SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY - INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Master's Thesis in Informatics

Evaluating and Improving Automatic Speech Recognition for Industry Specific Jargon in Low-Resource Scenarios

Leonhard Siegfried Stengel

SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY - INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Master's Thesis in Informatics

Evaluating and Improving Automatic Speech Recognition for Industry Specific Jargon in Low-Resource Scenarios

Evaluierung und Verbesserung automatischer Spracherkennung für branchenspezifischen Fachjargon in datenarmen Szenarien

Author:	Leonhard Siegfried Stengel
Supervisor:	Prof. Dr. Florian Matthes
Advisor:	Alexandre Mercier
Submission Date:	13. November 2025

I confirm that this master's thesis in informatics is my own work and I have documented all sources and material used.

Munich, 13. November 2025 Leonhard Siegfried Stengel

Location, Submission Date Author

AI Assistant Usage Disclosure

Introduction

Performing work or conducting research at the Chair of Software Engineering for Business Information Systems (sebis) at TUM often entails dynamic and multi-faceted tasks. At sebis, we promote the responsible use of *AI Assistants* in the effective and efficient completion of such work. However, in the spirit of ethical and transparent research, we require all student researchers working with sebis to disclose their usage of such assistants.

For examples of correct and incorrect AI Assistant usage, please refer to the original, unabridged version of this form, located at this link.

Use of *AI Assistants* for Research Purposes

I have used AI Assistant(s) for the purposes of my research as part of this thesis.

Yes ◯ No ◯

Explanation: Gemini-2.5-Pro, Claude Sonnet 4 and ChatGPT-5 were used exclusively for language formulation, proofreading, translation, and text improvement of this thesis. These AI tools were not used to generate content, ideas, or research insights. All ideas, concepts, arguments, and research findings presented in this work are original thoughts and contributions of the author. The AI assistants were solely employed to enhance the linguistic expression and clarity of pre-existing ideas and to refine the textual presentation, while the intellectual content and substance of the thesis remain entirely the author's work.

I confirm in signing below, that I have reported all usage of AI Assistants for my research, and that the report is truthful and complete.

Munich, 13. November 2025 — Leonhard Siegfried Stengel

Location, Date — Author

Abstract

This thesis investigates the challenge of improving Automatic Speech Recognition (ASR) performance for industry-specific jargon in low-resource scenarios where in-domain audio data is unavailable. While state-of-the-art ASR systems achieve high accuracy on general speech, they often fail when encountering specialized terminology, creating a significant barrier to professional adoption.

Through a systematic literature review, this work addresses three fundamental research questions: how to define and represent jargon in speech datasets, which evaluation metrics best capture jargon recognition performance, and what methods can improve ASR systems without requiring in-domain audio data.

Three diverse datasets were collected representing aviation (FAA-Glossary), medical (UNITED-SYN-MED), and financial (Earnings-21) domains. Traditional Word Error Rate proved insufficient for evaluating jargon recognition, necessitating keyword-focused metrics including Precision, Recall, and F1-score. Baseline evaluation of state-of-the-art models (Whisper-large-v3, Canary-Qwen-2.5b, Gemini-2.5-Flash) revealed performance gaps in detecting jargon terms, with F1-scores as low as 43.5% for highly specialized medical terminology.

Three improvement methods were systematically evaluated: keyword-guided adaptation using keyword spotting, LLM-based error correction, and LLM zero-shot in-context learning. Zero-shot in-context learning emerged as the most promising approach, achieving consistent improvements across all datasets in recognizing industry-specific jargon terms. The method demonstrated absolute F1-score improvements of up to 20.9 percentage points but also introduced trade-offs in terms of increased operational costs.

This research provides practical insights for organizations seeking to deploy ASR technology in specialized domains, demonstrating that while significant challenges remain, viable solutions exist for improving jargon recognition without requiring domain-specific audio datasets.

Contents

1 Introduction

1.1 Motivation and Problem Statement

Personal voice assistants like Apple's Siri, Amazon's Alexa, and Google Assistant have become helpful tools for many people in their daily lives [1, 2]. But also in professional settings, Automatic Speech Recognition (ASR) systems are being integrated into workflows [3]. The advantages of ASR technology are compelling. Voice input is often faster, more natural, and more accurate than typing and it allows hands-free interaction with devices [4]. The implementation of ASR offers the potential to streamline workflows, increase efficiency and improve accessibility across businesses. A practical example can be seen in healthcare. Based on a case study, the implementation of ASR significantly reduces the clinical documentation burden, allowing clinicians to capture patient notes in real-time and up to 40% faster than typing. This increase in efficiency leads to more complete and accurate patient reports, which improves the flow of communication by reducing information-related delays and enables doctors to spend more time treating patients. [3]

However, a critical gap remains between the performance of these systems in everyday conversation and their reliability in specialized professional domains. State-of-the-art ASR models, despite being trained on vast datasets, often fail when confronted with industry-specific jargon like acronyms and complex terminology that lie outside their general training distribution [5, 6]. This can lead to transcription errors that range from comical to critical. An illustrative example demonstrates this challenge: when testing the sentence "Please create a BOM" with current ASR systems, including YouTube's automatic captioning [7] and Google Gemini's voice input feature, both systems transcribe BOM as "bomb" instead of correctly identifying it as the engineering acronym for Bill of Materials. Even though these terms sound similar, their meanings are vastly different. The substitution of BOM to bomb completely changes the sentence's intent from a routine manufacturing request to something potentially alarming.

The problem is particularly severe because jargon terms typically carry more semantic weight than common words [8]. In professional contexts, these specialized terms often represent critical concepts, procedures, or entities that are central to understanding and decision-making. When an ASR system misrecognizes a jargon term, it does not simply introduce a minor error but it can fundamentally alter the meaning of the entire communication. In high-stakes environments, such errors are unacceptable. An inaccurately transcribed medical diagnosis or a misunderstood air traffic control command can have severe consequences, undermining the very efficiency and safety the technology is meant to enhance. [8, 9]

This presents a significant barrier for companies and organizations that want to adopt ASR technology. According to an industry report by Speechmatics, a significant majority of businesses identify low accuracy as the main reason they hesitate to implement speech recognition technology [10]. The challenge lies in the scarcity of domain-specific audio data. Unlike general conversation datasets with thousands of hours of speech, specialized domains often lack sufficient transcribed audio for traditional model training approaches. Creating these audio datasets is either too expensive or the data is protected by privacy and confidentiality constraints [11]. This creates a low-resource scenario where organizations need effective ASR adaptation methods that work with minimal or no in-domain audio.

Given these constraints, this thesis focuses on lightweight adaptation methods that can improve jargon recognition without requiring in-domain audio datasets or extensive computational resources for model retraining. While existing research has explored various domain adaptation techniques, there remains a need for systematic evaluation of methods that can effectively handle jargon recognition in truly low-resource scenarios where no audio datasets are available. Based on a systematic literature review, this thesis explores three promising approaches, which are schematically illustrated in Figure 1.1. First, applying a keyword spotting adapter to a ASR system that provides contextual guidance to the model by detecting pre-defined jargon terms in the audio. Second, the use of a Large Language Model (LLM) as a post-processing step to correct domain-specific errors in the generated transcript. Third, a zero-shot in-context learning method that boosts the recognition of specific keywords by providing them as a list in the prompt to a multimodal LLM.

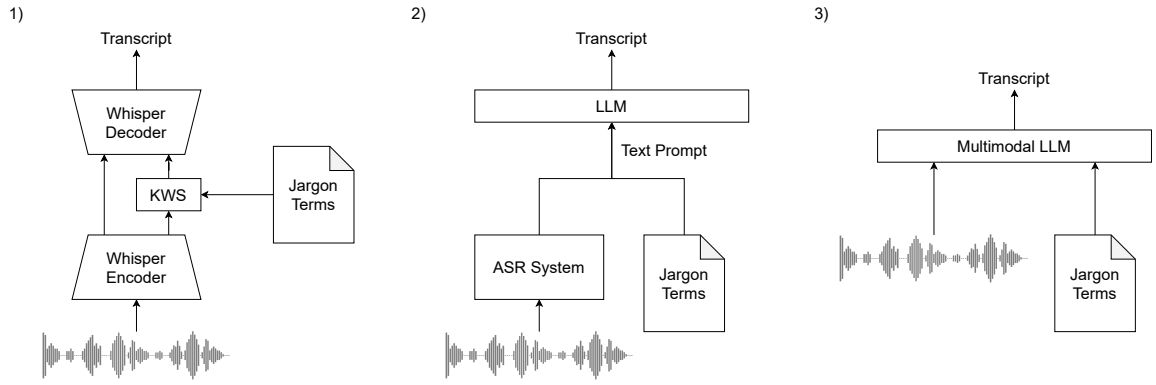

Figure 1.1: Approaches to improve ASR performance on jargon-heavy speech, illustrating 1) a keyword spotting approach, 2) LLM-based error correction, and 3) LLM in-context learning.

1.2 Research Questions

To guide this investigation, the thesis is organized around the following three research questions:

1. How can industry-specific jargon be defined and represented in speech datasets to enable benchmarking of ASR systems in domains?
2. Which evaluation metrics best capture the performance of ASR systems on jargon-heavy speech?
3. What methods can improve ASR performance in recognizing industry-specific jargon with no availability of in-domain audio data?

1.3 Structure of the Thesis

Following this introduction, Chapter 2 provides the theoretical foundation through a comprehensive literature review of fundamental ASR concepts and systematic research on promising low-resource adaptation techniques for improved jargon recognition. The subsequent chapters detail the methodology and experiments performed. Chapter 3 addresses how jargon can be represented in speech datasets by defining the term and presenting the datasets used for this study. With the data established, Chapter 4 outlines the evaluation metrics and benchmarks the baseline performance of several state-of-the-art ASR models on the jargon-heavy datasets. The core experimental work is presented in Chapter 5, which explores and evaluates three distinct methods for improving jargon recognition in low-resource scenarios. Chapter 6 presents the quantitative results of these experiments, followed by Chapter 7, which provides a detailed discussion that interprets the findings, considers their practical implications, and acknowledges the limitations of this work. The thesis concludes in Chapter 8 by summarizing the key contributions, revisiting the initial research questions, and offering an outlook on future research directions.

2 Related Work

This chapter reviews existing literature and research relevant to the challenges of ASR in handling industry-specific jargon. First it covers foundational concepts relevant for the problem at hand. Based on a systematic literature review, it then explores various speech datasets used for benchmarking ASR systems, discusses common evaluation metrics, and examines domain adaptation techniques aimed at improving ASR performance in specialized contexts.

2.1 Automatic Speech Recognition

Automatic Speech Recognition, also known as speech recognition or Speech-to-Text (STT), is a sub-field of computational linguistics that develops methods and technologies to translate spoken language into written text [12].

2.1.1 Basic Architecture of ASR Systems

An ASR system is composed of several key components that work together to convert spoken language into text, as illustrated in Figure 2.1. The architecture typically includes four main parts: signal processing and feature extraction, an Acoustic Model (AM), a Language Model (LM), and a hypothesis search decoder [12]:

- **Signal Processing and Feature Extraction:** This initial stage processes the raw audio input. It enhances the speech signal by reducing noise and channel distortions. From this, it extracts a sequence of feature vectors that capture the essential characteristics of the speech, making them suitable for the AM.
- **Acoustic Model (AM):** The AM links the extracted feature vectors to linguistic units, such as phonemes. It takes the feature sequence as input and calculates the probability that a given sequence of acoustic features corresponds to a particular sequence of phonemes.
- **Language Model (LM):** The LM evaluates the likelihood of a sequence of words. It learns statistical relationships between words from large text corpora, allowing it to predict which word is most likely to follow a given sequence. This helps the system distinguish between acoustically similar but linguistically different phrases.

- **Hypothesis Search (Decoder):** The final component hypothesis search integrates the information from both the acoustic and language models. It searches for the most probable sequence of words given the acoustic features by combining the scores from the AM and LM. The word sequence with the highest combined score is then output as the final transcription. [12]

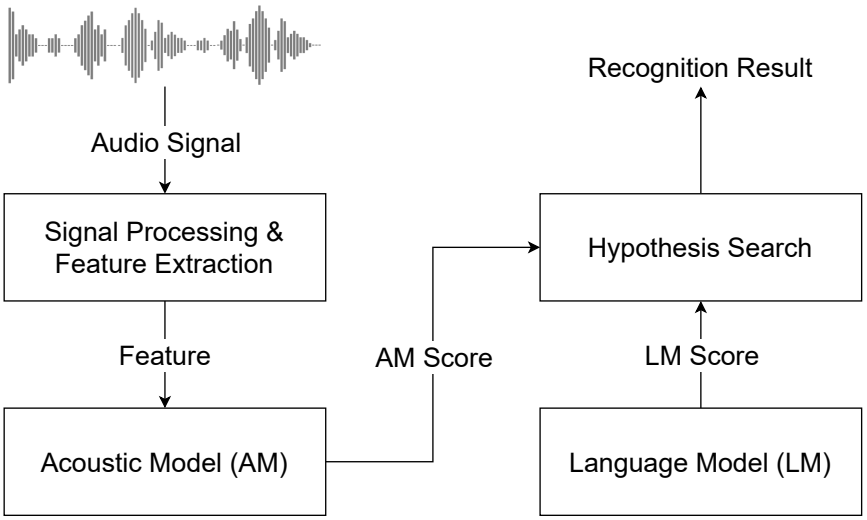

Figure 2.1: Basic Architecture of ASR Systems [12].

2.1.2 History of ASR Systems

While the basic architecture outlines the fundamental components of an ASR system, the specific methods and technologies used to realize these components have varied greatly throughout the history of the field and continue to evolve. The journey began with early pattern-matching systems, progressed to statistical methods like Hidden Markov Models (HMM), and has now entered the era of deep learning and neural networks, which power modern state-of-the-art systems [13].

Early Pattern-Based Systems

The earliest approaches to ASR in the 1950s and 1960s were primarily pattern-matching systems. These systems worked by matching the acoustic properties of a spoken word against a pre-stored template or pattern. [13]

A classic example is the Audrey system built by Davis, Biddulph, and Balashek in 1952 [14]. As one of the very first ASR systems [15], Audrey was designed to recognize isolated spoken digits from a single speaker. The system's methodology involved creating a reference pattern for each digit from zero to nine. These patterns were based on the unique acoustic

signature of the vocal sounds in each digit, specifically their formants. Formants are the resonant frequency peaks where sound energy is most concentrated. When a user spoke a digit, Audrey would analyze the sound to trace its formant pattern. This input pattern was then compared against the ten stored reference patterns and the digit corresponding to the closest match was selected as the output. For its designated speaker, it achieved an impressive accuracy of over 98% on single-digit recognition. [14]

However, these early systems faced two major challenges: speaker dependence and temporal variability. Speaker-dependent systems could only reliably recognize the voice of a single speaker to whom the system was adjusted on [13]. Temporal variability refers to the fact that people do not pronounce the same word identically or at the same speed in different repetitions. To address this, techniques like Dynamic Time Warping (DTW) were developed. DTW used dynamic programming to non-linearly align two speech patterns in time, allowing for a more flexible and robust comparison between the spoken word and the stored template. [16]

Statistical HMM-Based Systems

The 1980s marked a significant shift in ASR, moving from template-based methods to a statistical framework. The HMM became the dominant methodology, providing the foundation for ASR systems for the next two decades. [13]

A Hidden Markov Model is a doubly embedded stochastic process. It consists of an underlying Markov chain, which has a finite number of states, a state transition probability matrix, and an initial state distribution. This chain itself is not observable, it is hidden. What is observed is a sequence of outputs produced as the model moves between states. Each hidden state is characterized by an observation probability distribution, which defines the probability of emitting a particular observable output while in that state. [17]

In speech recognition, this framework can be applied to create an acoustic model. The hidden states are used to represent the underlying, abstract linguistic units, such as phonemes within a word. The observable outputs are the acoustic feature vectors extracted from the speech signal over time. Therefore, an HMM can model an entire word or phone as a sequence of hidden states that generates the sequence of acoustic features observed when that unit is spoken. [17]

While HMMs model the acoustic properties, a separate, often simpler, Markov Model like a bigram or N-gram model can be used to model the language component of an ASR system [18]. This language model represents the grammar or likelihood of word sequences, assigning a prior probability to indicate how likely a sequence is to occur [19].

The recognizer's final decision is based on finding the word sequence that maximizes the combined likelihood from both the acoustic model (how well the sound matches) and the language model (how probable the word sequence is) [17, 20].

Building on successes in speaker-dependent systems, researchers applied the HMM framework to the significantly harder task of speaker-independent phone recognition. For example, Lee and Hon used discrete HMMs with multiple codebooks of acoustic features to recognize English phones from continuous speech, demonstrating the robustness of the HMM framework. [20]

Neural Network-Based Systems

Following the dominance of HMMs, the field of ASR saw another major shift with the rise of neural networks and deep learning. Neural networks offered powerful new ways to model the complex relationships between acoustic and linguistic features [13].

The idea to use artificial neural networks was reintroduced in the late 1980s, following earlier attempts in the 1950s that had not produced notable results [13]. The appeal of neural networks lies in their ability to learn complex, non-linear mappings from input to output without requiring explicit feature engineering. In ASR, this meant it could be used to classify acoustic features into linguistic units. Also there was no need for carefully designed features for a HMM, as neural networks learn relevant features directly from the input data [12]. However, in spite of their effectiveness in classifying short-time units such as individual phonemes and isolated words, early neural networks were rarely successful for continuous recognition because of their limited ability to model temporal dependencies [13].

To overcome this, research focused on integrating neural networks with the established HMM framework. In these hybrid systems, the neural network would function as a more powerful acoustic model, replacing the traditional probability distributions to better classify acoustic features. The HMM structure, however, retained its role in handling the temporal sequence, while a separate statistical language model was still needed to provide the probability of different word sequences [12, 13].

End-to-end Systems

Latest developments in ASR have centered on End-to-End (E2E) models. These models aim to simplify the ASR pipeline by directly mapping acoustic features to text transcriptions in a single neural network architecture, eliminating the need for separate acoustic and language models [21]. There are several prominent architectures for E2E ASR systems. These include early approaches like Connectionist Temporal Classification (CTC) and more recent attention-based models.

Attention-based E2E architectures use an encoder-decoder structure. First, the encoder processes the entire audio input into a sequence of high-level representations. Then, an autoregressive decoder generates the output text one label at a time. The crucial component linking the two is the attention mechanism [22]. At each step of the decoding process, the attention mechanism computes weights over all the encoded audio frames, allowing the decoder to look back and dynamically focus on the most relevant part of the audio needed

to predict the next text label. These weights are used to create a context vector, which the decoder combines with its own internal state to make a prediction. This process repeats, with the attention dynamically shifting, until a special <eos> (end-of-sentence) token is produced. Because the decoder's choice is based on previously generated labels, it learns an implicit language model. This powerful attention-based architecture was a key driver in bringing new performance highs to the field of ASR. [21]

Building on the success of these architectures and their implicitly learned language models, the most recent developments have focused on explicitly integrating the vast contextual power of dedicated, pre-trained LLMs. LLMs are advanced artificial intelligence systems, trained on extensive text corpora, that have demonstrated exceptional capabilities in contextual understanding and multitask performance within Natural Language Processing (NLP) [23]. Researchers are now actively integrating these powerful models into the domain of speech understanding, particularly for ASR tasks. This integration has led to the development of Speech-LLMs which leverage the deep reasoning abilities of LLMs to enhance transcription accuracy and robustness. The most common architecture for these systems follows a unified pipeline. In this approach, audio features are first extracted from the speech signal. Then, a fusion step aligns the audio modality with the text modality, effectively turning the core model into a multimodal LLM capable of processing and understanding both speech and text inputs. By using the LLM as the central reasoning component, these Speech LLMs can better understand context and have consistently achieved higher accuracy on ASR benchmarks compared to conventional models. [24]

2.1.3 State-of-the-Art ASR Systems

Today E2E transformer and LLM based architectures dominate the field of ASR, achieving state-of-the-art performance across various benchmarks [24, 25]. In the following sections, three of the most prominent modern ASR capable systems are presented.

