions. *arXiv preprint arXiv:1712.05884*. Retrieved from https://arxiv.org/abs/1712.05884 (Commonly referred to as Tacotron 2; later widely cited in 2018)

Song, Z., Zhuo, J., Yang, Y., Ma, Z., Zhang, S., & Chen, X. (2024). LoRA-Whisper: Parameter-efficient and extensible multilingual ASR. In *Interspeech 2024* (pp. 3934–3938). Retrieved from https://www.isca-speech.org/archive/interspeech_2024/song24_interspeech.html doi: 10.21437/Interspeech.2024-892

Suh, J., Na, I., & Jung, W. (2024). Improving domain-specific ASR with LLM-generated contextual descriptions. In *Interspeech 2024* (pp. 1255–1259). Retrieved from https://www.isca-speech.org/archive/interspeech_2024/suh24_interspeech.html doi: 10.21437/Interspeech.2024-377

Tran, M., Pang, Y., Paul, D., Pandey, L., Jiang, K., Guo, J., ... Lei, X. (2025). A domain adaptation framework for speech recognition systems with only synthetic data. *arXiv preprint arXiv:2501.12501*. Retrieved from https://arxiv.org/abs/2501.12501 doi: 10.48550/arXiv.2501.12501

Vanderreydt, G., Prasad, A., Khalil, D., Madikeri, S., Demuynck, K., & Motlicek, P. (2023). Parameter-efficient tuning with adaptive bottlenecks for automatic speech recognition. In *2023 IEEE automatic speech recognition and understanding workshop (ASRU)* (pp. 1–7). IEEE. doi: 10.1109/ASRU57964.2023.10389769

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. In *International conference on learning representations*. Retrieved from https://openreview.net/forum?id=SkeHuCVFDr (arXiv:1904.09675)

Zhong, G., Song, H., Wang, R., Sun, L., Liu, D., Pan, J., ... Dai, L. (2022). External text based data augmentation for low-resource speech recognition in the constrained condition of OpenASR21 challenge. In *Interspeech 2022* (pp. 4860–4864). Retrieved from https://www.isca-speech.org/archive/interspeech_2022/zhong22_interspeech.html doi: 10.21437/Interspeech.2022-649

Table 8. Held-out synthetic test data performance of the base Whisper model and LoRA fine-tuned models trained on increasing fractions of the diagnostic training set. The test set contains 1,136 audio samples with a mean duration of 20.21 seconds. Bold indicates the best value.

Metric	Baseline	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
WER	0.0842	0.0799	0.0791	0.0780	0.0769	0.0762	0.0758	0.0767	0.0751	0.0739	**0.0738**
CER	0.0234	0.0203	0.0193	0.0182	0.0176	0.0173	0.0166	**0.0163**	**0.0163**	**0.0163**	0.0167
Normalized WER	0.0385	0.0337	0.0336	0.0319	0.0313	0.0296	0.0290	0.0277	0.0268	**0.0264**	0.0266
Normalized CER	0.0120	0.0091	0.0082	0.0071	0.0067	0.0062	0.0055	**0.0051**	**0.0051**	0.0053	0.0055
ASR-normalized WER	0.0838	0.0795	0.0788	0.0776	0.0764	0.0758	0.0754	0.0763	0.0746	0.0733	**0.0732**
ASR-normalized CER	0.0267	0.0236	0.0225	0.0211	0.0207	0.0205	0.0197	**0.0191**	**0.0191**	0.0192	0.0193
Alphanumeric error rate	0.0840	0.0669	0.0575	0.0555	0.0553	0.0559	0.0531	0.0561	0.0512	**0.0509**	0.0511
Semantic similarity	0.9769	0.9788	0.9787	0.9796	0.9799	0.9802	0.9807	0.9814	**0.9817**	0.9815	0.9813
BERTScore F1	0.9835	0.9847	0.9853	0.9857	0.9859	0.9860	0.9862	0.9860	**0.9866**	**0.9866**	0.9864

Zhu, J., Tong, W., Xu, Y., Song, C., Wu, Z., You, Z., ... Meng, H. (2023). Text-only domain adaptation for end-to-end speech recognition through down-sampling acoustic representation. In *Interspeech 2023* (pp. 1334–1338). Retrieved from https://www.isca-speech.org/archive/interspeech_2023/zhu23f_interspeech.html doi: 10.21437/Interspeech.2023-1378

A. QUALITATIVE EXAMPLES WITH METRIC COMPARISON

To further illustrate the effect of domain-specific fine-tuning, we present qualitative examples together with their evaluation metrics. Lower values are better for WER, CER, and alphanumeric error rate, while higher values are better for semantic similarity and BERTScore.

A.1. Positive Examples

Table 9 presents a case involving the measurement expression *MVT 7 x 110 mm*. The base model produces three distinct error types: (i) the unit token `mm` is misrecognized as the capital letter `M` throughout; (ii) the alphanumeric token `7x110mm` is corrupted into `MVT7X110M`, collapsing the product prefix into the dimension string; and (iii) the delimiter `/` is expanded to the spoken word `slash`, breaking downstream parsing of the dimension range. The fine-tuned model recovers the correct measurement structure, unit tokens, and delimiter, yielding an 80.04% reduction in alphanumeric error rate and complete elimination of normalized CER. This example illustrates that domain adaptation directly targets the most diagnostically harmful error type: corruption of structured measurement expressions.

