Optimizing Rare Word Accuracy in Direct Speech Translation with a Retrieval-and-Demonstration Approach

Master's Thesis of

Siqi Li

Artificial Intelligence for Language Technologies (AI4LT) Lab
Institute for Anthropomatics and Robotics (IAR)
KIT Department of Informatics

Reviewer:	Prof. Dr. Jan Niehues
Second reviewer:	Prof. Dr.-Ing. Rainer Stiefelhagen
Advisor:	M.Sc. Danni Liu

15. Jan 2024 – 15. July 2024

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

I declare that I have developed and written the enclosed thesis completely by myself, and have not used sources or means without declaration in the text.

PLACE, DATE

. . . . . .

(Siqi Li)

Abstract

Direct speech translation(ST) models often struggle with rare words. Incorrect translation of these words can severely impact translation quality and user trust. While rare word translation is inherently challenging for neural models due to sparse learning signals, real-world scenarios often allow access to translations of past recordings on similar topics.

To leverage these valuable resources, we propose a novel *retrieval-and-demonstration* approach to enhance rare word translation accuracy in direct speech translation (ST) tasks. The basic idea of our approach is to retrieve external sentences containing relevant rare word information and demonstrate them to the ST system to improve its rare word translation performance.

Our method involves a three-step process: *first*, adapting existing ST models to gain the ability to incorporate examples to enhance rare word translation, similar to in-context learning (ICL). This step is completed by fine-tuning the ST model with training data prepended with example sentences. *Second*, developing a cross-modal (speech-to-speech, speech-to-text, text-to-text) retrieval model to identify relevant examples from an external dataset containing the target rare words. The retrieval model is inspired by the Dense Passage Retrieval architecture, which incorporates a dual-encoder to map query and passage to the latent space. We propose to use joint speech-and-text encoders for our cross-modal retrieval tasks. *Third*, during the inference phase, we retrieve an example sentence for each sentence to translate as a demonstration to guide the ST model in better translating rare words.

We utilize the MuST-C dataset to construct a reduced training set by moving rare words to a synthesized rare words test set for evaluation and a rare-word pool from which examples are retrieved.

Our experiments demonstrate that standard ST models can be effectively adapted to leverage examples for rare word translation, improving rare word translation accuracy over the baseline by 17.6% with gold examples and 8.5% with retrieved examples. Moreover, our speech-to-speech retrieval approach outperforms other modalities and exhibits higher robustness to unseen speakers.

Zusammenfassung

Direkte Sprachübersetzungsmodelle (ST) haben oft Schwierigkeiten mit seltenen Wörtern. Eine falsche Übersetzung dieser Wörter kann die Übersetzungsqualität und das Vertrauen der Nutzer erheblich beeinträchtigen. Da die Übersetzung seltener Wörter aufgrund spärlicher Lernsignale für neuronale Modelle von Natur aus herausfordernd ist, ermöglichen reale Szenarien häufig den Zugang zu Übersetzungen vergangener Aufnahmen zu ähnlichen Themen.

Um diese wertvollen Ressourcen zu nutzen, schlagen wir einen neuartigen *Retrieval-and-Demonstration*-Ansatz vor, um die Genauigkeit der Übersetzung seltener Wörter in direkten Sprachübersetzungsaufgaben zu verbessern. Die Grundidee unseres Ansatzes besteht darin, externe Sätze mit relevanten Informationen zu seltenen Wörtern abzurufen und sie dem ST-System zu demonstrieren, um dessen Leistung bei der Übersetzung seltener Wörter zu verbessern.

Unsere Methode umfasst einen dreistufigen Prozess: *Erstens* die Anpassung bestehender ST-Modelle, um die Fähigkeit zu erlangen, Beispiele zur Verbesserung der Übersetzung seltener Wörter einzubeziehen, ähnlich wie beim In-Context-Learning (ICL). Dieser Schritt wird durch Feinabstimmung des ST-Modells mit Trainingsdaten, die mit Beispielsätzen versehen sind, abgeschlossen. *Zweitens* die Entwicklung eines cross-modalen (Sprache-zu-Sprache, Sprache-zu-Text, Text-zu-Text) Retrieval-Modells, um relevante Beispiele aus einem externen Datensatz mit den Zielwörtern zu identifizieren. Das Retrieval-Modell ist inspiriert von der Dense Passage Retrieval-Architektur, die einen Dual-Encoder verwendet, um Abfrage und Passage in den latenten Raum zu überführen. Für unsere cross-modalen Retrieval-Aufgaben schlagen wir die Verwendung gemeinsamer Sprach-und-Text-Encoder vor. *Drittens* rufen wir während der Inferenzphase für jeden zu übersetzenden Satz ein Beispielsatz als Demonstration ab, um das ST-Modell bei der besseren Übersetzung seltener Wörter zu unterstützen.