OpenAI Whisper

One of the most famous architectures is Whisper. On the online platform Hugging Face, the current version of Whisper whisper-large-v3 is one of the most popular models for the task of ASR [26]. Whisper is an open-source ASR model developed by OpenAI and was first released in 2022 [27]. It is built on an encoder-decoder transformer architecture. The initial version was trained on a dataset of 680,000 hours of multilingual supervised data collected from the web. The most recent version of whisper large-v3 was trained on more than 5 million hours of audio data and has a size of about 1.5 billion parameters [28, 29]. Its architecture is illustrated in Figure 2.2. First the audio input is split into 30-second chunks and converted into a spectrogram, which is then processed by the encoder to produce high-level feature representations. The decoder takes these features and generates the corresponding text transcription autoregressively, using an attention mechanism to focus on relevant parts of the audio input at each step. A defining feature of Whisper's architecture is its multitask format.

The model is trained to perform multilingual speech recognition, speech translation, spoken language identification, and voice activity detection. These different tasks are specified by feeding a unique sequence of special tokens to the decoder, allowing one model to replace a complex pipeline of different components. [27]

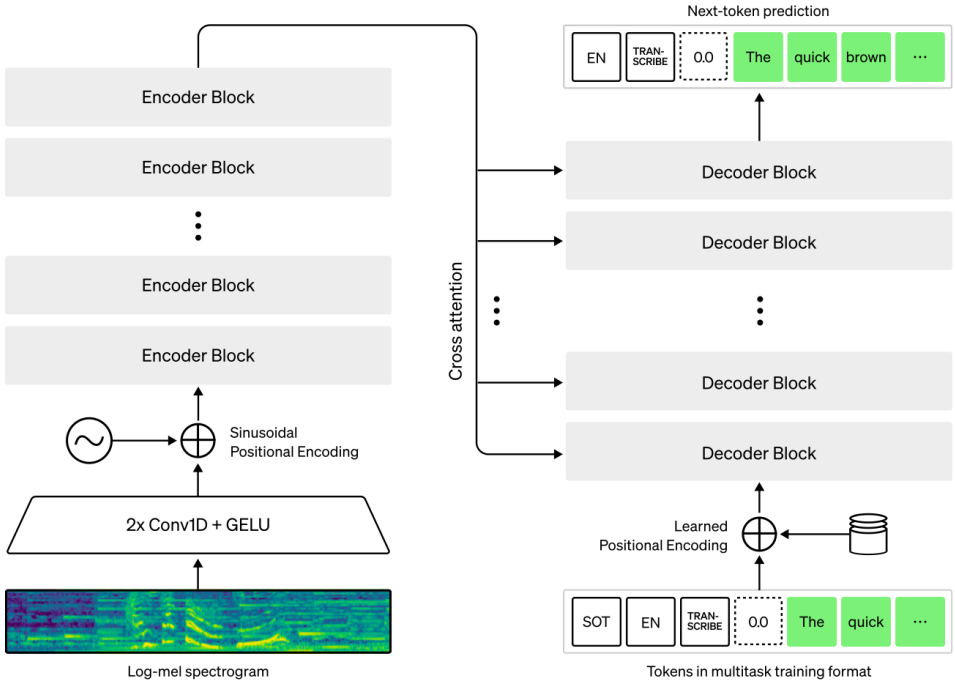

Figure 2.2: Whisper ASR Architecture [30].

NVIDIA Canary-Qwen-2.5b

The Hugging Face Open ASR Leaderboard is a platform that ranks and evaluates speech recognition models. The selection of models is limited to those that are available on the Hugging Face Model Hub [25]. The leaderboard benchmarks models on both english and several multilingual datasets, ranking them based on their performance, which is measured by the average Word Error Rate. The leaderboard also reports the model's processing speed. As of August 2025, the top-performing model on the english leaderboard is Canary-Qwen-2.5b by Nvidia, achieving an average Word Error Rate of 5.63% across various datasets [25]. Canary-Qwen-2.5b is a 2.5-billion-parameter Speech-Augmented Language Model (SALM) designed for high-performance english ASR. Its hybrid architecture is constructed from two base models: a specialized speech encoder, and the Qwen3-1.7B LLM, which functions as the transformer decoder. These components are integrated via a linear projection layer that maps the audio representations from the encoder into the LLM's embedding space. The audio embeddings are then concatenated with standard text token embeddings for example

"Transcribe the following:". Figure 2.3 illustrates the architecture of such a Speech-Augmented Language Model (SALM). This architecture enables two distinct operational modes: an ASR mode, which performs pure transcription by prompting "Transcribe the following:", and an LLM mode, which leverages the original capabilities of the Qwen3-1.7B LLM to perform text-based tasks like summarization or Q&A on the generated transcript. Key limitations include its exclusive focus on English and its ASR-oriented design, which does not extend the full LLM capabilities directly to the speech modality but only to the generated transcript. [31, 32]

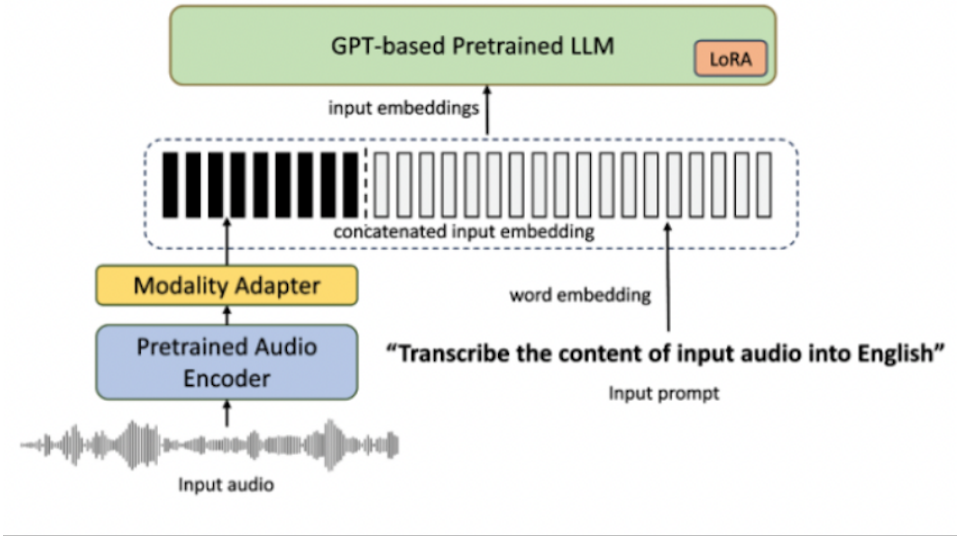

Figure 2.3: SALM Architecture [33].

Google Gemini

While Canary-Qwen-2.5b offers limited multimodal capabilities, Google's Gemini models are designed from the ground up as multimodal LLMs that can process and reason over text, audio, image, and video inputs [34].

The Gemini 2.5 models utilize a sparse Mixture of Experts (MoE) transformer architecture. Unlike traditional deep learning models that reuse the same parameters for all inputs, this approach works by activating only a subset of the model's parameters for each input token. This is achieved through a routing mechanism that selects the most relevant sub-networks or experts based on the input context. [34]

The Switch Transformer paper introduced a simple and efficient version of this architecture as illustrated in Figure 2.4. This design implements the MoE concept by replacing the standard dense feed-forward layer in a Transformer with a new layer. This layer consists of multiple

expert sub-networks. The core mechanism involves a router that processes each incoming token representation independently. Based on the token, the router calculates probabilities and selects a single expert to process that token. The token is then sent to its designated expert for computation. The final output of the layer for that token is the result from the selected expert, multiplied by the probability assigned by the router. [35] By doing so, the model can scale to billions of parameters without a proportional increase in computational cost for each inference, as only a fraction of the model is used at any given time [34].

By using this architecture the Gemini models demonstrate exceptionally strong audio processing capabilities, particularly in ASR. These capabilities are derived from a large-scale pre-training dataset that spans over 200 languages. As a result, Gemini-2.5-Pro achieves state-of-the-art performance on public ASR benchmarks. It also supports streaming audio inputs and outputs, making it suitable for real-time applications like chatbots and voice assistants [34].

The Gemini-2.5 family is offered in several versions, with the primary distinction being between Pro and Flash models. Gemini-2.5-Pro is designated as the most intellectually capable model, achieving top-tier results on complex challenges, especially high-level coding and reasoning benchmarks. In contrast, Gemini 2.5 Flash is engineered as a hybrid reasoning model that optimizes the balance between performance, cost, and speed. It delivers impressive reasoning capabilities while demanding significantly less computational power and offering lower latency than the Pro version, making it suitable for high-volume or time-sensitive applications. [34]

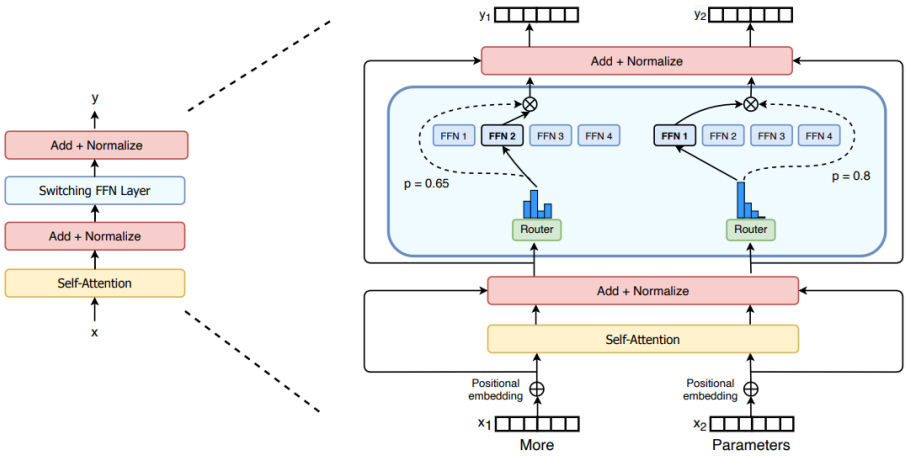

Figure 2.4: Switch Transformer encoder block [35].

2.2 Speech Datasets for ASR

To train and evaluate large deep learning-based ASR systems, vast amounts of high-quality speech data are required [25]. The lack of available datasets has been identified as a factor that can prevent advancements in the field [36]. A typical speech dataset consists of a collection of audio files paired with their corresponding text transcriptions. To ensure that an ASR system is robust and performs well in real-world scenarios, these datasets must cover a wide range of variations. Key aspects include acoustic conditions like audio quality and noise, speaker characteristics such as dialect, and linguistic features like the specific language or industry-specific jargon [5, 25].

A general-purpose ASR system should be robust to these variations to be effective across different applications [25]. Several large-scale datasets have been developed to facilitate research and development in ASR. Some of the most commonly used datasets in academic and industry research include LibriSpeech, VoxPopuli and FLEURS.

- **LibriSpeech:** The LibriSpeech corpus is a freely available dataset of read English speech, specifically designed for training and evaluating ASR systems. It contains 1,000 hours of audio, derived from public domain audiobooks from the LibriVox project. The corresponding text transcriptions are sourced from Project Gutenberg. The corpus is partitioned into multiple training, development, and test sets, which are further divided into the categories clean and other to represent different levels of audio quality and speaker clarity [37].
- **VoxPopuli:** VoxPopuli is a large-scale, open-access multilingual speech corpus sourced from 2009-2020 European Parliament event recordings. It provides 400K hours of unlabeled speech data in 23 languages, making it the largest open dataset for unsupervised representation learning at the time of its publication. The corpus also contains 1.8K hours of transcribed speech in 16 languages derived from the European Parliament plenary sessions. A key feature is its 17.3K hours of aligned speech-to-speech data, which pairs the transcribed source speeches with their simultaneous oral interpretations into 15 target languages. This dataset is designed to advance research in representation learning, semi-supervised learning, and speech interpretation [38].
- **FLEURS:** FLEURS is a parallel speech dataset in 102 languages designed to evaluate universal speech representations, particularly in few-shot learning scenarios. It was sourced by building on the FLoRes-101 machine translation benchmark, which contains sentences from English Wikipedia that were professionally translated. To create FLEURS, natural human speech recordings of these translated sentences were collected for each language. The corpus provides approximately 12 hours of speech supervision per language, and it is designed to catalyze research in low-resource speech understanding across tasks like ASR, speech language identification, and translation [39].

These datasets provide a solid foundation for training and benchmarking ASR systems. They are widely used in research to evaluate model performance and drive advancements in speech recognition technology [25, 27, 32, 40].

While general-domain datasets are essential for building foundational models, they often lack the specialized terminology needed to evaluate ASR performance on jargon-heavy speech. To address this challenge, specialized datasets are required that contain a significant amount of industry-specific language.

Some datasets are based on real-world recordings from specific industries and domains. For example, the UWB-ATCC dataset contains real recordings from air traffic control communications, which are rich in aviation jargon [41]. Another example is Earnings-21, which offers authentic recordings of corporate earnings calls, providing a valuable resource for financial jargon [42].

Due to the scarcity of large, publicly available jargon-heavy datasets, a common approach is to create synthetic datasets using text-to-speech synthesis. This approach allows researchers to generate large amounts of domain-specific speech data by converting text containing specialized terminology into spoken audio. This method follows a two-step process: first, relevant text data is collected from industry-specific sources, such as medical journals or technical manuals. Next, this text is converted into speech using high-quality text-to-speech systems. This synthetic speech data can then be used to train or fine-tune ASR models, having the respective transcriptions inherently available from the original text. Research has shown that using synthetic speech to fine-tune ASR systems for domain adaptation can yield positive results [43, 36, 44]. An example of such a synthetic dataset is UNITED-SYN-MED, which was created from medication terms and drug names [11].

2.3 Evaluation of ASR Systems

Evaluating the performance of ASR systems requires a comprehensive approach that goes beyond simple transcription accuracy. While quantitative metrics provide essential benchmarks for comparing systems and tracking improvements, the choice of evaluation method must align with the specific application context and requirements. This section examines the primary metrics used in ASR research, which include traditional edit-distance measures, semantic metrics that assess meaning preservation, and keyword-focused assessments that evaluate performance on domain-specific terminology. Each type of metric serves distinct purposes and provides different insights into system capabilities, particularly when dealing with specialized domains where accurate recognition of technical jargon is paramount. [45]

2.3.1 Edit-Distance-Based Metrics

One of the most common and widely accepted metric for evaluating ASR systems is the Word Error Rate (WER). It provides a simple, interpretable measure of transcription accuracy by

quantifying the number of errors in the ASR system's output (the hypothesis) compared to a verified ground-truth transcript (the reference). The WER is derived from the Levenshtein distance, which calculates the minimum number of edits required to change one sequence into another. Specifically, it counts the number of substitutions, deletions and insertions at the word level. [45] The formula is:

$$\text{WER} = \frac{S + D + I}{N} \tag{2.1}$$

where:

- S = number of substitutions,
- D = number of deletions,
- I = number of insertions,
- N = number of words in the reference [45].

A lower WER indicates a more accurate transcription, with a WER of 0 representing a perfect match. The higher the WER, the more errors the system made in the transcription process. [45] A related metric is Word Recognition Rate (WRR), which is simply calculated as

$$\text{WAcc} = 1 - \text{WER}. \tag{2.2}$$

It therefore follows the properties of classical accuracy metrics with 100% being a perfect match and 0% indicating no overlap. [46, 47]

Another commonly used edit-distance-based metric is the Character Error Rate (CER). Similar to the WER, the CER operates on the same principle, calculating the number of substitutions, deletions and insertions required to transform the hypothesis into the reference. However, it performs this calculation at the character level instead of the word level. The formula is analogous to the WER formula:

$$\text{CER} = \frac{S_c + D_c + I_c}{N_c} \tag{2.3}$$

where S_c, D_c, and I_c are the number of substituted, deleted and inserted characters and N_c is the total number of characters in the reference transcript [45].

The CER metric is especially relevant for languages that employ complex scripts, such as Chinese or Arabic. In these writing systems, a single character mistake can completely alter the meaning of a word, making character-level accuracy crucial [48]. While CER provides a more detailed error analysis than WER, it is also more vulnerable to being influenced by minor differences in punctuation or formatting if normalization is not performed carefully [45].

2.3.2 Semantic Metrics

A key limitation of WER and CER is that they treat all errors as equally important. A substitution of a critical jargon term like a medical diagnosis is penalized the same as a substitution of a minor filler word. This can be misleading, as the functional impact of these two errors is vastly different. To address this, semantic metrics have been developed to evaluate whether the meaning of the hypothesis is equivalent to the reference, even if the exact words differ. [45]

A classic example is BERTScore, which leverages the contextual embeddings from transformer models like BERT to compute the cosine similarity between the reference and hypothesis transcript representations. This allows it to recognize synonyms and paraphrasing, providing a better measure of semantic alignment than simple word overlap. [45]

More recently, newer methods use LLMs to act like human evaluators. For example a LLM can be trained to read the original text and the transcription and then decide if the core meaning was kept or lost. This provides a more realistic measure of a transcription's quality, which is especially helpful for challenging situations like transcribing low-resource domains or languages [49]. This approach has proven useful in other specialized fields, too. For example, when transcribing communication between pilots and air traffic controllers, researchers found that simply counting errors was insufficient. They developed a GPT-based score to weigh the contextual importance of transcription mistakes. This LLM-based metric proved much more effective than WER, especially when evaluating model performance on out-of-distribution data where capturing the correct meaning of specialized terms is more important than achieving a verbatim transcription. [50]

2.3.3 Keyword-Focused Metrics

When a specific set of domain-relevant keywords is critical to the application, evaluating the ASR system's ability to accurately recognize these terms becomes essential. This is particularly true in specialized fields like medicine, law, or aviation, where misrecognizing key terminology can lead to significant misunderstandings or errors. In this case the recognition performance of a predefined set of domain-specific keywords in ASR can be measured by using precision, recall, and the F1-score. These metrics are calculated from the classifications shown in the confusion matrix in Table 2.1. This matrix categorizes the ASR output for each target keyword by comparing the system's transcript (prediction) to the ground truth reference [32, 40]:

		Reference	
		Keyword Present	**Keyword Absent**
Prediction	**Keyword Present**	True Positive (TP)	False Positive (FP)
	Keyword Absent	False Negative (FN)	True Negative (TN)

Table 2.1: Confusion matrix for keyword-based evaluation.

- **True Positive (TP):** The system correctly transcribes a keyword that is present in the reference.
- **False Positive (FP):** The system transcribed a keyword, but that keyword was not present in the reference.
- **False Negative (FN):** The system fails to transcribe a keyword that was present in the reference.
- **True Negative (TN):** The system correctly does not transcribe a keyword, and no keyword was present in the reference at that position.

Using these counts, Precision measures the proportion of keywords identified by the system that were correct. In other words, it reflects how many of the keywords the system recognized were actually present in the reference. It penalizes hallucinating keywords that are not in the reference. Precision is calculated as follows [51]:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{2.4}$$

Recall measures the proportion of all actual keywords in the reference that the system successfully recognized. It reflects the system's ability to find all relevant keywords, penalizing missed keywords. Recall is calculated as follows [51]:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{2.5}$$

The F1-score then combines these two metrics into a single, balanced measure by calculating their harmonic mean. This is useful because high precision can sometimes be achieved at the cost of low recall, and vice-versa. The F1-score provides a balanced assessment of the system's performance on the keyword set. The formula to calculate the F1-score is [51]

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{2.6}$$

2.3.4 Normalization

A significant challenge in calculating metrics for ASR performance is that spoken language can be written down in multiple, equally valid ways. For instance, a speaker might say "it's five dollars", which could be transcribed as "it's 5 dollars", "it is five dollars" or "it is $5". If the reference transcript uses one form and the ASR system outputs another, the system would be unfairly penalized for producing a semantically identical but textually slightly different output. This makes text normalization a crucial pre-processing step before any error rate calculation. [52]

Normalization involves applying a consistent set of rules to both the reference and hypothesis transcripts to reduce ambiguity and ensure a fair comparison. Classic normalization steps include [46, 52]:

- Converting all text to a single case (typically lowercase).
- Removing all punctuation marks.
- Expanding common contractions (e.g., “it’s” to “it is”).
- Converting numerals to their word form (e.g., “5” to “five”).
- Removing filler words, hesitations, and discourse markers (e.g., “uhm”, “ah”).