Table 10 illustrates a case involving the product identifier *INCOFLUX NT100*. The base model silently deletes the entire identifier, a typical out-of-vocabulary deletion for a general-purpose model trained on broad web speech data. This omission causes a downstream sentence boundary error and degrades semantic similarity to 0.876, demonstrating that missing identifiers have consequences beyond the immediate token. The fine-tuned model reconstructs it correctly within its sentential context, achieving a 100% reduction in alphanumeric error rate and restoring semantic similarity to 0.991.

A.2. Negative Example

Table 11 presents a case where domain adaptation introduces an error. The base model correctly transcribes the GM automatic transmission code *4L60*, while the fine-tuned model converts it to *4.0L 60*. This substitution is caused by over-generalization: `4.0L` (denoting a 4.0-litre engine displacement) occurs frequently in the training corpus and is phonetically indistinguishable from `4L60`, leading the model to reinterpret the identifier through the more common pattern. The error propagates to every occurrence of the identifier in the passage, increasing alphanumeric error rate from 0.071 to 0.190 (+167.61%). This highlights a systematic failure mode of domain adaptation: the model can over-generalize high-frequency training patterns and reinterpret structurally ambiguous identifiers in ways that are acoustically plausible but factually incorrect. Potential mitigations include: (i) including more diverse training examples containing `4L60`-type codes in varied sentential contexts; (ii) applying a whitelist-constrained decoder that prevents rewriting of known part-code tokens; or (iii) post-processing with a regular-expression filter that flags suspected alphanumeric reinterpretations for human review.

Table 9. Example illustrating improved recognition of measurement expressions and technical formatting after domain-specific fine-tuning.

Ground Truth Cylinder Components: Cylinder Studs - MVT 7 x 110mm Stroke 44.8 / 45mm Connecting: This entry describes cylinder studs from MVT. They are 7mm in diameter and 110mm long.
Base Model Cylinder components. Cylinder studs MVT7X110 Mstroke 44.8 slash 45 M connecting. This entry describes cylinder studs from MVT. They are 7 M in diameter and 110 M long.
fine-tuned Model Cylinder Components. Cylinder Studs, MVT 7x110mm Stroke 44.8-45mm Connecting. This entry describes cylinder studs from MVT. They are 7mm in diameter and 110mm long.

Model	**WER**	**CER**	**NormWER**	**NormCER**	**ASRWER**	**ASRCER**	**AlphaErr**	**SemSim**	**BERTSc**
Base	0.517	0.085	0.407	0.067	0.500	0.186	0.536	0.913	0.924
Fine-tuned	0.345	0.028	0.185	0.000	0.333	0.058	0.107	0.980	0.978
Improvement (%)	33.27↓	66.61↓	54.52↓	100.00↓	33.33↓	68.82↓	80.04↓	7.36↑	5.84↑

Table 10. Example illustrating improved recognition of domain-specific product names such as *INCOFLUX NT100*.

Ground Truth SAW: INCOFLUX NT100 Submerged Arc Flux is used for the SAW process. NILO Filler Metal 365 NILO Filler Metal 365 is an age-hardenable alloy designed for welding NILO alloy 365, which is used in fiber-reinforced epoxy-resin tooling applications.
Base Model SAW is used for the SAW process. NILO filler metal 365 is an age-hardenable alloy designed for welding NILO alloy 365, which is used in fiber-reinforced epoxy resin tooling applications.
Fine-tuned Model SAW, INCOFLUX NT100 Submerged Arc Flux, is used for the SAW process. NILO Filler Metal 365 NILO Filler Metal 365 is an age-hardenable alloy designed for welding NILO alloy 365, which is used in fiber-reinforced epoxy resin tooling applications.

Model	**WER**	**CER**	**NormWER**	**NormCER**	**ASRWER**	**ASRCER**	**AlphaErr**	**SemSim**	**BERTSc**
Base	0.316	0.238	0.289	0.236	0.316	0.238	0.268	0.876	0.952
Fine-tuned	0.105	0.015	0.053	0.000	0.105	0.015	0.000	0.991	0.990
Improvement (%)	66.77↓	93.70↓	81.66↓	100.00↓	66.77↓	93.70↓	100.00↓	13.13↑	3.99↑

Table 11. Counterexample where domain adaptation introduces an error in a technical identifier. The fine-tuned model incorrectly converts the transmission code *4L60* into *4.0L 60*, leading to higher lexical and alphanumeric error rates.

Ground Truth FTI 3800 Hard Hit 9.5" Converter for 4L60: This torque converter is designed for high-performance applications and is compatible with the 4L60 automatic transmission. The "hard hit" designation indicates a higher stall speed, allowing for quicker acceleration off the line.
Base Model FTI 3800 Hard Hit 9.5, converter for 4L60. This torque converter is designed for high-performance applications and is compatible with the 4L60 automatic transmission. The Hard Hit designation indicates a higher stall speed, allowing for quicker acceleration off the line.
Fine-tuned Model FTI 3800 Hard-Hit 9.5, Converter for 4.0L 60. This torque converter is designed for high-performance applications and is compatible with the 4.0L 60 automatic transmission. The Hard-Hit designation indicates a higher stall speed, allowing for quicker acceleration off the line.

Model	**WER**	**CER**	**NormWER**	**NormCER**	**ASRWER**	**ASRCER**	**AlphaErr**	**SemSim**	**BERTSc**
Base	0.100	0.017	0.000	0.000	0.100	0.017	0.071	0.987	0.981
Fine-tuned	0.225	0.043	0.200	0.009	0.225	0.117	0.190	0.953	0.971
Change (%)	125.00↑	152.94↑	–↑	–↑	125.00↑	588.24↑	167.61↑	3.44↓	1.02↓