By transforming both texts into a canonical format, normalization ensures that the calculated error rate reflects genuine recognition errors rather than stylistic variations. Modern libraries, such as JiWER, are commonly used to handle both the complex normalization process and the subsequent calculation of metrics like WER and CER, simplifying the evaluation pipeline [46].

2.4 Domain Adaptation in ASR

Domain adaptation is a subfield of machine learning that addresses the challenge of applying a model trained on a specific source domain to a different target domain [53].

2.4.1 Challenge of Domain Shift

The fundamental problem domain adaptation seeks to solve is domain shift, a phenomenon where the statistical distribution of data in the target domain differs significantly from that of the source domain on which the model was trained. This discrepancy can lead to a decline in model performance, as the patterns learned by the model may no longer be applicable or effective in the new context. For instance, an image classifier trained exclusively on high-quality, professional product photos (the source domain) would likely fail to accurately identify objects in casual, low-light smartphone pictures (the target domain) due to differences in lighting, composition, and image quality. [54]

In the context of ASR, domain shift is a persistent and significant issue. A state-of-the-art ASR system trained on large, general-domain corpora such as LibriSpeech will often exhibit a substantial drop in performance when deployed in specialized environments. This performance degradation can be attributed to several factors [5]:

- **Acoustic Conditions:** There can be a huge difference between the clean, studio-quality recordings typical for training data and the acoustic environments of real-world applications. Target domains might include noisy factory floors or radio channel distortions in pilot communication.

- **Speaker Characteristics:** Speaker accents, dialects or speaking rates can vary dramatically between the general population used for training and the specific user group in a target domain.
- **Linguistic Features:** This is a critical factor for the focus of this thesis. Specialized domains are often characterized by unique vocabularies, including industry-specific jargon, technical terms or acronyms that are rare or non-existent in general-purpose training data. An ASR model that has never encountered these out-of-vocabulary terms will inevitably fail to recognize them correctly.

The core challenge of domain shift in ASR is that the model's learned acoustic and language representations from the source domain do not generalize well to the distinct characteristics of the target domain [5].

2.4.2 Classic Adaptation Techniques

To mitigate the effects of domain shift, several techniques have been developed to perform domain adaptation. Classic approaches are primarily centered on the concept of transfer learning, which leverages the knowledge captured in a pre-trained model and applies it to a new task. These pre-trained models are typically trained on large, general-domain corpora like LibriSpeech or Common Voice. A specific and widely used form of transfer learning is fine-tuning. In this process, the parameters of the pre-trained model are further trained on a smaller, domain-specific dataset. This allows the model to adjust its internal weights to the nuances of the target domain without the need to learn everything from scratch, which would require a vast amount of data. [55]

A major challenge in ASR domain adaptation is the common problem of not having enough data. Acquiring and accurately transcribing large volumes of in-domain audio data is often too expensive, time-consuming, and impractical. In specialized fields the amount of available in-domain data is typically very limited. This transforms domain adaptation for ASR into a low-resource problem, where the central goal is to effectively adapt a powerful, general-purpose model using only a limited and sometimes tiny amount of target-domain data. [11, 53]

This limitation has pushed researchers to develop new strategies that can achieve better performance with very little data. The work in this thesis focuses on these low-resource methods and looks for solutions that are practical for real-world use.

2.5 Systematic Literature Review

To establish a comprehensive understanding of the current state-of-the-art in domain adaptation for ASR, a Systematic Literature Review was conducted. The primary objective of this review was to identify recent research focused on enhancing ASR performance for industry-specific jargon and specialized terminology. The review followed a clear and structured approach to ensure comprehensive coverage of the topic according to the systematics presented in [56].

This process was guided by the PICOC (Population, Intervention, Comparison, Outcome, and Context) framework, which serves to deconstruct a systematic literature research's objectives into searchable keywords. Each PICOC category can be explained as follows [56]:

- **Population:** Refers to the specific application area that is the focus of the research.
- **Intervention:** The methodology, tool, or technology being studied that addresses a particular issue.
- **Comparison:** Represents the alternative technology that the main intervention is being compared against.
- **Outcome:** The measurable results or consequences of applying the intervention.
- **Context:** Defines the environment or setting in which the research and any comparisons take place.

The keywords for each PICOC category were carefully selected to align with the primary research question, which focuses on adapting ASR systems for specialized terminology. The Population was defined as ASR since it is the core technology under investigation. The Intervention of interest is "Adaptation", as the review seeks methods to modify existing models. This is Compared against baseline "state-of-the-art ASR systems" to measure improvement. The desired Outcome is "improved recognition performance", which is the primary goal of any adaptation technique. Finally, the Context is specified as "industry-specific jargon" to narrow the search to the specific challenge being addressed. Table 2.2 provides a summary of the primary keywords used for the PICOC categories.

PICOC	**Keyword**
Population	Automatic Speech Recognition
Intervention	Adaptation
Comparison	State-of-the-art ASR Systems
Outcome	Improved recognition performance
Context	Industry-specific jargon

Table 2.2: Defining PICOC keywords for systematic literature research.

To broaden the search scope and capture a wider range of relevant literature, a set of synonyms was also defined for each category. For instance, "customisation" and "optimisation" were

used as synonyms for the Intervention keyword "adaptation", while "low-resource domain" and "out-of-vocabulary" were used for the Context keyword "jargon". The full list of synonyms for each PICOC category can be found in the Appendix at A.1.

The search was conducted using the Scopus digital library, which was chosen because it is currently the largest multidisciplinary abstract and citation database, offering extensive coverage of peer-reviewed literature [56]. To ensure the review focused on the most current and relevant research, the following inclusion and exclusion criteria were strictly applied:

- **Time Period:** Only publications from the year 2023 onwards were considered. This temporal constraint was deliberately chosen to focus on research conducted after the influential release of OpenAI's Whisper model in late 2022 [27].
- **Language:** Only articles published in English were included to ensure consistency and avoid translation-related ambiguities.
- **Accessibility:** Only articles that were fully accessible through the Scopus database were considered for the final analysis.

These criteria, combined with the PICOC keywords and their defined synonyms, were formulated into a detailed query string, which can be found in the Appendix at A.2. Executing this search query on the Scopus database yielded an initial result of 55 papers.

To further refine the search results and ensure the relevance of the selected papers, a Quality Assessment checklist was employed. This checklist was designed to evaluate each paper against specific criteria related to the research focus. The Quality Assessment was performed on the abstracts of the 55 papers, allowing for a preliminary screening before delving into full-text reviews. The checklist included the following questions:

- Does the article propose or compare specific frameworks, tools, or methodologies for domain adaptation in ASR?
- Does the proposed approach explicitly aim to improve the recognition performance of domain-specific terms or jargon?
- Is the article's primary focus on low-resource scenarios with no in-domain audio data available?
- Is the code or model made available by the authors, and are the reported results reproducible?

Applying this quality assessment checklist as a filtering mechanism resulted in a final selection of 10 highly relevant papers. These selected articles were then subjected to a thorough and detailed examination to get a comprehensive understanding of the current state of research in the field. While the systematic literature review identified ten highly relevant papers, the scope and time constraints of this thesis necessitated a selection for implementation and experimentation. The three chosen papers represent distinct and innovative strategies for domain adaptation in ASR, covering different stages of the recognition pipeline. The first

approach focuses on guiding the ASR model with a keyword spotter, the second utilizes an LLM for post-processing error correction, and the third integrates keyword boosting directly into a SALM architecture through in-context learning. These methodologies were selected for their practical relevance, clear implementation pathways, and their representation of the current state-of-the-art in low-resource domain adaptation. Although other identified papers also presented valuable techniques, this thesis will concentrate on these three approaches to provide a focused and in-depth analysis. The following sections summarize the methodologies and findings from these papers.

2.5.1 Keyword-Guided Adaptation with Keyword Spotting

The paper by Shamsian et al. addresses the challenge in ASR of performance degradation when encountering specialized jargon. Their methodology proposes a novel contextual biasing technique that uses a Keyword Spotting (KWS) model as adapter in the Whisper architecture to guide the ASR transcription. [40]

Figure 2.5 illustrates the overall architecture of this approach. The core idea is to leverage Whisper's powerful encoder for two purposes simultaneously. First, the encoder representation is fed to a KWS model (AdaKWS) to identify the presence of pre-defined, domain-specific keywords in the audio. Second, the same encoder representation is passed to the Whisper decoder for transcription, as usual. [40]

The innovation lies in how the output of the KWS model is used. The keywords detected by the KWS model are formatted into a prompt, which is then passed to the Whisper decoder. This prompt acts as a strong contextual signal, guiding the decoder to recognize and transcribe the specialized jargon correctly. [40]

In a practical application, the methodology requires two primary inputs: the raw audio stream and a pre-defined list of domain-specific keywords that the user wishes to prioritize. The KWS model continuously scans the audio for these terms, and upon detection, dynamically generates the guidance prompt for the ASR decoder. This makes the system highly adaptable to specific tasks, such as recognizing technical part numbers in a factory or specific medical terminology in a hospital. [40]

The authors propose and implement two distinct approaches for integrating this keyword guidance:

- **KG-Whisper (Keyword-Guided Whisper):** This method involves fine-tuning the entire Whisper decoder. While the encoder remains frozen, the decoder's parameters (approx. 906M) are updated to learn how to best utilize the contextual prompts provided by the KWS model.
- **KG-Whisper-PT (Keyword-Guided Whisper with Prompt-Tuning):** This is the approach illustrated in Figure 2.5. It is a significantly more parameter-efficient alternative to fine-tuning the entire decoder. In this approach, the entire Whisper model (both

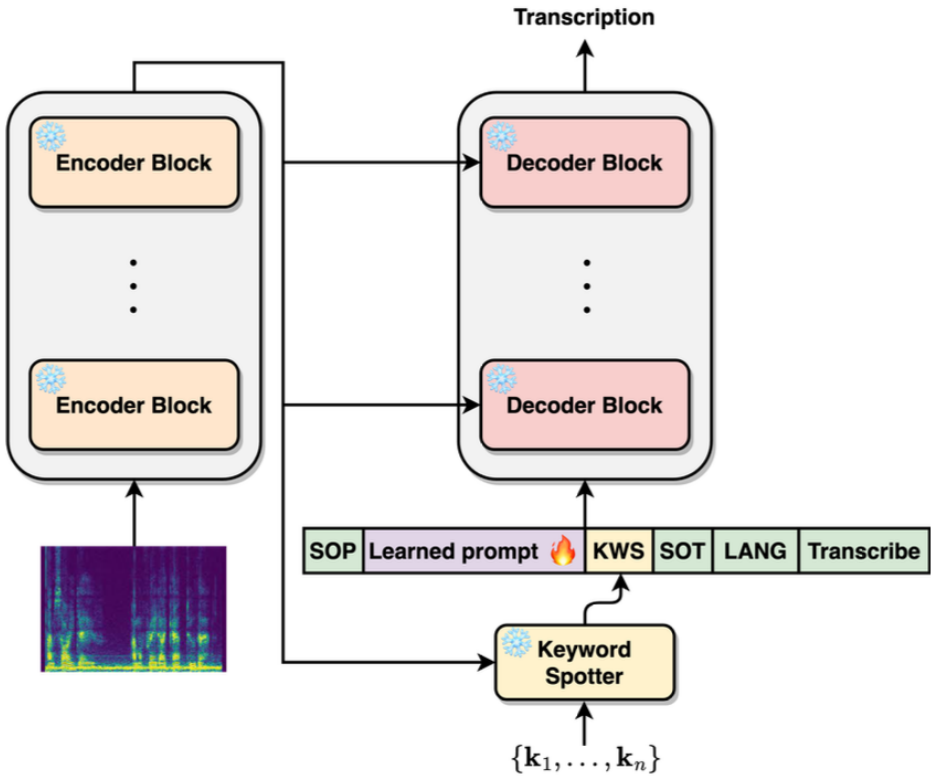

Figure 2.5: KG-Whisper-PT Architecture [40].

encoder and decoder) is kept frozen. Instead of tuning the model, a small, learnable prompt prefix (e.g., 12 tokens, totaling only ~15K parameters) is trained. This prefix is added before the KWS-generated keyword prompt, effectively teaching the frozen model how to interpret and follow the keyword guidance without altering its original weights.

Evaluation was conducted using two primary metrics: WER for overall transcription accuracy, and the F1 score to measure the specific accuracy of the target keywords. For the evaluation, keywords were programmatically selected from the datasets by sampling terms with high TF-IDF (term frequency-inverse document frequency) scores, which effectively identifies domain-specific jargon terms. The methods were then tested across several diverse datasets, including Voxpopuli, UWB-ATCC, Medical (patient reports), and Fleurs, to demonstrate in-domain performance and out-of-domain generalization. [40]

Key findings from their experiments show that both methods substantially outperform the Whisper baselines. As baseline they used the version large-v2 of the Whisper ASR models. On the Voxpopuli dataset, Whisper-large-v2 achieved a WER of 13.39% and a keyword F1 score of 81.77%. In comparison, KG-Whisper reduced the WER to 9.78% (a 3.61% absolute improvement) and increased the F1 score to 91.54% (a 9.77% absolute improvement). The

highly efficient KG-Whisper-PT method achieved a WER of 11.10% (a 2.29% absolute improvement) and a remarkable F1 score of 95.25% (a 13.48% absolute improvement), demonstrating results comparable to, and in terms of F1 score better than, the full fine-tuning approach. This methodology proved robust, demonstrating significant improvements in challenging out-of-domain scenarios, achieving an average WER improvement of 5.1% over Whisper in unseen language generalization tasks. [40]

This methodology was developed by researchers at aiOla Research. The company offers a free Application Programming Interface (API) trial for its ASR system, allowing users to evaluate this keyword-guided adaptation approach. [57]

2.5.2 LLM-Based Error Correction

The paper by Matasyoh et al. investigates a different approach to improving ASR performance in specialized fields. Instead of modifying the ASR model itself, this methodology uses a LLM as a post-processing error correction module to clean up the output of an existing ASR system. The paper is specifically focusing on the surgical domain. [58]

In this practical application, the system first transcribes speech using a standard ASR model (e.g., Whisper-medium or ASR-LibriSpeech). The resulting text transcription, which may contain errors due to domain-specific jargon or accents, serves as the primary input to the error correction module. For post-correction the transcript is passed to a LLM (GPT-3.5 or GPT-4) with a specific prompt. [58]

The authors' methodology centered on finding the most effective prompting technique to guide the LLM in correcting the transcription. They experimented with several methods: [58]

- **Zero-Shot:** Instructing the LLM to correct errors without providing any examples or domain context.
- **Few-Shot (with Leading Questions):** Providing the LLM with context by asking domain-related questions (e.g., "Do you specifically understand the cerebral ventricular anatomy?").
- **Few-Shot (with Medical Terms):** This was the most successful approach. The prompt provided a list of sample domain-specific medical terms (e.g., names of intracranial structures) to help the LLM identify and correct misspelled jargon.

The primary metric used for evaluation was WER. The experiments were conducted on a self-curated dataset tailored to the surgical domain, which was generated using Google Text-to-Speech to create audio files for 80 descriptions of intracranial structures, further augmented with 7 different English dialects. [58]

Key findings from their experiments demonstrated that the initial ASR model choice was critical, with Whisper (11.93% WER) significantly outperforming ASR-LibriSpeech (32.09%

WER). The most substantial improvement came from the few-shot with medical terms prompting method. The combination of using Whisper for transcription and GPT-3.5 for correction guided by medical terms resulted in a 64.29% relative reduction in WER (from 11.93% down to 4.26%) compared to Whisper's original output. Using the more advanced GPT-4 model yielded even further improvements, achieving a 67.66% relative WER reduction with Whisper. [58]

2.5.3 Keyword-Boosting via Zero-Shot In-Context Learning

Chen et al. present a SALM which leverages a pre-trained text LLM to perform various speech tasks, including keyword boosting, through in-context learning. This approach differs by directly integrating speech information into the LLM framework rather than solely relying on post-processing or modifying the ASR decoder only. [32]

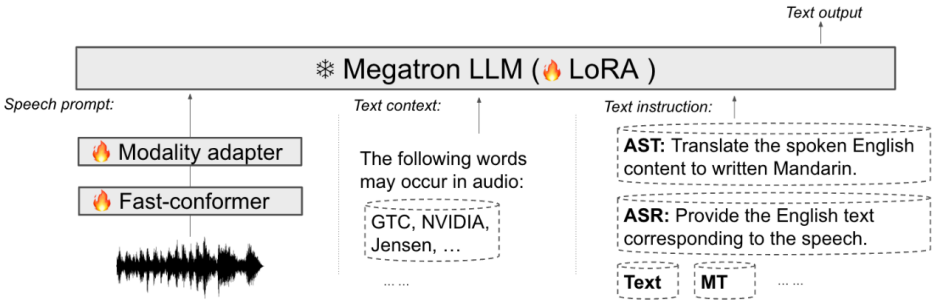

Figure 2.6: SALM for keyword boosting via in-context learning [32].

The SALM architecture, illustrated in Figure 2.6, consists of a frozen text LLM, a pre-trained audio encoder, a trainable modality adapter to bridge the speech and text representations, and trainable LoRA layers added to the LLM to adapt it for speech tasks. The model is trained using multitask supervised speech instruction tuning, where paired speech/text data from ASR and Speech Translation tasks are presented with task-specific instructions (e.g., "Provide the English text corresponding to the speech"). [32]

A key contribution of the paper is demonstrating zero-shot in-context learning for keyword boosting. The practical input to the model includes the speech signal, an optional text context, and the text instruction defining the task. For keyword boosting, the text context is used to provide a list of keywords or phrases of interest (e.g., "The following words may occur in audio: [nvidia, gpu, cpu,...]"). The model learns to prioritize these keywords during transcription solely based on their presence in the prompt context, without requiring explicit model parameter updates, training for specific keywords, or external biasing graphs. [32]

Evaluation for standard ASR was performed using WER on LibriSpeech. The keyword boosting capability was specifically evaluated on an internal dataset, which is rich in technical

jargon. Metrics used for evaluating keyword boosting were Precision, Recall, and F1-score, calculated based on the correct transcription of the provided keywords. [32]

The results showed that the proposed zero-shot in-context learning method for keyword boosting significantly improved the recognition of specified keywords compared to the model without the context prompt. On the internal dataset, providing the keywords in the prompt improved the keyword F1-score from 0.38 to 0.56 using the in-context learning strategy. This performance demonstrated substantial gains compared to the baseline without boosting and achieved results competitive with traditional shallow-fusion methods, but without the need for external graphs or specialized training for the boosted words. [32]

3 Representing Jargon in Speech Datasets

This chapter addresses the first research question of this thesis by examining how industry-specific jargon can be defined and represented in speech datasets. This chapter first establishes a working definition of jargon in the context of speech recognition, then outlines the key requirements for creating jargon-heavy datasets, and details the methodologies used for data collection. Finally, it presents the three datasets that were created or adapted for this thesis to serve as a benchmark for evaluating ASR models and improvement methods for jargon recognition.

3.1 Defining Industry-Specific Jargon

Jargon refers to the specialized terminology used within a particular profession or group, which is often difficult for outsiders to understand [59]. It serves as a shorthand that allows for precise and efficient communication among experts within industries. However, this same efficiency can create a barrier for those not versed in the specific language of that domain.

In the context of ASR, the term jargon is often used interchangeably with related concepts such as rare words or out-of-vocabulary (OOV) words. While these terms overlap, for the purpose of this thesis, jargon is defined by a specific set of characteristics that distinguish it from general vocabulary. Jargon terms in this thesis are defined as nouns or abbreviations that exhibit low frequency in general language corpora, possess a specialized meaning within a particular domain, and carry high semantic importance. The accurate recognition of jargon is critical because these terms often hold significant informational value and are key to understanding user intent. Substituting a jargon term with a more common but phonetically similar word can lead to a significant loss of meaning, as illustrated by the "BOM" versus "bomb" example in the introduction.

Jargon can be found in various sources, including domain-specific literature like documents, manuals, or glossaries. Identifying and extracting these terms is a crucial step in creating jargon-heavy datasets. This can be achieved through several methods, such as manual annotation by domain experts, who can use their knowledge to identify specialized terms [42]. Alternatively, automated extraction using techniques like Term Frequency-Inverse Document Frequency (TF-IDF) can help identify terms that are frequent in a specialized corpus but rare in a general one [40]. The use of pre-existing domain-specific lexicons or glossaries is another effective method for compiling a list of relevant jargon [11].

3.2 Requirements for Jargon-Heavy Datasets

While general-purpose ASR evaluation datasets prioritize diversity in speakers, accents, and acoustic environments, the specific challenge of recognizing jargon requires a more targeted set of criteria. To guide the collection of suitable datasets for this thesis, several key requirements were established:

- **Authentic Speech:** The dataset should consist of real or highly realistic speech recordings that reflect the acoustic properties and speaking styles found in real-world professional environments.
- **High-Quality Transcriptions:** Accurate and detailed transcriptions are essential. They must correctly capture not only the jargon terms but also the surrounding words, as transcription errors can affect the evaluation of the model's overall performance.
- **High Jargon Density:** To ensure sufficient representation for evaluation, the dataset must have a high density of domain-specific jargon. This contrasts with general ASR datasets, where broad vocabulary coverage is more important. A higher concentration of jargon allows for a more focused analysis of the model's ability to handle specialized terms.
- **Labeled Jargon Terms:** The dataset should include annotations or labels that indicate the presence and location of jargon terms within the transcriptions. This labeling is crucial for evaluating model performance on these specific terms and enables targeted training, fine-tuning, or adaptation strategies.
- **Sufficient Data Volume:** The dataset must contain a sufficient volume of data to allow for effective model evaluation and to prevent overfitting if the data is used for model adaptation. A larger volume of data provides a more reliable measure of a model's generalization capabilities.

3.3 Dataset Collection Methodology

To assemble datasets that meet the defined requirements, multiple strategies were developed. This includes the use of existing resources and the creation of new, tailored datasets.

The first strategy involved searching for publicly available datasets that already satisfy most, if not all, of the requirements. For this process the systematic literature review conducted provided valuable insights and identified publications and specialized forums for datasets containing domain-specific speech. In some cases, these datasets might require adaptation, such as annotating jargon terms, to fully align with the needs of this thesis. This approach is cost-effective and time-efficient, and it provides access to datasets that have been validated in prior research. However, the availability of such datasets is often limited.

A second strategy was to create a custom dataset from scratch using authentic speech. This method involves collecting real-world speech recordings from industry-specific sources, such as professional lectures, meetings, training videos, or podcasts. The collected audio is then transcribed and annotated with jargon terms. While this approach offers complete control over the dataset's characteristics, ensuring it is perfectly tailored to the research questions, it is also highly resource-intensive and time-consuming.

The third strategy focused on the creation of synthetic datasets. This involves generating jargon-heavy speech data using high-quality text-to-speech systems. The process begins with compiling industry-specific texts rich in jargon, which are then converted into speech. This method allows for precise control over jargon density and representation, and the transcriptions are inherently perfect. Although synthetic data may not fully capture the acoustic authenticity of real-world recordings, research has shown that it can be highly effective for training and evaluating ASR systems, particularly in low-resource scenarios where real data is scarce [36, 43, 44].

3.4 Collected Datasets

Based on the defined requirements and collection methodologies, three datasets were selected or created for this thesis. The second strategy to create a custom dataset from scratch was not pursued due to resource constraints, but the other two strategies yielded valuable datasets for evaluating ASR performance on jargon-heavy speech. Table 3.1 provides an overview of the key characteristics of the collected datasets.

Dataset	Domain	Example	# Datapoints	Total Length
UNITED-SYN-MED	Medical	"**Capiclav** tab contains **amoxicillin** as its active ingredient."	15,000	25.98 hours
Earnings-21	Financial	"Second quarter adjusted **EBITDA** was $349 million."	14329	36.30 hours
FAA-Glossary	Aviation	"I need the latest **ATIS** information for our arrival."	2616	5.17 hours

Table 3.1: Resulting jargon datasets created using different strategies.

3.4.1 UNITED-SYN-MED

The UNITED-SYN-MED dataset is a large-scale, synthetic dataset developed for the domain of medical ASR. It was created to train ASR models to accurately recognize specialized medical vocabulary. The dataset's creation was a multi-step process. First, medical text data, including terms and descriptions with a primary focus on medication and drugs, was scraped from authoritative online sources such as ICD-10, MIMS, and FDA databases. This extracted text was then processed by a generative AI model to create realistic and contextually accurate medical sentences. Finally, these text sentences were converted into high-quality synthetic audio using a Text-to-Speech (TTS) model, creating files with both male and female voices. In terms of size, the dataset consists of 395,000 unique sentences, which were used to generate 790,000 individual audio files. This collection totals around 5,486 hours of labeled audio. The UNITED-SYN-MED dataset is publicly accessible and available on the Hugging Face platform. [11]

The dataset meets several of the key requirements for this thesis. By design, the transcriptions are perfect, as the audio was generated directly from the text, eliminating the possibility of human transcription errors. It also features an exceptionally high jargon density, as it was specifically created to train medical ASR models with a focus on medication terms and every sample includes at least one jargon term. Furthermore, with over 5,000 hours of audio, it offers an exceptionally large volume of data. However, it also has limitations. Being synthetic, it may lack the acoustic variety and background noise found in real-world clinical environments. Most importantly, the original dataset does not include explicit labels for jargon terms, which is a critical requirement for the evaluation planned in this thesis.

To make the dataset suitable for this research, two main adaptations were performed. First, to address the lack of jargon labels, a domain-specific glossary of medication names was compiled. Since the creators of the dataset documented the sources used for scraping medical terminologies, it was possible to extract a comprehensive list of medication names from these same sources. This list was then used to automatically annotate the transcriptions, marking the presence and location of these jargon terms. The second adaptation was to reduce the dataset to a more manageable size for evaluation. A subset of 15,000 samples was randomly selected from the full dataset, resulting in a total length of approximately 26 hours of audio.

UNITED-SYN-MED was chosen for this thesis because its high density of specialized medical terms provides an excellent testbed for evaluating jargon recognition. Its synthetic nature, while a limitation in terms of acoustic realism, ensures perfect transcriptions, which allows for a clean and focused analysis of the models' ability to recognize jargon without the confounding factor of transcription errors. The ability to adapt the dataset by programmatically labeling jargon terms made it a valuable and practical resource for this study.

3.4.2 Earnings-21

The Earnings-21 dataset is a public corpus designed to benchmark ASR systems in a practical, real-world scenario, and includes named entity recognition as an additional feature for evaluation. The dataset's domain is financial, consisting of 44 public earnings calls from 2020. These calls cover nine different corporate sectors, providing entity-dense speech and varied, real-world recording qualities. To create the dataset, the audio files were first transcribed by a human transcription service to get highly accurate verbatim transcripts, which include filler words and disfluencies. These transcripts were then richly annotated with punctuation, true-casing, and Named Entity Recognition (NER) labels. Its detailed NER labels are a key feature of the dataset. The NER tagging was performed automatically using NER models with a subsequent manual review to ensure accuracy. These NER tags include various entity types, such as organizations, locations, dates, percentages, and more. The dataset contains a total of about 36 hours of speech data. It is publicly available on GitHub. [42]

The Earnings-21 dataset aligns well with the requirements for this thesis. It consists of authentic speech recordings from real earnings calls, capturing the natural speech and acoustic environment of professional financial discussions. The transcriptions are of high quality, as they have been produced by human transcribers. Earnings calls are known for their high density of financial jargon and named entities, which makes the dataset an excellent source of domain-specific language. While the dataset does not explicitly label jargon, its rich NER annotations can be leveraged to identify many relevant terms. With 36 hours of speech, it also provides a sufficient volume of data for effective model evaluation.

To adapt the dataset for this thesis, the NER tags were used to systematically label jargon terms. Several NER categories were identified as being particularly relevant to financial jargon:

- **ABBREVIATION:** In a financial context, abbreviations often have specific meanings (e.g.,EBITDA or GAAP) that an ASR system must correctly recognize.
- **ORG:** Refers to organisations. These include the company giving the call, competitors, partners, and subsidiaries. These proper nouns are often unique, easily confusable, and essential to understanding the business context.
- **PRODUCT:** Mentions of specific services, software, or, in the case of healthcare-related calls, drug names, function as technical jargon.

For better manageability, the audio files were split into smaller segments of approximately 5 to 20 seconds. This was done by using the provided aligned transcripts to create segments at sentence boundaries, which facilitates easier processing and evaluation.

The Earnings-21 dataset was chosen for its authenticity and its focus on the financial domain, which is rich in specialized terminology. The availability of high-quality human transcriptions and detailed NER tags provided a solid foundation for creating a well-defined jargon evaluation set. Also the dataset is highly relevant and used in prior research on ASR, making it a valuable benchmark for this thesis [25].

3.4.3 FAA-Glossary

Unlike the other two datasets, the FAA-Glossary dataset was created from scratch for this thesis to create a jargon-heavy dataset in the aviation domain. The foundation of this dataset is the official Aviation Glossary provided by the Federal Aviation Administration (FAA), which contains a comprehensive list of terms and acronyms used in the aviation industry [60].

The creation process began by extracting specific terms and their definitions from the FAA glossary. For each term, three example sentences were generated using the GPT-4o LLM. To ensure the sentences were contextually relevant and realistic, the model was prompted to create sentences that a pilot or air traffic controller might use in real-world communications. The definitions from the glossary were also provided to the model to help it understand the meaning and usage of each term. These generated sentences were then converted into speech using the Higgs TTS system by Boson AI, which produces high-quality speech with a variety of speakers and dialects [61]. To ensure authenticity, a manual review of the generated audio files was conducted. For each term, one of the three audio files was sampled and reviewed for naturalness, clarity, and correct pronunciation of the jargon term. Any audio file that showed issues was re-generated until a satisfactory quality was achieved, and in such cases, the other two audio files for that term were also investigated and re-generated if necessary. The final dataset consists of 2,616 audio samples, with a total length of approximately 5 hours of speech data.

The FAA-Glossary dataset was designed to meet all the requirements for a jargon-heavy dataset. The use of a high-quality TTS system and a manual review process ensures a high degree of authenticity. By design, the transcriptions are perfect, as the audio was generated from the text. The dataset has a high jargon density, as each sentence is specifically constructed to include at least one aviation-specific term. The jargon terms are also inherently labeled, as the dataset was built around the terms from the FAA glossary. While smaller in size compared to the other two datasets, the 5 hours of audio data is sufficient for evaluating model performance on aviation-specific jargon.

This dataset was created to provide a clean, controlled, and highly specific testbed for evaluating ASR performance on aviation terminology. By building the dataset from the ground up, it was possible to ensure that all the requirements for this thesis were met, including perfect transcriptions and clearly labeled jargon terms. This makes the FAA-Glossary dataset a valuable resource for assessing how well ASR models can handle the specialized and safety-critical language of the aviation industry.

4 Benchmark for ASR Performance on Jargon-Heavy Speech

This chapter addresses the methodology of the second research question: Which evaluation metrics best capture the performance of ASR systems on jargon-heavy speech? To answer this, a systematic benchmark is established to evaluate state-of-the-art ASR systems on the industry-specific datasets introduced in Chapter 3. The chapter begins by defining the evaluation metrics used to measure performance, including both standard transcription accuracy and keyword-specific recognition. It then introduces the selected ASR models, detailing their architectures and reasons for inclusion. Finally, it outlines the evaluation methodology, explaining how the datasets were processed and how the performance of each model was systematically measured.

4.1 Metrics

Evaluating the performance of jargon detection systems requires robust and interpretable metrics. As introduced in Chapter 2.3, in ASR several metrics are commonly used to assess the accuracy and effectiveness of transcription systems. One of the most widely adopted metrics is WER, which provides a quantitative measure of transcription accuracy by calculating a kind of distance between the predicted and reference transcriptions. As WER is an industry standard for evaluating ASR systems, it is also used in this benchmark. It offers a clear and interpretable measure of overall transcription accuracy, allowing for straightforward comparisons between different systems and configurations.

However, a key limitation of WER is, that it treats all words equally. In jargon-heavy contexts, certain terms carry significantly more importance than others. For example, in a medical dictation, misrecognizing the word “a” as “the” has a negligible impact, whereas substituting one medication name for another, such as “aspirin” for “ibuprofen”, constitutes a critical failure. While both are single-word substitution errors from a WER perspective, their consequences are vastly different. Semantic metrics like BERTScore attempt to address this by measuring meaning, but they can also be insufficient. “Aspirin” and “ibuprofen” are semantically similar as both are pain relievers, yet confusing them is unacceptable.

That is why this benchmark incorporates keyword-focused metrics to specifically evaluate the recognition of industry-specific terms. The metrics used in this benchmark are Precision, Recall, and F1-score, as defined in Chapter 2.3. These metrics specifically assess the system’s

ability to correctly identify and transcribe jargon terms from a predefined glossary. Precision measures the accuracy of the identified jargon terms, Recall evaluates the system's ability to capture all relevant jargon terms, and F1-score provides a balanced measure that considers both Precision and Recall.

4.2 State-of-the-Art ASR Systems

To establish a robust benchmark, three distinct state-of-the-art ASR systems were selected. Each represents a different architectural philosophy and holds a prominent position in the current ASR landscape, providing a comprehensive basis for evaluating performance on jargon-heavy speech.

OpenAI Whisper

OpenAI's Whisper is a widely adopted open-source ASR model based on an encoder-decoder transformer architecture. Trained on a massive dataset of supervised multilingual audio, its latest version, whisper-large-v3, has become a standard for high-quality transcription [27].

Whisper was chosen for this benchmark for several key reasons. As one of the most popular and accessible high-performance ASR models, it serves as an essential baseline. Its well-documented architecture and strong performance make it an ideal reference point against which to measure other models and the proposed improvement techniques. Furthermore, its open-source nature makes it a practical foundation for adaptation, as demonstrated by one of the methods explored in this thesis.

NVIDIA Canary-Qwen-2.5b

NVIDIA's Canary-Qwen represents a more recent architectural paradigm known as a SALM. It features a hybrid architecture that combines a specialized speech encoder with a powerful pre-trained text LLM as the decoder [31]. As of August 2025, it is the top-performing English ASR model on the Hugging Face Open ASR Leaderboard [25].

The inclusion of Canary-Qwen-2.5b is motivated by its state-of-the-art performance and its novel architecture. By integrating a dedicated LLM into the transcription process, it represents a different approach to language modeling compared to Whisper's more traditional decoder. This makes it a critical point of comparison to assess how different architectural philosophies handle the challenge of industry-specific jargon.

Google Gemini

Google's Gemini models are designed from the ground up as natively multimodal systems capable of processing and reasoning over text, audio, image, and video inputs. They are built on a sparse MoE transformer architecture, which allows them to scale to billions of parameters while maintaining computational efficiency [34].

Gemini was selected because it represents the cutting edge of multimodal AI. Unlike Whisper or Canary-Qwen, its audio processing capabilities are part of a broader reasoning framework. The 2.5-Flash version was specifically chosen for this benchmark as it is optimized for a balance of performance, cost, and speed, aligning with the thesis's focus on identifying practical, real-world solutions. Its inclusion allows for an evaluation of how a truly multimodal, large-scale model performs on specialized ASR tasks and provides a powerful platform for exploring in-context learning and post-correction methods.

4.3 Data Preparation

The datasets were stored on a remote server with GPU resources to facilitate efficient processing. For each dataset, a CSV file was created to organize the data. Each row in the CSV file corresponds to a single audio segment and contains the path to the audio file, the corresponding reference transcript, and an array of the jargon terms present in that audio file. For the Earnings-21 dataset, which consists of earnings calls from different companies and industries, each call was handled individually, resulting in separate CSV files. This was necessary because the jargon terms are specific to each company's domain. This structured approach allowed for the datasets to be easily iterated over and processed by each ASR system in an automated fashion.

4.4 Implementation of ASR Inference

The inference process, or the method of generating transcriptions, varied for each of the three models due to their different architectures and deployment methods.

Whisper-large-v3

For Whisper, the Hugging Face Transformers library was used. The Hugging Face Transformers library is an open source Python library created to make state-of-the-art advances in NLP accessible to the wider machine learning community. It can easily be installed on a Python environment with pip. The library's primary functions are to provide a collection of carefully engineered, state-of-the-art Transformer architectures accessible through a unified API and to facilitate the distribution of pretrained models. A central feature is the Model Hub, where the community can access, use, and share a wide variety of pretrained and fine tuned models. The library is designed to be simple for practitioners while also being extensible for

researchers and robust enough for industrial deployments. [62] The library offers a simple powerful Pipeline API that abstracts away the complexities of model loading and tokenization, allowing users to perform tasks such as text summarization, image classification, and speech recognition with just a few lines of code [63].

Whisper-large-v3 is also available via the Hugging Face Model Hub, which makes it easy to load and use the model for inference [28]. On a remote GPU server, a Python script was deployed that reads the CSV files for each dataset, loads the audio files, and transcribes them with the Whisper-large-v3 model by utilizing the Pipeline API of Hugging Face Transformers library. The generated transcripts were then saved as new column in the respective CSV files for further analysis.

Canary-Qwen-2.5b

NVIDIA published its Canary-Qwen-2.5b model on the Hugging Face Model Hub as well. However in contrast to Whisper, it is not available through the Pipeline API [31]. NVIDIA provides its own framework called NeMo. NeMo is a Python toolkit for building AI applications. Its main idea is to use Neural Modules, which are conceptual building blocks of a neural network, like an encoder, decoder, or loss function. NeMo makes it easy to combine and re-use these blocks. It provides pre-built modules for common tasks in NLP like ASR. [64]

In practice the approach to transcribe with Canary-Qwen-2.5b is similar to Whisper. The NeMo framework can be installed on a local Python environment with pip. In a Python script the NeMo library was imported, which allowed loading the SALM model Canary-Qwen-2.5b. The script then iterates over the CSV files of each dataset, loads the audio files, and transcribes them with the model. For Canary-Qwen-2.5b it's important to note that the model does not only require audio input but also needs to be instructed by the text prompt "Transcribe the following:" to perform the transcription task. The results were again saved as new column in the respective CSV files.

Gemini-2.5-Flash

Google Gemini-2.5-Flash is not available as open source model. Instead, it can be accessed via Google's Gemini Developer API [65]. The Google Gemini Developer API allows developers to integrate Google's advanced generative AI models into their applications. It provides access to the Gemini family of models, such as Gemini-2.5-Pro for complex reasoning and Gemini-2.5-Flash for high-speed, cost-efficient tasks. Like the models, the API supports multimodal capabilities, enabling processing of text, images, audio, and video inputs. To facilitate integration, Google provides the GenAI Software Development Kit (SDK) for various programming languages, including Python. To use the API, developers need to set up an API key in Google AI Studio. After installing the GenAI SDK in a Python environment with pip, the API key can be used to authenticate requests. [65]

To transcribe audio files with Gemini-2.5-Flash, the SDK and API key can be used to send a request to the model. Similar to Canary-Qwen-2.5b, Gemini-2.5-Flash requires the input audio and a text prompt to specify the task. In this case, the prompt "Generate a transcript of the speech." is used. To increase efficiency and reduce costs, batch processing was implemented for the process of transcribing the datasets. In that way, multiple audio files can be sent in a single API call and transcribed simultaneously in the background. Once the transcription was completed, the results were fetched and saved in the respective CSV files.

4.5 Evaluation Methodology

After all datasets were transcribed by the three models, the generated transcripts were compared to the reference transcripts in the CSV files to calculate the evaluation metrics introduced earlier.

To compute the WER, the open-source Python library JiWER was used. JiWER is a simple and efficient library for evaluating ASR systems [66]. First the reference and predicted transcripts were normalized. JiWER provides several built-in normalization functions. For this purpose the following normalizations were applied in sequence:

- Converting to lowercase (e.g., "Hello" to "hello"),
- expanding common english contractions (e.g., "don't" to "do not"),
- removing kaldi tags (e.g., "[noise]"),
- removing leading and trailing white spaces (e.g., " hello " to "hello"),
- removing multiple white spaces (e.g., "hello world" to "hello world"),
- and removing punctuation (e.g., "hello!" to "hello").

Removing punctuation is not always standard practice when calculating WER, but it was deemed necessary here to ensure that the evaluation focuses on word recognition rather than punctuation accuracy, which is less relevant in spoken language contexts.

After normalization JiWER provides a straightforward way to calculate WER. The library correctly handles the alignment of reference and hypothesis sentences and computes the number of substitutions, deletions, and insertions required to match them. JiWER also handles normalization steps, which are necessary for fair comparisons of reference and predicted transcripts. [66] For the keyword-based F1-score, a custom Python function was implemented. First, a comprehensive list of all unique jargon terms appearing in the reference transcripts of each dataset was created to serve as the baseline vocabulary for evaluation. This list of terms was also normalized using the same normalization steps as described above for WER calculation to ensure consistency in matching. The function then iterates through each audio file's data, comparing the generated transcript with the list of jargon terms associated with that audio file and the complete jargon list for the dataset. The matching logic performed

a simple substring search within the transcripts to identify the presence or absence of each jargon term, which could be a single word or a multi-word phrase. Based on this True Positives, False Positives, and False Negatives were counted as follows:

- A True Positive is counted if a jargon term is present in both the reference and the ASR transcript.
- A False Positive is counted if a word or phrase from the dataset's complete jargon list appears in the ASR transcript but is not present in the specific reference transcript for that audio clip. This method correctly penalizes the model for hallucinating valid jargon terms that were not actually spoken.
- A False Negative is counted if a jargon term is present in the reference transcript but is missed by the ASR system.

TNs were not counted, as they are not relevant for the calculation of precision, recall, and the F1-score. Furthermore, defining a True Negative in this context is challenging, as it would represent any non-jargon word correctly transcribed as such, which is an overwhelmingly large and uninformative set. From the aggregated TP, FP, and FN counts, precision, recall, and the F1-score were calculated for each model on each dataset.

5 Improving ASR for Jargon Recognition

This chapter addresses the third research question of this thesis: What methods can improve ASR performance in recognizing industry-specific jargon with no availability of in-domain audio data? Building upon the systematic literature review presented in Chapter 2 and the baseline performance methodology established in Chapter 4, this chapter presents the experimental investigation of three distinct methods for enhancing jargon recognition in ASR systems. The chapter details the methodology and implementation of each approach, providing the foundation for the quantitative results that will be presented in Chapter 6. The focus is on practical solutions that can be applied without requiring in-domain audio datasets, making these methods particularly valuable for organizations operating in specialized domains where such data is scarce or unavailable.

5.1 Selection of Improvement Methods

Based on the systematic literature review conducted in Chapter 2, three methods were selected for implementation and evaluation in this study. These methods were chosen for their demonstrated potential effectiveness in low-resource scenarios, where extensive in-domain audio data is not available, and for their practical applicability in real-world settings.

The three methods investigated are:

- **Keyword-guided adaptation using a specialized keyword spotting model:** This approach was selected because it provides direct acoustic guidance during the transcription process, allowing the ASR system to be more sensitive to specific jargon terms without requiring model retraining.
- **Post-processing step using a LLM for error correction:** This method was chosen for its ability to leverage the extensive linguistic knowledge of modern LLMs to identify and correct domain-specific transcription errors after the initial ASR processing.
- **Zero-shot in-context learning approach that leverages multimodal LLM capabilities:** This approach was selected because it exploits the reasoning capabilities of multimodal models to understand both audio input and contextual information about expected jargon terms simultaneously.

Each method represents a different strategy for providing domain-specific context to the ASR process. Crucially, all three approaches operate with zero requirements for in-domain audio

data from the target domain. Instead, each method works by incorporating a list of domain-specific jargon terms to provide contextual guidance during or after the transcription process. This characteristic makes these methods particularly suitable for low-resource scenarios where in-domain audio data is unavailable, addressing a key practical barrier to ASR adoption in specialized domains. This chapter describes the detailed implementation of each method, establishing the experimental framework for evaluating their effectiveness in the subsequent results chapter.

5.2 Keyword Selection Strategy

As each method requires a list of domain-specific jargon terms to provide contextual guidance, a consistent keyword selection strategy was developed to identify the most relevant terms for each dataset. The keyword selection was based on the baseline performance analysis presented in chapter 4 (detailed results in chapter 6). For each dataset, the most frequently missed jargon terms were selected based on the baseline transcription results. Specifically, these were the keywords that exhibited the highest false negative rates across all baseline ASR models tested in Chapter 4. This selection strategy aims to directly address the most significant weaknesses of existing ASR systems by focusing computational resources and contextual guidance on the jargon terms that proved most challenging for state-of-the-art models.

The false negative analysis was conducted by comparing the transcriptions of the baseline models against the ground truth references, identifying instances where jargon terms present in the reference were missed or incorrectly transcribed by the ASR systems. The terms were then ranked by their frequency of occurrence as false negatives. Depending on the dataset the number of keywords in this list varied.

5.3 Evaluation Methodology

The same evaluation metrics established in Chapter 4 were employed to assess the effectiveness of each improvement method. These include WER for overall transcription accuracy, and Precision, Recall, and F1-score for jargon-specific performance evaluation. Additionally the same normalization steps were applied to ensure consistency in the evaluation process. This consistency ensures direct comparability between the baseline performance and the results achieved by each improvement method.

For the methods that utilize LLM services, additional cost tracking was implemented by monitoring token consumption during the experiments. This analysis provides important practical insights into the trade-offs between performance improvements and operational costs, which is crucial for real-world deployment considerations.

5.4 Keyword-Guided Adaptation with Keyword Spotting

Keyword-guided adaptation with keyword spotting represents a direct approach to improving ASR performance on domain-specific terminology by providing explicit guidance to the model during the transcription process. As discussed in Chapter 2, this method works by making the ASR system more sensitive to predefined keywords, effectively biasing the recognition process toward expected jargon terms when acoustically similar alternatives are encountered. Figure 5.1 illustrates the simplified architecture of the ASR system with integrated keyword spotting capabilities.

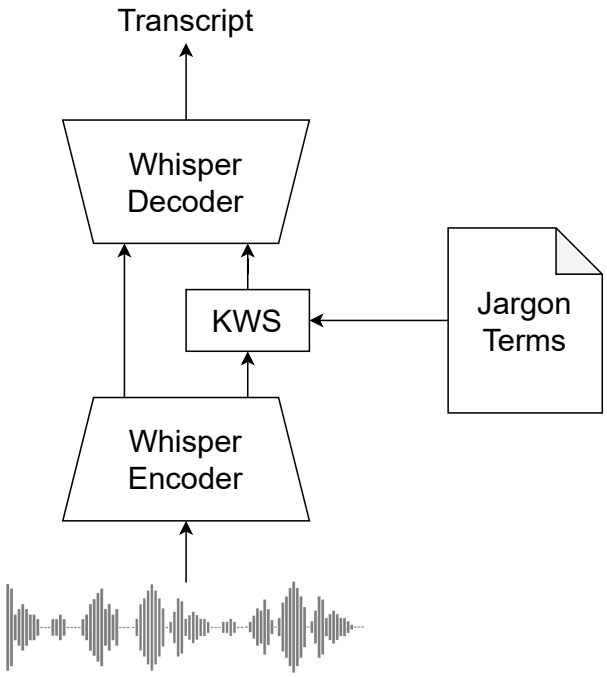

Figure 5.1: Schematic architecture of Whisper with integrated keyword spotting capabilities.

Rather than implementing a keyword spotting system from scratch, this study leveraged the commercial ASR service provided by aiOla, a company specializing in domain-specific speech recognition. aiOla offers an API access to their Jargonic model, which is specifically designed for handling industry-specific jargon and provides built-in keyword spotting capabilities. This approach was chosen for several practical reasons: it provides access to a production-grade system optimized for jargon recognition, it eliminates the substantial development time required to build and train a keyword spotting model, and it offers a realistic evaluation of commercially available solutions.

For evaluation purposes, aiOla provided access to their service through a free trial period. The integration process involved creating an API key and accessing the models through aiOla's Python SDK. According to the aiOla documentation, their API allows users to provide a list of keywords to the model for keyword spotting during transcription. These keywords are used to boost the recognition confidence of specific terms when they are acoustically detected in the input audio.

The aiOla documentation recommends providing between 10 and 50 keywords for optimal performance, balancing recognition improvement with computational efficiency. Since some of the datasets used in this thesis contain significantly more than 50 unique jargon terms, the keyword selection strategy detailed before was applied. For each dataset, the top 50 most frequently missed terms from the baseline transcriptions were selected for keyword spotting.

This selection strategy directly targets the most problematic terms identified in the baseline evaluation, focusing the keyword spotting capabilities on the jargon terms that proved most challenging for state-of-the-art ASR systems. The selected keywords were provided to the aiOla ASR model via the API during the transcription process.

The aiOla API also supports providing phonetic or spoken forms for each keyword to further enhance recognition accuracy. However, since the datasets used in this study do not contain phonetic transcriptions or alternative pronunciations of the jargon terms, the keywords were provided in their written form only. This limitation represents a realistic constraint that many organizations would face when implementing such a system.

The experimental implementation followed a similar structure to the benchmarking process described in Chapter 4. A Python script was developed to iterate through each of the jargon-heavy datasets systematically. To isolate the impact of keyword spotting, two parallel experiments were conducted for each dataset: one without keyword spotting (serving as the aiOla baseline) and one with keyword spotting using the selected top 50 jargon terms.

This experimental design enables a direct comparison of the aiOla system's performance with and without keyword guidance, providing clear insights into the effectiveness of this approach. Using aiOla's SDK, the audio data from each dataset was submitted to the API for transcription under both conditions. The generated transcripts were systematically saved to CSV files, maintaining the same data organization structure used in the baseline evaluation for consistent analysis and comparison.

5.5 LLM-Based Error Correction

The LLM-based error correction approach represents a post-processing strategy that leverages the extensive linguistic knowledge and reasoning capabilities of modern LLMs to identify and correct domain-specific transcription errors. As detailed in Chapter 2, this method treats the initial ASR output as a draft transcript that can be refined through intelligent post-processing,

taking advantage of the LLM's understanding of context, domain-specific terminology, and linguistic patterns. The approach is schematically illustrated in Figure 5.2.

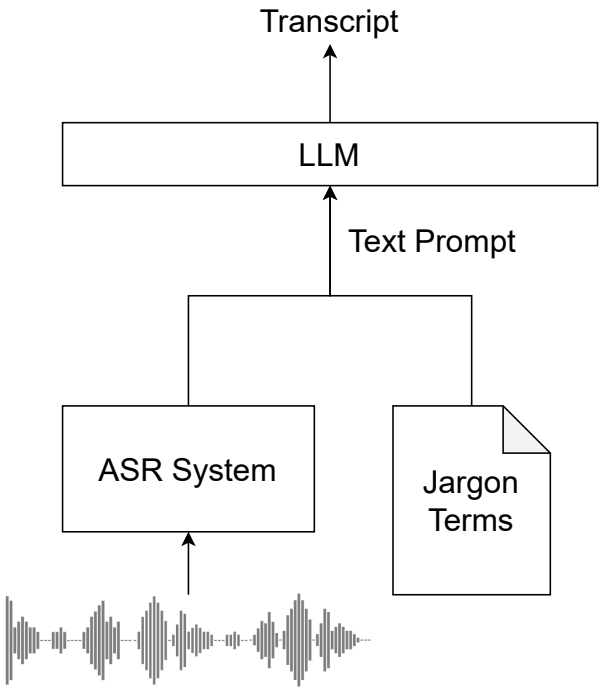

Figure 5.2: Schematic of the LLM-based error correction approach.

The implementation of this approach required several key adaptations from the methods described in the related work literature. Most significantly, Google's Gemini-2.5-Flash was selected as the LLM error correction module, representing one of the most advanced multimodal language models available at the time of this research. This choice was informed by findings in the literature that demonstrated superior performance when using more advanced LLM architectures, similar to how GPT-4 outperformed GPT-3.5 in the study [58].

Following the approach identified in the systematic literature review, few-shot prompting combined with a list of domain-specific terms was implemented to achieve optimal results. However, since the exact prompt formulations were not published in the referenced literature, a custom prompt had to be designed based on the methodological descriptions available in the papers.

The prompt design focused on creating clear instructions that would guide the LLM to focus specifically on domain-specific terminology while preserving the overall structure and content of the original transcript. The final prompt template is presented below:

```
You are acting as a highly specialized proofreader for transcripts from
automatic speech recognition (ASR). Your focus is on the precise cor-
rection of domain-specific jargon, terms and abbreviations. Goal is to
ensure that the final text accurately reflects the intended terminology
used in the original speech.

You will be provided with a raw ASR transcript. This transcript may or
may not contain errors or inaccuracies, particularly with regard to spe-
cialized terminology. The underlying audio is from the [DOMAIN] domain.
Your task is to correct potential errors in the provided ASR transcript
based on the provided list of jargon terms. Only replace a word if it
fits semantically into the context of the sentence. If the replacement
makes the sentence illogical, keep the original transcription.

[LIST_OF_JARGON_TERMS]
```

The industry-specific terminology list used the same keyword selection strategy described before, employing the top 50 most frequently missed jargon terms from the baseline transcriptions. This list was dynamically inserted into the prompt template for each dataset, with the domain specification adjusted accordingly. The limit of 50 keywords was chosen to balance the need for comprehensive jargon coverage with the practical constraints of prompt length and token consumption, which are important considerations when using commercial LLM services. This limit also maintains comparability with the keyword spotting approach, which also utilizes 50 keywords.

A critical implementation decision involved the selection of baseline transcriptions for error correction. The outputs from Gemini-2.5-Flash generated during the benchmarking phase were used as the initial transcripts for correction. This choice was made for two reasons: Gemini-2.5-Flash demonstrated the lowest average WER across the datasets in the baseline evaluation, providing the highest quality starting point for correction, and using a consistent baseline across all improvement methods enables fair comparison of their relative effectiveness.

The technical implementation used Google's GenAI SDK for Python to interface with the Gemini API. A processing script was developed to systematically iterate through each dataset, where for each audio file and its corresponding Gemini baseline transcription, the prompt template was populated with the appropriate jargon list and the transcript text. This complete prompt was then submitted to the Gemini-2.5-Flash model via the API to generate the corrected transcript.

To monitor the operational costs associated with this approach, token consumption was carefully tracked during the experiments. A representative subset of each dataset was

initially processed to estimate the total token usage and associated costs, providing important practical insights into the economic feasibility of this method for large-scale deployment. The corrected transcripts were systematically saved to CSV files using the same organizational structure as the baseline evaluation, ensuring consistent formatting for subsequent analysis and comparison.

5.6 Keyword-Boosting via Zero-Shot In-Context Learning

The zero-shot in-context learning approach is the last strategy explored in this thesis. Rather than using the LLM as a post-processing correction module, this method leverages the multimodal capabilities of modern language models to handle the entire ASR process while incorporating domain-specific context directly into the transcription task. As discussed in Chapter 2, this approach exploits the in-context learning abilities of LLMs, where the model can adapt its behavior based on instructions and examples provided within the input prompt without requiring parameter updates or fine-tuning. The method is schematically illustrated in Figure 5.3.

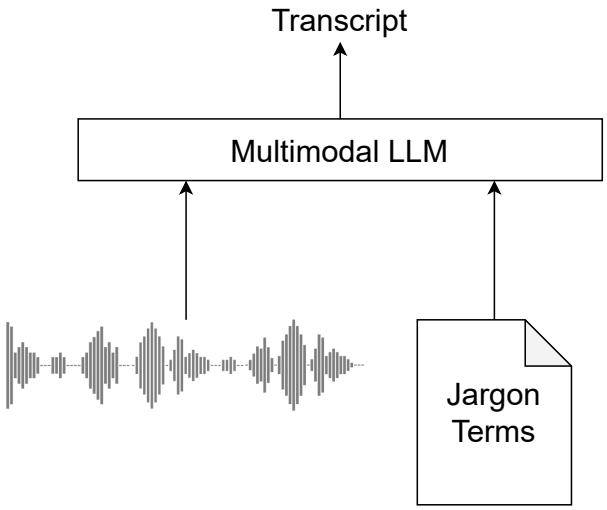

Figure 5.3: Schematic of the zero-shot in-context learning approach.

The implementation builds upon the baseline transcription methodology established in Chapter 4, where Gemini-2.5-Flash was used for direct audio transcription. However, instead of providing a simple transcription instruction, the prompt is enhanced with explicit guidance about domain-specific jargon terms that the model should prioritize during the transcription process.

The core innovation of this approach lies in its simplicity and directness. By providing the multimodal LLM with both the audio input and a contextual list of expected jargon terms, the model can leverage its reasoning capabilities to make more informed transcription decisions when encountering acoustically ambiguous segments. This method effectively combines the acoustic processing and linguistic reasoning capabilities of the model in a single, integrated step.

The prompt design focused on creating clear, domain-specific instructions that would guide the model's attention toward critical terminology while maintaining natural transcription behavior for general vocabulary. The following template illustrates the approach:

```
You are an expert transcriptionist in the [DOMAIN] domain. Your task is
to accurately transcribe the provided audio file. Pay extremely close at-
tention to the following list of [DOMAIN] terms. These words are critical
and are often misinterpreted by standard transcription services. When
you hear sounds in the audio that are phonetically similar to any term on
this list, you must use the exact spelling provided below.

[LIST_OF_JARGON_TERMS]
```

To explore this trade-off systematically, two different keyword list sizes were investigated: 50 and 1,000 of the most frequently missed jargon terms from the baseline transcription. For the experiment utilizing 1,000 keywords, only a representative subset of 1,500 samples (10% of the UNITED-SYN-MED dataset) was processed to limit cost and computational time expenditure. For the subset UNITED-SYN-MED-1.5k 1,500 samples were randomly selected from the full UNITED-SYN-MED dataset. This sampling approach was not necessary for the FAA-Glossary and Earnings-21 datasets due to their smaller jargon vocabularies and more manageable dataset sizes.

The technical implementation used the same Google GenAI SDK for Python that was employed in the error correction approach. A processing script was developed to systematically iterate through each dataset, where for each audio file, the prompt template was populated with the appropriate domain-specific jargon list and submitted to the Gemini-2.5-Flash model via the API for direct audio transcription.

The generated transcripts were systematically saved to CSV files maintaining consistency with the data organization used throughout the study. Similar to the error correction method, comprehensive token consumption tracking was implemented to assess the economic implications of this approach. Representative subsets of each dataset were initially processed to estimate total token usage and associated costs, providing crucial insights into the practical feasibility of deploying such methods at scale in real-world applications.

6 Results

This chapter presents the quantitative results of the experiments conducted in this thesis. It begins by establishing a performance baseline through a benchmark of the three selected state-of-the-art ASR systems on the jargon-heavy datasets. Subsequently, it details the outcomes of the three improvement methods investigated: keyword-guided adaptation, LLM-based error correction, and in-context learning for keyword boosting. The performance of each method is evaluated using the metrics defined in Chapter 4, namely WER, Precision, Recall, and F1-score. Finally, a cost analysis provides insights into the practical implications of implementing these methods in real-world scenarios.

6.1 Baseline

To establish a performance baseline, an initial benchmark was conducted. This benchmark evaluates the capabilities of three prominent, state-of-the-art ASR models: Whisper-large-v3, Canary-Qwen-2.5b, and Gemini-2.5-Flash. It is crucial to note that these models were used in their out-of-the-box state, meaning no fine-tuning or adaptation was performed on them prior to the evaluation. The benchmark was executed on the three specialized, jargon-heavy datasets, following the methodology outlined in Chapter 4. The comprehensive results, presented in Table 6.1, provide a baseline against which the subsequent improvement techniques are compared.

On the FAA-Glossary dataset, all models performed exceptionally well. Whisper-large-v3 emerged as the top performer in most categories, delivering the lowest WER at 1.2%, the highest F1-score at 94.9%, and the best Recall at 93.2%. Gemini-2.5-Flash achieved the highest Precision (99.2%). Notably, the performance differences across all models were minimal. The WERs were tightly clustered (1.2%, 1.4%, 1.6%) as were the F1-scores (94.9%, 94.0%, 94.7%). This suggests that the FAA-Glossary dataset, despite its technical vocabulary, is relatively easy for these state-of-the-art ASR systems.

The UNITED-SYN-MED dataset revealed more distinct performance profiles. Gemini-2.5-Flash was the clear leader, achieving the lowest WER (7.8%), which was notably better than Whisper's 10.2% and Canary's 13.0%. Gemini also led in keyword identification with the highest F1-score (62.2%) and Recall (45.2%). A critical observation on this dataset is the trade-off between Precision and Recall. While all models achieved outstanding Precision (Whisper: 99.6%, Canary: 99.3%, Gemini: 99.3%), both Whisper-large-v3 and Canary-Qwen-2.5b exhibited relatively low Recall of 35.1% and 27.9%, respectively. This poor recall directly

Model	Metric	FAA-Glossary	UNITED-SYN-MED	Earnings-21
Whisper-large-v3	**WER**	**1.2%**	10.2%	12.7%
	F1	**94.9%**	51.9%	**74.1%**
	Precision	96.5%	**99.6%**	73.8%
	Recall	**93.2%**	35.1%	**74.4%**
Canary-Qwen-2.5b	**WER**	1.4%	13.0%	14.1%
	F1	94.0%	43.5%	66.8%
	Precision	96.4%	99.3%	**74.4%**
	Recall	91.6%	27.9%	60.5%
Gemini-2.5-Flash	**WER**	1.6%	**7.8%**	**11.3%**
	F1	94.7%	**62.2%**	67.6%
	Precision	**99.2%**	99.3%	72.9%
	Recall	90.6%	**45.2%**	63.0%

Table 6.1: Benchmark results with state-of-the-art ASR systems on jargon-heavy speech datasets.

resulted in their lower F1-scores (51.9% and 43.5%).

For the Earnings-21 dataset, the results were mixed. Gemini-2.5-Flash again recorded the lowest overall WER at 11.3%, indicating the best general transcription accuracy. However, when evaluating the jargon-specific metrics, Whisper-large-v3 showed superior performance, achieving the highest F1-score (74.1%) and the highest Recall (74.4%). Precision scores were very similar across all three models, with Canary-Qwen-2.5b (74.4%) and Whisper-large-v3 (73.8%) holding a slight edge over Gemini-2.5-Flash (72.9%).

In summary, all models demonstrate reasonable baseline capabilities on these specialized datasets with similar performance through all datasets. The FAA-Glossary dataset appears to be the easiest, with all models achieving low error rates and high F1-scores. The Earnings-21 dataset presented the highest average WER, suggesting greater difficulty in general transcription. The UNITED-SYN-MED dataset proved particularly challenging for keyword spotting, as evidenced by the lowest average F1-scores, largely driven by significant issues with recall.

These baseline results establish the performance benchmarks against which the improvement methods will be evaluated.

6.2 Keyword-Guided Adaptation with Keyword Spotting

The keyword-guided adaptation approach was evaluated using aiOla's Jargonic ASR model with its integrated keyword spotting capabilities. To establish a fair comparison, the aiOla system was first tested without keyword spotting to determine its baseline performance, followed by evaluation with keyword spotting enabled using the top 50 most frequently missed jargon terms from the baseline results.

Table 6.2 presents the baseline performance of the aiOla Jargonic model without keyword spotting activated. These results serve as the reference point for measuring the impact of keyword spotting.

Metric	FAA-Glossary	UNITED-SYN-MED	Earnings-21
WER	2.1%	11.7%	15.1%
F1	93.7%	49.5%	65.0%
Precision	96.7%	99.6%	73.1%
Recall	90.9%	33.0%	58.6%

Table 6.2: aiOla Jargonic baseline results without keyword spotting.

The baseline aiOla results show performance characteristics similar to the other ASR systems evaluated in the benchmark. The FAA-Glossary dataset yielded the best overall performance with the lowest WER (2.1%) and highest F1-score (93.7%). The UNITED-SYN-MED dataset proved most challenging, exhibiting the poorest F1-score (49.5%) due to low recall (33.0%), despite maintaining high precision (99.6%). The Earnings-21 dataset showed intermediate performance across all metrics and again the highest WER (15.1%).

Table 6.3 presents the results when keyword spotting was activated, with changes calculated relative to the aiOla baseline in percentage points. The most notable improvements occurred in the Recall metric across all datasets, with the largest improvement on the Earnings-21 dataset (8.2 percentage points), followed by FAA-Glossary (1.4 percentage points) and UNITED-SYN-MED (0.5 percentage points). Notably the precision metric decreased in all datasets, indicating a trade-off between precision and recall when keyword spotting is enabled. The most significant precision drop occurred in the Earnings-21 dataset (9.5 percentage points).

Metric	FAA-Glossary	UNITED-SYN-MED	Earnings-21
WER	2.0% (-0.1)	11.6% (-0.1)	15.1% (±0)
F1	94.2% (+0.5)	50.1% (+0.6)	65.2% (+0.2)
Precision	96.2% (-0.5)	99.1% (-0.5)	63.6% (-9.5)
Recall	92.3% (+1.4)	33.5% (+0.5)	66.8% (+8.2)

Table 6.3: aiOla Jargonic with keyword spotting compared to baseline aiOla results in percentage points.

However, when comparing the aiOla keyword spotting results to the baseline models from the benchmark, the keyword-guided adaptation approach generally underperformed. For the FAA-Glossary dataset, aiOla with keyword spotting achieved a WER of 2.0% compared to Gemini's baseline of 1.6%, and an F1-score of 94.2% compared to Whisper's 94.7%. Similarly, on the UNITED-SYN-MED dataset, aiOla's F1-score of 50.1% was lower than Gemini's baseline of 62.2%. The Earnings-21 dataset showed the largest performance gap, with aiOla's F1-score of 65.2% significantly trailing Gemini's baseline of 67.6%.

6.3 LLM-Based Error Correction

The LLM-based error correction approach employed Gemini-2.5-Flash to post-process initial transcriptions and correct domain-specific errors. This method used the initial Gemini-2.5-Flash transcriptions as input and applied a specialized prompting strategy that included the top 50 most frequently missed jargon terms for each dataset to guide the error correction process.

Table 6.4 presents the results of the LLM-based error correction method, with changes calculated relative to the baseline Gemini-2.5-Flash transcriptions in percentage points. The results reveal a mixed performance pattern across the three datasets, with notable improvements in some metrics counterbalanced by degradation in others.

Metric	FAA-Glossary	UNITED-SYN-MED	Earnings-21
WER	1.7% (+0.1)	8.7% (+0.9)	12.4% (+1.1)
F1	90.5% (-4.2)	51.9% (-10.2)	66.8% (-0.8)
Precision	90.1% (-9.1)	70.6% (-28.7)	54.7% (-18.2)
Recall	90.9% (+0.3)	41.0% (-4.2)	85.8% (+22.8)

Table 6.4: LLM-based error correction results compared to Gemini-2.5-Flash baseline in percentage points.

The post-correction approach resulted in worse overall transcription quality, as evidenced by increased WER across all datasets. The FAA-Glossary dataset experienced a 0.1 percentage point increase in WER, while the UNITED-SYN-MED and Earnings-21 datasets showed more substantial degradations of 0.9 and 1.1 percentage points, respectively.

The precision metric suffered significant decreases across all datasets, with the most severe impact on the UNITED-SYN-MED dataset (-28.7 percentage points), followed by the Earnings-21 dataset (-18.2 percentage points), and a smaller but still notable decline in the FAA-Glossary dataset (-9.1 percentage points). However, the method showed the most significant improvement in recall for the Earnings-21 dataset, achieving a 22.8 percentage point increase. The FAA-Glossary dataset showed minimal recall improvements of 0.3 percentage points. In contrast, the UNITED-SYN-MED dataset's recall decreased by 4.2 percentage points.

The F1-scores, which balance precision and recall, decreased across all datasets, with the UNITED-SYN-MED dataset experiencing the largest decline (-10.2 percentage points). Only the Earnings-21 dataset maintained relatively stable F1 performance with a minor decrease of 0.8 percentage points.

6.4 Keyword-Boosting via Zero-Shot In-Context Learning

The zero-shot in-context learning approach leveraged Gemini-2.5-Flash's multimodal capabilities to simultaneously process audio input and textual context containing domain-specific jargon terms. This method involved providing the ASR model with a list of relevant jargon terms directly in the prompt, enabling the model to use this contextual information during the transcription process.

Two variants of this approach were evaluated: one using 50 jargon terms per dataset, and another using up to 1,000 terms specifically for the UNITED-SYN-MED dataset to assess the impact of increased contextual information.

Table 6.5 presents the results for the in-context learning approach using a maximum of 50 jargon terms, with changes calculated relative to the baseline Gemini-2.5-Flash transcriptions in percentage points.

Metric	FAA-Glossary	UNITED-SYN-MED	Earnings-21
WER	0.8% (-0.8)	7.3% (-0.5)	10.2% (-1.1)
F1	96.8% (+2.1)	63.4% (+1.3)	76.8% (+9.2)
Precision	95.6% (-3.6)	95.4% (-3.9)	65.8% (-7.1)
Recall	98.1% (+7.5)	47.5% (+2.3)	92.2% (+29.2)

Table 6.5: Keyword-boosting via zero-shot in-context learning with up to 50 jargon terms compared to Gemini-2.5-Flash baseline in percentage points.

The most significant improvement was observed on the Earnings-21 dataset, where the approach yielded substantial improvements in recall (29.2 percentage points) and F1-score (9.2 percentage points), with a 1.1 percentage point reduction in WER.

The FAA-Glossary dataset again achieved the best absolute performance, with the WER decreasing by 0.8 percentage points from 1.6% to 0.8%, accompanied by a 7.5 percentage point increase in recall and a 2.1 percentage point improvement in F1-score, resulting in the highest F1-score (96.8%) and lowest WER (0.8%) achieved across all methods and datasets in this study.

The UNITED-SYN-MED dataset showed more modest improvements, with increases in F1-score (1.3 percentage points) and recall (2.3 percentage points). The WER improved by 0.5 percentage points. The precision decreased across all datasets.

To explore the impact of providing more contextual information, an additional experiment was conducted on the subset UNITED-SYN-MED-1.5k of the UNITED-SYN-MED dataset using up to 1,000 jargon terms instead of only 50. The results, shown in Table 6.6, reveal the effects of significantly expanding the contextual vocabulary.

Metric	UNITED-SYN-MED-1.5k
WER	10.4% (+2.8)
F1	84.8% (+20.9)
Precision	98.6% (-1.3)
Recall	74.4% (+27.4)

Table 6.6: Keyword-boosting via zero-shot in-context learning with up to 1,000 jargon terms on UNITED-SYN-MED compared to Gemini-2.5-Flash baseline in percentage points.

The expanded contextual information produced dramatically improved performance in jargon recognition, with recall increasing by 27.4 percentage points and F1-score improving by 20.9 percentage points compared to the baseline Gemini-2.5-Flash. This substantial enhancement in keyword detection capability came at the cost of increased WER (2.8 percentage points). The precision remained nearly unchanged (-1.3 percentage points).

6.5 Cross-Method Performance Summary

A comprehensive comparison of all improvement methods against the established baselines reveals distinct patterns in effectiveness across the three specialized datasets.

Figure 6.1 illustrates the WER performance of each adaptation method and the Gemini-2.5-Flash baseline grouped by dataset. The results reveal that only the in-context learning approach with 50 terms consistently reduces WER across all datasets, achieving improvements of 0.8, 0.5, and 1.1 percentage points for FAA-Glossary, UNITED-SYN-MED, and Earnings-21, respectively. In contrast, both the keyword-guided adaptation and LLM-based error correction methods result in increased WER compared to the baseline. The aiOla keyword spotting approach shows minimal WER changes but still underperforms the best baseline models. Notably, the in-context learning approach with 1,000 terms, while dramatically improving jargon recognition, comes at the cost of an increase in WER.

Figure 6.2 presents the F1-score comparison, which measures the balance between precision and recall for jargon term detection. The in-context learning approach again delivers the best performance, being the only method to consistently improve F1-scores across all datasets. The approach with 50 terms achieves F1-score improvements of 2.1, 1.3, and 9.2 percentage points for FAA-Glossary, UNITED-SYN-MED, and Earnings-21, respectively. The variant with 1,000 terms on the UNITED-SYN-MED dataset produces the most substantial improvement of 20.9 percentage points, demonstrating the value of expanded contextual information for challenging medical terminology. In contrast, both the keyword-guided adaptation and

LLM-based error correction methods result in reduced F1-scores, indicating poorer jargon detection performance compared to the baseline systems.

The cross-method analysis reveals a clear performance hierarchy. The in-context learning approach consistently outperforms all other methods, achieving both improved transcription accuracy and enhanced jargon recognition. The keyword-guided adaptation with keyword spotting shows modest improvements over its own baseline but fails to match the performance of the benchmark models. The LLM-based error correction method, while showing promise in specific scenarios (particularly for recall improvement in the Earnings-21 dataset), generally degrades overall performance due to significant precision losses.

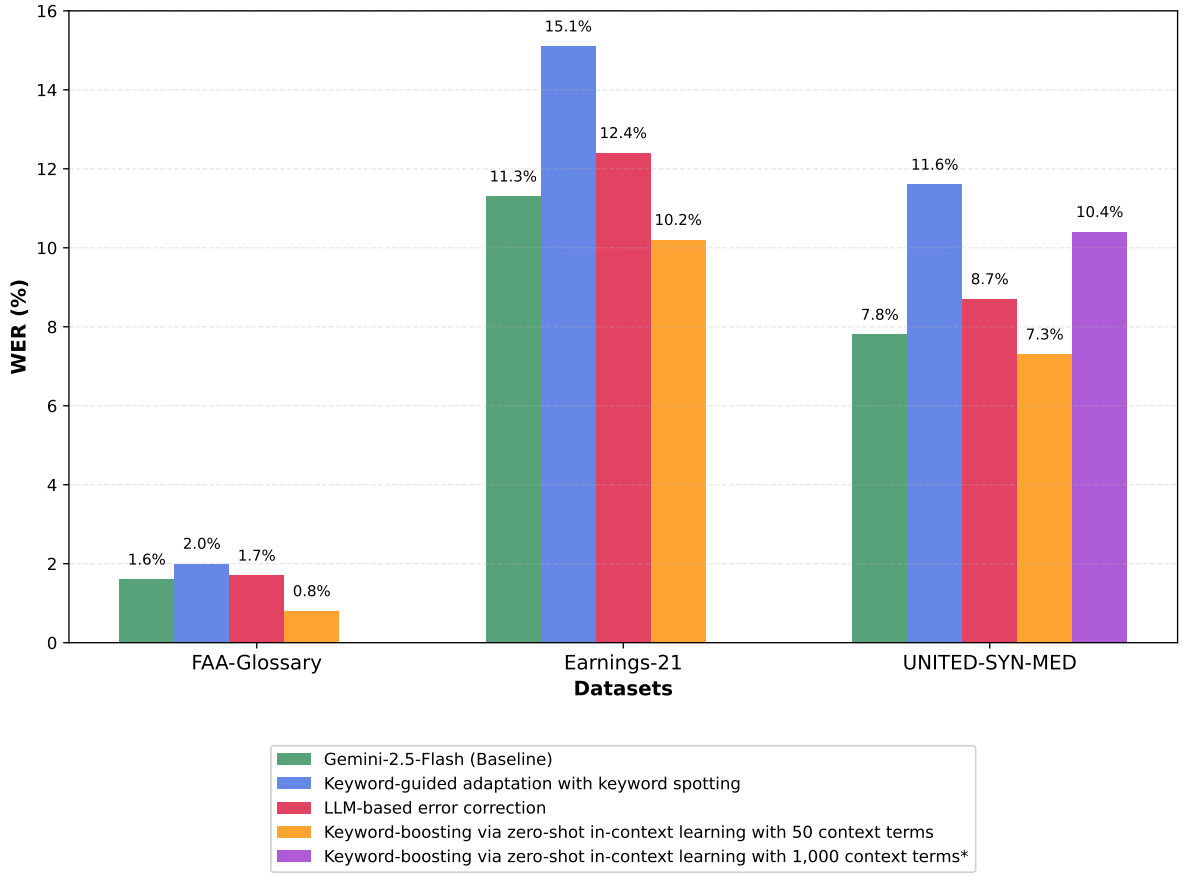

Figure 6.1: WER performance comparison of baseline and improvement methods across datasets. *For the 1,000 context terms approach, the subset UNITED-SYN-MED-1.5k was used.

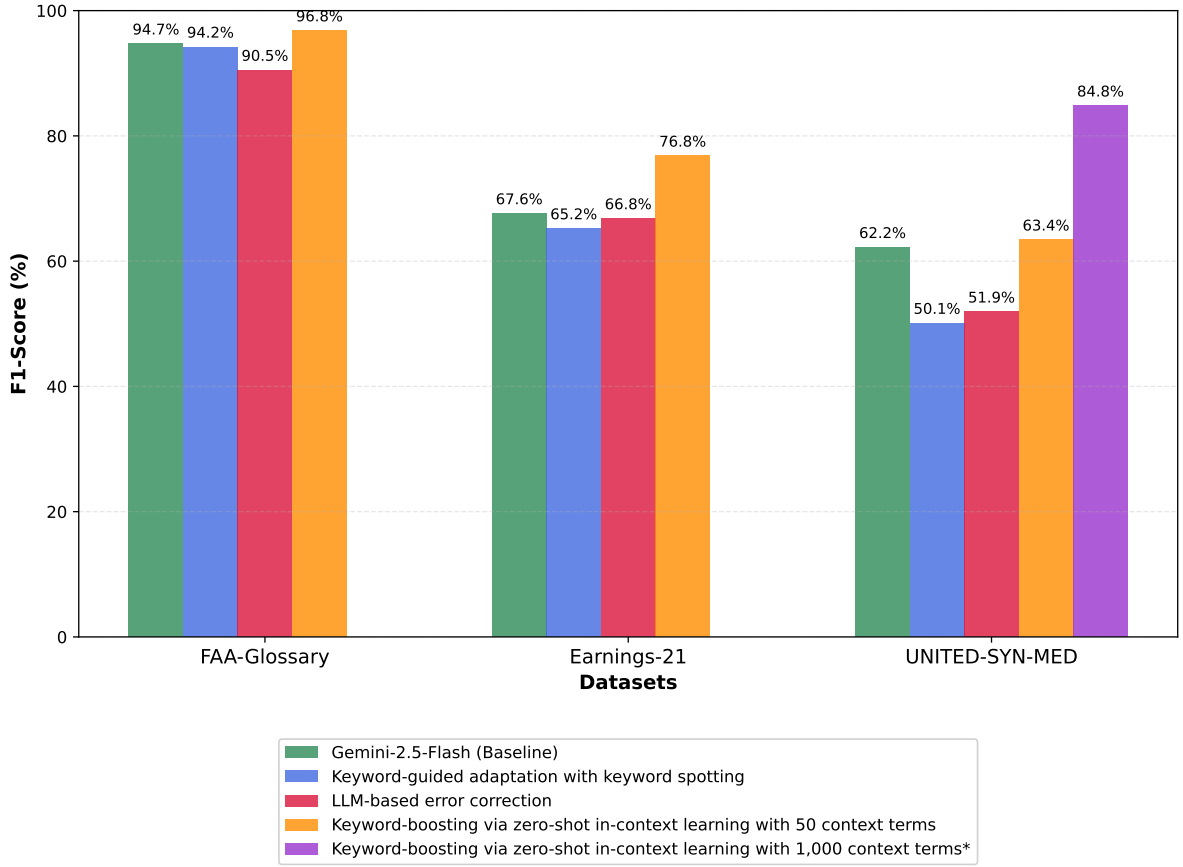

Figure 6.2: F1-score performance comparison of baseline and improvement methods across datasets. *For the 1,000 context terms approach, the subset UNITED-SYN-MED-1.5k was used.

6.6 Cost Tracking

Understanding the computational and financial costs associated with each improvement method is crucial for evaluating their practical viability in real-world applications. This section presents a comprehensive cost analysis based on token consumption patterns observed during the experimental evaluation and current pricing structures.

For the aiOla keyword-guided adaptation approach, detailed cost analysis could not be performed as aiOla does not provide transparent pricing information for their Jargonic model. The service was accessed through a free trial period, which limits the ability to assess the long-term financial implications of this approach for production deployments.

The LLM-based methods utilizing Gemini-2.5-Flash enable detailed cost tracking through token consumption monitoring. The cost calculations are based on Gemini-2.5-Flash's pricing structure: $0.30 per 1 million text input tokens, $1.00 per 1 million audio input tokens, and $2.50 per 1 million output tokens [67].

Table 6.7 presents the average token consumption per audio sample across different approaches, broken down into text input tokens, audio input tokens, output tokens, and the calculated cost per 1,000 audio samples. To calculate this a small subset of the whole dataset was sampled and the average token consumption was recorded for each approach.

Approach	**Text Input**	**Audio Input**	**Output**	**Cost per 1k**
Baseline transcription	18.0	240.8	24.5	$0.31
LLM-based error correction	377.5	0.0	25.3	$0.18*
In-context learning (50 terms)	356.0	240.8	25.0	$0.41
In-context learning (1,000 terms)	5,051.0	240.8	24.8	$1.82

Table 6.7: Average token consumption per audio sample and associated costs for different Gemini-2.5-Flash approaches. *LLM-based error correction cost excludes initial transcription.

The baseline transcription approach represents the reference point at $0.31 per 1,000 audio samples. This approach consumes 18.0 text input tokens (primarily for system prompts), 240.8 audio input tokens, and generates 24.5 output tokens on average.

With 377.5 text input tokens (containing both the original transcript and correction prompts), the LLM-based error correction method shows an apparent cost of $0.18 per 1,000 samples for the correction step alone. However, this figure is misleading as it excludes the initial transcription cost. Since post-correction requires an initial audio transcription, the total cost when using Gemini-2.5-Flash for both steps would be $0.49 per 1,000 samples ($0.31 + $0.18), representing a 58% increase over the baseline and making it more expensive than the in-context learning approach.

The in-context learning approach with 50 terms costs $0.41 per 1,000 samples, representing a 32% increase over the baseline. This moderate cost increase stems from expanded text input requirements (356.0 tokens) needed to provide contextual jargon information while maintaining the same audio processing costs.

The most significant cost impact occurs with the in-context learning approach using 1,000 terms, resulting in $1.82 per 1,000 samples. That's nearly 6 times the baseline cost. This increase is primarily attributed to the substantial text input requirements (5,051.0 tokens) needed to provide jargon vocabulary. The audio processing and output token costs remain stable across all approaches as every approach processes the same audio input and generates similar output lengths.

7 Discussion

This chapter interprets and analyzes the quantitative results presented in Chapter 6, providing insights into the performance characteristics of state-of-the-art ASR systems on jargon-heavy speech and the effectiveness of the three improvement methods investigated. The discussion is structured around the three core research questions, using the experimental findings to provide answers and interpretations, and to evaluate the strengths, limitations, and practical implications.

7.1 Defining and Representing Jargon in Speech Datasets

The first research question addresses how industry-specific jargon can be defined and represented for benchmarking. The baseline evaluation revealed distinct performance patterns across the three jargon-heavy datasets, demonstrating that the nature of the jargon, the audio quality characteristics, and the relationship between domain-specific terms and general vocabulary significantly impact recognition performance.

The Earnings-21 dataset exhibited the highest WER across all models, making it the most challenging for general transcription accuracy. This difficulty can be attributed to the real-world nature of the recordings, which introduce variability in audio quality, speaker accents, and background noise that are absent from controlled, synthetic datasets. The authentic earnings call environment presents the acoustic complexities that ASR systems encounter in practical deployments, including overlapping speech, telephone compression artifacts, and varying recording conditions. For jargon-specific performance, the Earnings-21 dataset showed intermediate F1-scores and recall rates. The specialized financial acronyms and company-specific terms present in earnings calls are typically not commonly used in everyday language, making them challenging for models trained primarily on general-domain data. However, the relatively low precision scores revealed an interesting pattern. Investigation of false positive errors showed that some issues stemmed from NER tagging inconsistencies in the ground truth annotations and ambiguous short words such as "us" versus "U.S." that suffer from text normalization processes during evaluation.

The FAA-Glossary dataset demonstrated the lowest WER across all models, making it the easiest for general transcription. This superior performance results from the synthetic generation process, which produces controlled audio quality with clear pronunciation and no background noise. The synthetic nature eliminates the acoustic variability that challenges ASR systems in real-world scenarios, creating an idealized testing environment. Notably, the FAA-

Glossary dataset also achieved the highest F1-scores and recall rates for jargon recognition, with only approximately 7-9% of keywords missed across the baseline models. This strong performance can be explained by the nature of the aviation jargon used in the dataset. Many keywords of the FAA-Glossary are common English words that acquire specialized meanings within the aviation context, such as "altitude", "aircraft", "runway", or "approach". These terms are likely well-represented in the training data of general-purpose ASR models, even if their specialized aviation meanings are not explicitly learned. This demonstrates the importance of the established jargon definition, particularly the criterion that jargon terms should be rare in general speech. Despite containing terms with high semantic importance within the aviation domain, many terms do not exhibit low frequency in general language corpora, which explains the strong baseline performance across all models.

The UNITED-SYN-MED dataset presented an intermediate WER despite being synthetically generated, indicating that the highly specialized vocabulary of drug names significantly affects transcription performance even under controlled acoustic conditions. This suggests that lexical complexity can override the advantages of clean synthetic audio when dealing with truly specialized terminology. Most significantly, the UNITED-SYN-MED dataset exhibited the lowest F1-scores and recall rates for jargon recognition across all baseline models. The highly specialized medical terms, particularly drug names, are rarely used in everyday language and are likely severely under-represented in the training data of general-purpose ASR models. This underrepresentation creates a fundamental challenge where the models lack sufficient exposure to learn robust representations of these specialized terms, leading to systematic recognition failures even in optimal acoustic conditions.

These results provide important insights into the first research question regarding how industry-specific jargon can be defined and represented in speech datasets. The FAA-Glossary results demonstrate the importance of lexical rarity in jargon definition, as many aviation terms like "altitude" and "aircraft" are common in general language, explaining the strong baseline performance. Synthetic datasets like UNITED-SYN-MED can provide controlled environments to isolate lexical complexity challenges, while real-world datasets like Earnings-21 capture the acoustic variability essential for authentic evaluation. Therefore, a combination of realistic audio conditions and comprehensive jargon coverage is essential for robust evaluation.

7.2 Evaluation Metrics for Jargon Recognition

The second research question asked which evaluation metrics best capture the performance of ASR systems on jargon-heavy speech. The experimental results confirm the hypothesis that traditional WER alone is insufficient and can be actively misleading when evaluating performance in specialized domains.

This insufficiency is clearly demonstrated by the baseline results. On the UNITED-SYN-MED dataset, Gemini-2.5-Flash achieved the lowest WER of 7.8%, a figure that might be

considered acceptable in a general context. However, its corresponding F1-score was only 62.2%, indicating that it failed to correctly identify a substantial part of the critical medical terms. The discrepancy is even more obvious with Canary-Qwen-2.5b, which posted an F1-score of only 43.5%. This proves that a model can be correct on the vast majority of common words while still failing severely on the specialized, high-semantic-value terms that are the entire focus of the application.

Similarly, on the Earnings-21 dataset, Gemini-2.5-Flash again recorded the lowest WER (11.3%) but was outperformed in jargon recognition by Whisper-large-v3, which achieved a higher F1-score (74.1% vs. 67.6%). This indicates that while one model was better at transcribing the overall call, another was more reliable at identifying the specific financial entities. An important trade-off is evident here: optimizing for overall transcription accuracy does not necessarily translate to better performance on the specialized jargon that matters most in the domain.

This shows that a multi-metric approach is essential. WER remains useful for measuring overall transcription quality, but it must be complemented by keyword-focused metrics to meaningfully evaluate a system's practical utility in a specialized domain. For this the F1-score provides a balanced and compareable measure of jargon recognition performance.

The evaluation also revealed that differentiating between precision and recall is crucial for comprehensive performance assessment. These metrics capture fundamentally different failure modes that have distinct practical implications. High recall with low precision indicates a system that successfully identifies most jargon terms but introduces many false positives. Conversely, high precision with low recall suggests a conservative system that rarely makes false identifications but misses many actual jargon terms. This precision-recall trade-off became pronounced in the improvement methods, where attempts to boost jargon recognition consistently decreased precision across all approaches. Understanding this trade-off is therefore essential for closely monitoring the effects of adaptation methods on model behavior.

An important limitation in the current evaluation framework is the absence of latency analysis and real-time performance metrics. While this study provides comprehensive accuracy assessments, the exclusive focus on transcription quality metrics overlooks temporal performance characteristics that are critical for practical deployment. Real-world applications often require near-instantaneous transcription, particularly in live scenarios such as meetings, customer service interactions, or voice-controlled systems. The temporal constraints imposed by these use cases create an additional dimension of the evaluation challenge: a system that achieves superior accuracy but requires several seconds of processing delay may be less valuable than a faster system with slightly lower accuracy. This limitation is particularly relevant when considering the LLM-based methods that demonstrated the best jargon recognition performance, as these approaches may introduce processing delays that could make them unsuitable for streaming applications where low latency is prioritized over perfect accuracy.

7.3 Low-Resource Improvement Methods

The third research question investigated what methods can improve ASR performance in recognizing industry-specific jargon without the availability of in-domain audio data. The evaluation of the three improvement methods revealed varying degrees of effectiveness and highlighted important trade-offs between different performance metrics. A consistent pattern across all methods was the decrease in precision compared to baseline performance, suggesting that adaptation techniques designed to boost specific keywords inherently introduce a bias toward increased keyword recognition, which can lead to false positive errors. This trade-off reflects a fundamental challenge: increasing sensitivity to target terms inevitably increases the risk of false positives when acoustically similar alternatives are encountered.

7.3.1 Keyword-Guided Adaptation with Keyword Spotting

The keyword-guided adaptation approach using aiOla's keyword spotting capabilities demonstrated measurable but limited improvements across the evaluated datasets compared to aiOla's baseline model without keyword spotting. The method's effectiveness varied significantly between datasets.

For the FAA-Glossary and UNITED-SYN-MED datasets, keyword spotting produced small but consistent improvements in WER, F1-score, and recall, while precision decreased slightly. These results suggest that the method had a minor positive effect on these datasets, successfully increasing the detection of target jargon terms with only modest increases in false positive rates. In contrast, the Earnings-21 dataset exhibited a different response pattern, showing major improvements in recall of 8.2 percentage points but also experiencing a substantial precision decrease of 9.5 percentage points. This trade-off resulted in only minimal overall F1-score improvement despite the significant recall gains, and the WER remained essentially unchanged.

The differential response across datasets can be explained by the varying numbers of jargon terms in each domain relative to the 50-keyword limit recommended by aiOla. The Earnings-21 dataset contains a maximum of approximately 50 unique jargon terms per earnings call, allowing the keyword spotting system to target nearly the entire jargon vocabulary. In contrast, the other two datasets contain larger jargon vocabularies, with UNITED-SYN-MED containing more than 11,000 unique drug names. The 50-keyword constraint meant that only a very small fraction of the total jargon vocabulary could be targeted for these larger domains, limiting the method's effectiveness.

While keyword spotting demonstrated measurable improvements compared to the aiOla baseline model without keyword enhancement, the overall performance remained inferior to the benchmark results achieved by Whisper-large-v3 and Gemini-2.5-Flash across nearly all metrics and datasets. This performance gap can be partially attributed to the underlying model architecture: aiOla's system is based on Whisper-large-v2 [40], which is an earlier version than the Whisper-large-v3 used in the benchmark evaluation. This generational

difference may contribute to the observed performance discrepancies, as newer model versions typically incorporate improvements in architecture, training data, and optimization techniques.

Additionally, the implementation did not utilize the phonetic representation feature available in aiOla's keyword spotting system. This feature allows users to provide phonetic transcriptions of keywords to guide pronunciation-based recognition, which could be particularly beneficial for specialized jargon terms that may have non-standard pronunciations or are commonly mispronounced. The absence of phonetic guidance may have limited the system's ability to effectively recognize jargon terms with challenging pronunciations, potentially contributing to the suboptimal performance observed.

7.3.2 LLM-Based Error Correction

The LLM-based error correction approach demonstrated significant limitations, achieving meaningful improvements only in specific circumstances while systematically degrading overall performance. The most notable positive result was a substantial recall improvement of 22.8 percentage points on the Earnings-21 dataset. This success can be attributed to the comprehensive coverage of the domain vocabulary: the 50 most frequently missed keywords represented nearly the entire jargon vocabulary for earnings calls, enabling the LLM to effectively identify and correct most missed financial terms.

However, this recall improvement was accompanied by a severe precision decrease of 18.2 percentage points, resulting in no net benefit to the F1-score (-0.8 percentage points) and increased WER. This pattern reflects the fundamental problem with the post-correction approach: while it successfully identifies more jargon terms, it introduces an even greater number of false positives, resulting in overall performance degradation.

The precision problems were consistent across all datasets, with particularly severe impacts on UNITED-SYN-MED (-28.7 percentage points) and FAA-Glossary (-9.1 percentage points). These substantial decreases indicate that the LLM systematically hallucinated jargon terms that were not present in the original audio, particularly targeting keywords explicitly mentioned in the correction prompt. The method's tendency toward false corrections outweighed its ability to identify genuine missed terms.

Two potential explanations emerge for these precision failures. First, more conservative prompting strategies might reduce false positive rates, though this study evaluated only a single prompt design. However, the prompt used was already explicitly conservative, instructing the model to "Only replace a word if it fits semantically into the context of the sentence. If the replacement makes the sentence illogical, keep the original transcription." The failure of this conservative approach suggests that prompt refinement alone may be insufficient.

The more fundamental limitation appears to be information loss during initial transcription. Once acoustic signals are processed by the initial ASR system, critical phonetic information

necessary for distinguishing similar-sounding terms is permanently lost. The LLM-based correction process lacks this acoustic context, creating an inherent disadvantage compared to approaches that process audio signals directly during transcription.

While alternative prompting strategies could potentially be investigated, the fundamental constraints of post-correction processing suggest that this approach cannot be recommended for practical deployment. The consistent precision degradation across all datasets, combined with the systematic information loss inherent in the two-stage process, renders this method inferior even to baseline models.

7.3.3 Keyword-Boosting via Zero-Shot In-Context Learning

The zero-shot in-context learning approach using Gemini-2.5-Flash produced the most consistently positive results across all evaluation metrics and datasets. This method demonstrated clear superiority over the other improvement approaches, achieving improvements in WER, F1-score, and recall compared to the baseline transcriptions for all datasets. While precision decreased across all datasets, the degradation was minimal compared to the substantial losses observed with the post-correction method.

Several factors contribute to this superior performance. First, the inherent domain knowledge embedded in the LLM's component of Gemini-2.5-Flash provides a foundation for understanding specialized terminology that may not be present in traditional ASR training data. This pre-existing knowledge allows the model to make more informed decisions when encountering ambiguous acoustic signals that could correspond to multiple words.

Second, the multimodal capabilities of Gemini-2.5-Flash enable it to directly process audio input rather than relying on potentially error-prone intermediate transcriptions. This direct audio processing eliminates the information loss that occurs when acoustic signals are first converted to text by a separate ASR system and then post-processed for correction. The model can leverage both acoustic and contextual information simultaneously to make more accurate transcription decisions.

Third, the large context window of Gemini-2.5-Flash allowed for the inclusion of comprehensive keyword lists in the prompt, providing extensive domain-specific guidance without the severe constraints faced by the keyword spotting method. This expanded context enables the model to recognize a broader range of jargon terms while maintaining better precision control than the LLM-based error correction approach.

7.3.4 Practical Implications

The practical deployment of these improvement methods requires careful consideration of both performance gains and implementation costs. The evaluation results provide important guidance for organizations considering the adoption of jargon-enhanced ASR systems in real-world scenarios.

The keyword-guided adaptation approach using aiOla's system lacks transparent cost information, making direct cost comparison challenging. However, given that in the performed experiments its overall performance was inferior to baseline Whisper-large-v3 and Gemini-2.5-Flash models across most metrics and datasets, this method cannot be recommended for practical deployment in its current form. The performance limitations suggest that the underlying aiOla model may not match the capabilities of more recent systems. However, if keyword spotting techniques could be applied to stronger base models, re-evaluation would be worthwhile to assess whether the combination of superior foundational performance and keyword enhancement could yield better results.

The LLM-based error correction method cannot be recommended for practical deployment due to fundamental limitations in the post-correction approach. Although the method initially appears cost-effective due to lower input token costs than the zero-shot in-context learning approach, this apparent advantage disappears when considering the complete workflow. Post-correction requires an initial ASR transcription, and if the highly capable model Gemini-2.5-Flash is used for this step, there is no cost advantage compared to using Gemini directly for transcription with in-context learning. While a cheaper model could potentially be used for the initial transcription, the significant performance decrease observed with post-correction does not justify potential cost savings. The method suffers from critical acoustic information loss during initial ASR transcription, permanently eliminating phonetic cues necessary for distinguishing similar-sounding jargon terms. Combined with severe precision degradation across all datasets, this information bottleneck renders the method inferior to baseline models and systematically disadvantaged compared to approaches that process audio signals directly during transcription.

The zero-shot in-context learning approach presents a more complex cost-performance trade-off. The cost analysis revealed that increasing the number of keywords in the context increases expenses. When the context size was increased by a factor of 10, the cost per 1,000 samples increased by only approximately a factor of 6. This sub-linear cost scaling can be explained by the token economics of multimodal models: audio tokens are the primary cost drivers, while adding text tokens to the prompt is relatively inexpensive. Since the audio samples remain unchanged when adding more keywords, the audio token costs remain constant, and the increased costs come primarily from the linear increase in text token consumption. The scaling experiment demonstrated that adding more keywords improved performance dramatically for domains with large jargon vocabularies, such as UNITED-SYN-MED. However, this performance improvement came with significant cost increases per audio sample. Therefore, practical deployment requires finding an optimal balance between performance requirements and cost constraints based on specific use case needs. Organizations must carefully optimize the number of keywords added to the context for each deployment scenario, considering both the size of the domain vocabulary and the acceptable cost per transcription. For organizations with critical accuracy requirements in jargon-heavy domains, the superior performance of the in-context learning approach may justify the additional costs. However, for applications where moderate accuracy improvements are sufficient, a smaller, carefully curated keyword

list may provide the optimal cost-performance balance.

In summary, this chapter has provided comprehensive insights into the challenges and opportunities of ASR systems in jargon-heavy environments. The research demonstrates that specialized vocabularies pose significant challenges that cannot be addressed by traditional evaluation approaches alone, requiring both sophisticated measurement frameworks and targeted improvement strategies. The findings reveal that while low-resource adaptation methods can provide meaningful improvements, their effectiveness varies significantly based on domain characteristics and implementation constraints. The superior performance of zero-shot in-context learning with LLMs represents a promising direction for practical deployment, though organizations must carefully balance performance gains against increased computational costs.

8 Conclusion and Outlook

This thesis explored methods for improving ASR performance on specialized industry jargon, focusing on low-resource scenarios where in-domain audio data is unavailable. Through a systematic literature review and an empirical evaluation across three datasets and three adaptation techniques, this work offers valuable insights into the current limitations of state-of-the-art ASR systems and demonstrates the potential of lightweight adaptation strategies for specialized domains.

The research was guided by three fundamental questions: how to define and represent jargon in speech datasets, which evaluation metrics best capture jargon recognition performance, and what methods can improve ASR systems for jargon recognition without requiring extensive in-domain audio data. Each question was addressed through careful methodology and empirical evaluation, leading to several important findings. The systematic literature review provided the theoretical foundation and identified promising adaptation techniques that formed the basis for the experimental investigation.

Regarding the representation of jargon in speech datasets, this thesis established a clear definition of industry-specific jargon as specialized terms that exhibit low frequency in general language, possess domain-specific meanings, and carry high semantic importance. Three diverse datasets were collected or created to represent different aspects of jargon recognition challenges. The FAA-Glossary dataset demonstrated that synthetic data can provide valuable controlled testing environments in domains where the available data is scarce. The Earnings-21 dataset highlighted the acoustic complexities of real-world recordings and the UNITED-SYN-MED dataset revealed how highly specialized terminology can challenge even state-of-the-art systems under optimal acoustic conditions.

For evaluation metrics, the research confirmed that traditional WER alone is insufficient for assessing jargon recognition performance. The keyword-focused metrics of Precision, Recall, and F1-score proved essential for capturing the specific challenges of specialized terminology recognition. The baseline evaluation revealed significant performance variations across the datasets, with F1-scores ranging from over 94% on the aviation terminology to under 63% on medical drug names, highlighting the importance of use case specific evaluation. The results underscored the limitations of current state-of-the-art ASR systems, which still struggle with highly specialized terminology, representing a barrier to professional adoption.

The third research question regarding improvement methods yielded mixed results. Three approaches were evaluated: keyword-guided adaptation using keyword spotting, LLM-based error correction, and zero-shot in-context learning. The keyword-guided adaptation showed

limited improvements, possibly constrained by vocabulary size limits. The LLM-based error correction degraded overall transcription quality while also hardly providing any jargon improvements. In contrast, zero-shot in-context learning emerged as the most promising method, achieving positive results across all datasets by providing the audio signal and jargon terms directly in the prompt to a multimodal LLM.

Looking toward future developments, several promising directions emerge from the findings of this thesis. The advancement of multimodal LLMs suggests that in-context learning approaches will become more sophisticated and cost-effective. More sophisticated prompt engineering strategies for LLM-based correction methods should be developed to encourage conservative correction behavior and reduce false positive errors. Dynamic context sizing strategies that optimize the balance between keyword coverage, performance improvement, and operational costs represent a crucial area for investigation. Future research must also address the temporal aspects of these methods through comprehensive latency analysis to evaluate their suitability for real-time applications.

This thesis demonstrates that while significant challenges remain in achieving reliable jargon recognition with current ASR technology, practical solutions exist for organizations willing to navigate the trade-offs between accuracy and cost. The continued evolution of multimodal AI systems and domain adaptation techniques promises to make specialized ASR applications increasingly viable for real-world deployment.

A Appendices

A.1 PICOC Terms and Synonyms for Literature Review

PICOC Element	Primary Keywords	Alternative Terms and Synonyms
Population	automatic speech recognition	asr, speech-to-text, speech to text, speech recognition
Intervention	adaptation	domain adaptation, customisation, optimisation, fine-tuning, error-correction, contextual, contextualized, contextualization
Comparison	sota asr systems	state-of-the-art asr systems, leading asr systems, baseline systems
Outcome	recognition performance	word error rate, wer, recognition accuracy, accuracy improvements
Context	industry-specific jargon	industry-specific terms, low-resource domain, low-data domain, out-of-vocabulary (oov), specialized terminology, technical vocabulary

Table A.1: Synonym Terms for Literature Review based on the PICOC Framework

A.2 Literature Review Query String

```
(TITLE(‘‘automatic speech recognition’’ OR asr OR ‘‘speech-to-text’’ OR ‘‘speech
to text’’ OR ‘‘speech recognition’’))
AND
(TITLE-ABS-KEY(‘‘domain adaptation’’ OR adaptation OR customisation OR op-
timisation OR ‘‘fine-tuning’’ OR ‘‘error-correction’’ OR ‘‘contextual’’ OR con-
textualized OR contextualization))
AND
(TITLE-ABS-KEY(‘‘recognition performance’’ OR ‘‘word error rate’’ OR wer OR
‘‘accuracy’’))
AND
```

```
(TITLE-ABS-KEY(jargon OR keywords OR ((domain OR rare OR ‘‘domain-specific’’
OR ‘‘industry-specific’’ OR ‘‘out-of-vocabulary’’ OR oov OR specialised OR
technical) PRE/1 (terms OR words OR terminology OR vocabulary))))
AND
(PUBYEAR > 2022)
AND
(LANGUAGE(english))
```

List of Figures

List of Tables

Acronyms

AM Acoustic Model. 4, 5

API Application Programming Interface. 23, 34–36, 40, 41, 43, 45

ASR Automatic Speech Recognition. iv, 1–24, 26–41, 44, 46–48, 50, 54–56, 58–63, 66, 67

CER Character Error Rate. 14, 15, 17

CTC Connectionist Temporal Classification. 7

E2E End-to-End. 7, 8

FN False Negative. 15, 16, 37

FP False Positive. 15, 16, 37

HMM Hidden Markov Models. 5–7

KWS Keyword Spotting. 21, 22

LLM Large Language Model. iv, 2, 8–10, 15, 21, 23, 24, 31, 33, 38, 39, 41–44, 46, 49, 51–54, 58–63

LM Language Model. 4, 5

MoE Mixture of Experts. 10, 34

NER Named Entity Recognition. 30, 55

NLP Natural Language Processing. 8, 34, 35

SALM Speech-Augmented Language Model. 9, 21, 24, 33, 35, 66

SDK Software Development Kit. 35, 36, 41, 43, 45

STT Speech-to-Text. 4

TN True Negative. 15, 16, 37

TP True Positive. 15, 16, 37

TTS Text-to-Speech. 29, 31

WER Word Error Rate. 9, 13–15, 17, 22–24, 32, 36, 39, 43, 46–51, 55–57, 59, 62

WRR Word Recognition Rate. 14

Bibliography

[1] J. Bourne. *Voice Assistant User Forecast 2024*. 2024. URL: https://www.emarketer.com/content/voice-assistant-user-forecast-2024.

[2] A. Feger. *Voice assistants are most popular AI-powered smartphone feature among adults*. 2024. URL: https://www.emarketer.com/content/voice-assistants-most-popular-ai-powered-smartphone-feature-among-adults.

[3] Ignetica. *The impact of clinical speech recognition in the Emergency Department: Results of a study at an NHS Foundation Trust*. 2021. URL: https://www.nuance.com/asset/en_uk/collateral/healthcare/success-stories/cs-hc-speech-recognition-in-emergency-department-en-uk.pdf.

[4] S. Ruan, J. O. Wobbrock, K. Liou, A. Ng, and J. A. Landay. "Comparing speech and keyboard text entry for short messages in two languages on touchscreen phones". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.4 (2018), pp. 1–23. ISSN: 2474-9567.

[5] S. Alharbi, M. Alrazgan, A. Alrashed, T. Alnomasi, R. Almojel, R. Alharbi, S. Alharbi, S. Alturki, F. Alshehri, and M. Almojil. "Automatic Speech Recognition: Systematic Literature Review". In: *IEEE Access* 9 (2021), pp. 131858–131876. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3112535.

[6] X. Luo, L. Zhou, K. Adelgais, and Z. Zhang. "Assessing the Effectiveness of Automatic Speech Recognition Technology in Emergency Medicine Settings: a Comparative Study of Four AI-Powered Engines". In: *Journal of Healthcare Informatics Research* 9.3 (2025), pp. 494–512. ISSN: 2509-498X. DOI: 10.1007/s41666-025-00193-w.

[7] SAP Australia Project Team. *Basic SAP Terms Part 2 - Master Data*. 2015. URL: https://www.youtube.com/watch?v=mK-qW2cz0pU (visited on 11/04/2025).

[8] F. R. Goss, L. Zhou, and S. G. Weiner. "Incidence of speech recognition errors in the emergency department". In: *International journal of medical informatics* 93 (2016), pp. 70–73.

[9] D. Wilson. "Failure to Communicate". In: *AeroSafety World* (2016). URL: https://flightsafety.org/asw-article/failure-to-communicate/.

[10] Speechmatics. *TRENDS AND PREDICTIONS FOR VOICE TECHNOLOGY IN 2021*. 2021. URL: https://www.speechmatics.com/wp-content/uploads/2021/01/Speechmatics-Report-Trends-Predictions-Voice-Technology-2021.pdf.

[11] S. Banerjee, A. Agarwal, and P. Ghosh. *High-precision medical speech recognition through synthetic data and semantic correction: UNITED-MEDASR*. 2024. URL: https://arxiv.org/abs/2412.00055.

[12] D. Yu and L. Deng. *Automatic Speech Recognition : A Deep Learning Approach*. London, UNITED KINGDOM: Springer London, Limited, 2014. ISBN: 9781447157793. URL: http://ebookcentral.proquest.com/lib/munchentech/detail.action?docID=1967878.

[13] B.-H. Juang and L. R. Rabiner. "Automatic speech recognition–a brief history of the technology development". In: *Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara* 1.67 (2005), p. 1.

[14] K. H. Davis, R. Biddulph, and S. Balashek. "Automatic recognition of spoken digits". In: *The Journal of the Acoustical Society of America* 24.6 (1952), pp. 637–642. ISSN: 0001-4966.

[15] D. Spicer. *Audrey, Alexa, Hal, and More*. 2021. URL: https://computerhistory.org/blog/audrey-alexa-hal-and-more/.

[16] T. K. Vintsyuk. "Speech discrimination by dynamic programming". In: *Cybernetics* 4.1 (1968), pp. 52–57. ISSN: 1573-8337. DOI: 10.1007/BF01074755.

[17] L. R. Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286. ISSN: 1558-2256. DOI: 10.1109/5.18626.

[18] M. Gales and S. Young. "The application of hidden Markov models in speech recognition". In: *Foundations and Trends® in Signal Processing* 1.3 (2008), pp. 195–304. ISSN: 1932-8346.

[19] D. Jurafsky and J. H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, with Language Models*". 3rd. 2025. URL: https://web.stanford.edu/~jurafsky/slp3/.

[20] K.-F. Lee and H.-W. Hon. "Speaker-independent phone recognition using hidden Markov models". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37.11 (2002), pp. 1641–1648. ISSN: 0096-3518.

[21] R. Prabhavalkar, T. Hori, T. N. Sainath, R. Schlüter, and S. Watanabe. "End-to-End Speech Recognition: A Survey". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 32 (2024), pp. 325–351. ISSN: 2329-9304. DOI: 10.1109/TASLP.2023.3328283.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[23] R. Bommasani, D. A. Hudson, E. Adeli, and R. Altman. *On the Opportunities and Risks of Foundation Models*. 2022. URL: https://arxiv.org/abs/2108.07258.

[24] J. Peng, Y. Wang, Y. Xi, X. Li, X. Zhang, and K. Yu. *A Survey on Speech Large Language Models for Understanding*. 2024. URL: https://arxiv.org/abs/2410.18908.

[25] V. Srivastav, S. Zheng, E. Bezzam, E. L. Bihan, N. Koluguri, P. Żelasko, S. Majumdar, A. Moumen, and S. Gandhi. *Open ASR Leaderboard: Towards Reproducible and Transparent Multilingual and Long-Form Speech Recognition Evaluation*. 2025. URL: https://arxiv.org/abs/2510.06961.

[26] Hugging Face. *Models - Automatic Speech Recognition*. 2025. URL: https://huggingface.co/models?pipeline_tag=automatic-speech-recognition&sort=trending.

[27] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever. "Robust Speech Recognition via Large-Scale Weak Supervision". In: *Proceedings of the 40th International Conference on Machine Learning*. Vol. 202. Proceedings of Machine Learning Research. PMLR, 2023, pp. 28492–28518. URL: https://proceedings.mlr.press/v202/radford23a.html.

[28] OpenAI. *Whisper*. 2025. URL: https://huggingface.co/openai/whisper-large-v3.

[29] OpenAI. *Whisper*. 2025. URL: https://github.com/openai/whisper.

[30] OpenAI. *Introducing Whisper*. 2022. URL: https://openai.com/index/whisper/.

[31] NVIDIA. *Canary-Qwen-2.5B*. 2025. URL: https://huggingface.co/nvidia/canary-qwen-2.5b.

[32] Z. Chen, H. Huang, A. Andrusenko, O. Hrinchuk, K. C. Puvvada, J. Li, S. Ghosh, J. Balam, and B. Ginsburg. *Salm: Speech-augmented language model with in-context learning for speech recognition and translation*. 2024.

[33] NVIDIA. *Speech-agumented Large Language Models (SpeechLLM)*. 2024. URL: https://docs.nvidia.com/nemo-framework/user-guide/24.09/nemotoolkit/multimodal/speech_llm/intro.html.

[34] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, and E. Rosen. *Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities*. 2025. URL: https://arxiv.org/abs/2507.06261.

[35] W. Fedus, B. Zoph, and N. Shazeer. "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity". In: *Journal of Machine Learning Research* 23.120 (2022), pp. 1–39. ISSN: 1533-7928.

[36] J. Li, R. Gadde, B. Ginsburg, and V. Lavrukhin. "Training Neural Speech Recognition Systems with Synthetic Speech Augmentation". In: *CoRR* abs/1811.00707 (2018). URL: http://arxiv.org/abs/1811.00707.

[37] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. "Librispeech: An ASR corpus based on public domain audio books". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964.

[38] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux. *VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation*. 2021. URL: https://arxiv.org/abs/2101.00390.

[39] A. Conneau, M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Rivera, and A. Bapna. *Fleurs: Few-shot learning evaluation of universal representations of speech*. 2023.

[40] A. Shamsian, A. Navon, N. Glazer, G. Hetz, and J. Keshet. *Keyword-guided adaptation of automatic speech recognition*. 2024. URL: https://arxiv.org/abs/2406.02649.

[41] L. Šmídl, J. Švec, D. Tihelka, J. Matoušek, J. Romportl, and P. Ircing. "Air traffic control communication (ATCC) speech corpora and their use for ASR and TTS development". In: *Language Resources and Evaluation* 53.3 (2019), pp. 449–464. ISSN: 1574-020X.

[42] M. Del Rio, N. Delworth, R. Westerman, M. Huang, N. Bhandari, J. Palakapilly, Q. McNamara, J. Dong, P. Żelasko, and M. Jetté. *Earnings-21: A Practical Benchmark for ASR in the Wild*. 2021. DOI: 10.21437/Interspeech.2021-1915.

[43] A. Rosenberg, Y. Zhang, B. Ramabhadran, Y. Jia, P. Moreno, Y. Wu, and Z. Wu. "Speech Recognition with Augmented Synthesized Speech". In: *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. 2019, pp. 996–1002. DOI: 10.1109/ASRU46091.2019.9003990.

[44] K. C. Yuen, L. Haoyang, and C. E. Siong. *ASR Model Adaptation for Rare Words Using Synthetic Data Generated by Multiple Text-To-Speech Systems*. 2023. DOI: 10.1109/apsipaasc58517.2023.10317116.

[45] A. Dhaka and N. R. Mahapatra. *Practical Considerations for Selecting Metrics to Evaluate Automatic Speech Recognition Systems*. 2025.

[46] Hugging Face. *Evaluation metrics for ASR*. 2025. URL: https://huggingface.co/learn/audio-course/chapter5/evaluation.

[47] M. Malik, M. K. Malik, K. Mehmood, and I. Makhdoom. "Automatic speech recognition: a survey". In: *Multimedia Tools and Applications* 80.6 (2021), pp. 9411–9457. ISSN: 1573-7721. DOI: 10.1007/s11042-020-10073-7.

[48] T. K, J. James, D. Gopinath, and M. K. *Advocating Character Error Rate for Multilingual ASR Evaluation*. 2024, pp. 4926–4935. DOI: 10.18653/v1/2025.findings-naacl.277.

[49] K. Tomanek, J. Tobin, S. Venugopalan, R. Cave, K. Seaver, J. R. Green, and R. Heywood. "Large Language Models As A Proxy For Human Evaluation In Assessing The Comprehensibility Of Disordered Speech Transcription". In: *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2024, pp. 10846–10850. DOI: 10.1109/ICASSP48485.2024.10447177.

[50] A. Arra, G. Achour, A. Payan, E. Harrison, and D. Mavris. "Automatic Speech Recognition Model Fine-Tuning and Development of a New Evaluation Metric for Terminal Airspace Safety Analysis". In: *AIAA AVIATION FORUM AND ASCEND 2024*. 2024, p. 4570.

[51] H. Schütze, C. D. Manning, and P. Raghavan. *Introduction to information retrieval*. Vol. 39. Cambridge University Press Cambridge, 2008.

[52] S. Seyfarth and P. Zhao. *Evaluating an automatic speech recognition service*. 2020. URL: https://aws.amazon.com/de/blogs/machine-learning/evaluating-an-automatic-speech-recognition-service/.

[53] L. A. Kumar, D. K. Renuka, B. R. Chakravarthi, and T. Mandl. *Automatic Speech Recognition and Translation for Low Resource Languages*. John Wiley and Sons, 2024. ISBN: 1394214170.

[54] W. M. Kouw and M. Loog. "An introduction to domain adaptation and transfer learning". In: *CoRR* abs/1812.11806 (2018). URL: http://arxiv.org/abs/1812.11806.

[55] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*. Vol. 1. MIT press Cambridge, 2016.

[56] A. Carrera-Rivera, W. Ochoa, F. Larrinaga, and G. Lasa. "How-to conduct a systematic literature review: A quick guide for computer science research". In: *MethodsX* 9 (2022), p. 101895. ISSN: 2215-0161. DOI: 10.1016/j.mex.2022.101895. URL: https://www.sciencedirect.com/science/article/pii/S2215016122002746.

[57] A. Asbag. *Introducing Jargonic: The World's Most Accurate Industry-Tuned ASR Model*. 2025. URL: https://aiola.ai/blog/introducing-jargonic-asr/.

[58] N. M. Matasyoh, R. A. Zeineldin, and F. Mathis-Ullrich. "Optimising speech recognition using LLMs: an application in the surgical domain". In: *Current Directions in Biomedical Engineering* 10.1 (2024). ISSN: 2364-5504.

[59] Cambridge Dictionary. *jargon*. 2025. URL: https://dictionary.cambridge.org/dictionary/english/jargon.

[60] Federal Aviation Administration. *Glossary*. 2023. URL: https://www.faa.gov/regulationspolicies/handbooksmanuals/aviation/phak/glossary.

[61] Boson AI. *Higgs Audio V2: Redefining Expressiveness in Audio Generation*. https://github.com/boson-ai/higgs-audio. GitHub repository. Release blog available at https://www.boson.ai/blog/higgs-audio-v2. 2025.

[62] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, and M. Funtowicz. *Huggingface's transformers: State-of-the-art natural language processing*. 2019. URL: https://arxiv.org/abs/1910.03771.

[63] Hugging Face. *Pipeline*. 2025. URL: https://huggingface.co/docs/transformers/pipeline_tutorial.

[64] O. Kuchaiev, J. Li, H. Nguyen, O. Hrinchuk, R. Leary, B. Ginsburg, S. Kriman, S. Beliaev, V. Lavrukhin, and J. Cook. "NeMo: a toolkit for building AI applications using Neural Modules". In: *CoRR* abs/1909.09577 (2019). URL: http://arxiv.org/abs/1909.09577.

[65] Google. *Gemini Developer API*. 2025. URL: https://ai.google.dev/gemini-api/docs.

[66] N. Vaessen. *jiwer*. 2025. URL: https://jitsi.github.io/jiwer/.

[67] Google. *Gemini Developer API Pricing*. 2025. URL: https://ai.google.dev/gemini-api/docs/pricing.