Wir verwenden den MuST-C-Datensatz, um einen reduzierten Trainingssatz zu erstellen, indem wir seltene Wörter in einen synthetisierten Testsatz für seltene Wörter zur Evaluierung und einen seltenen Wörterpool verschieben, aus dem Beispiele abgerufen werden.

Unsere Experimente zeigen, dass Standard-ST-Modelle effektiv angepasst werden können, um Beispiele zur Übersetzung seltener Wörter zu nutzen, was die Genauigkeit der Übersetzung seltener Wörter im Vergleich zur Basislinie um 17,6% mit goldenen Beispielen und um 8,5% mit abgerufenen Beispielen verbessert. Darüber hinaus übertrifft unser Sprache-zu-Sprache-Retrieval-Ansatz andere Modalitäten und zeigt eine höhere Robustheit gegenüber unbekannten Sprechern.

Contents

List of Figures

List of Tables

1. Introduction

Speech translation is the process of automatically converting spoken language from one language (source language) to another (target language). It is a complex task that involves understanding and processing human speech, recognizing the words being spoken, understanding the context and meaning of those words, translating them into another language, and finally producing the translated speech in the target language. Speech translation is used in various applications, such as real-time communication between speakers of different languages, multilingual meetings, and providing accessibility features in multimedia content. There are generally two common approaches to Speech Translation(ST): cascade ST and direct ST. The cascaded approach uses an Automatic Speech Recognition (ASR) model to generate the transcript from the audio in the source language and then a Machine Translation (MT) model to translate it into the target language. On the contrary, the direct ST system does not have the intermediate transcript and directly translates the speech in the source language into the target languages. With the advent of deep learning, the end-to-end approach has been developed recently and proven to have comparable performance to the cascaded approach [15]. In this thesis, we focus on end-to-end speech translation.

1.1. Motivation

End-to-end speech translation, which directly translates spoken language from one language to another without intermediate steps (like transcribing the speech into text in the source language), faces unique challenges in handling rare words due to their sparsity. On the other hand, the rare words in ST are crucial because they are important to understand the meaning of a sentence[88] or a speech. This is signified in scientific or academic scenarios, where rare words with specific meanings are more likely to appear. Mistakes in translating rare words also can undermine users' confidence in the translation system.

Compared with common words, the difficulty in translating rare words stems from two main factors: firstly, from a data perspective, these words are infrequent and have limited valid translation options. Unlike common words, which can be rendered in the target language using synonyms or paraphrases, rare words typically have only a few acceptable translations, restricting expressive freedom. Secondly, from a modeling perspective, the scarcity of these terms means that during training, the models do not receive enough examples to learn effectively, making it difficult for them to recognize and translate these terms accurately.

Improving rare word translation directly boosts the overall accuracy and reliability of speech translation systems. Research on rare word translation is crucial in 1) developing methods to learn from rare examples, also known as few-shot learning, which allows

models to learn from a minimal number of examples; 2) developing strategies to improve the model's ability to generalize from known data to new, unseen instances.

Hence, this work aims to develop the approach of *retrieval-and-demonstration* approach to improve rare word translation accuracy and enhance end-to-end speech translation performance. The basic idea of our approach is: For each sentence to translate that potentially contains the rare word, retrieve another sentence that contains the same rare word as an example. Then, demonstrate this sentence and its translation to the adapted ST model to facilitate its translation of the original sentence to translate. The adapted ST model is an existing ST model adapted to instill its ability to extract information from prepended examples.

The retrieval-and-demonstration approach we proposed tries to address the aforementioned difficulties as follows: 1) Limited valid translation options: By retrieving and demonstrating sentences that contain the same rare words to translate, we provide the ST system additional information about how the rare word should be correctly translated, which helps to provide translation options. 2) Lack of Data: By instilling the ST model's in-context learning ability, the ST model is able to process rare word information better, even if they only appear once in the demonstration.

1.2. Problem Statement and Research Question

Research Question 1: In what ways can the demonstration of sentences containing specific rare words from an external dataset improve the accuracy of rare word translation?

The response to this question guides us toward devising a systematic approach for automatically identifying and retrieving such demonstrative sentences.

Research Question 2: What methodologies can be developed to systematically extract sentences from an external dataset that share rare words with the sentence targeted for translation, thereby serving as a demonstrative example?

The effectiveness of this approach must then be gauged by examining the extent to which these demonstrations affect the overall translation performance and the precision with which rare words are translated.

Research Question 3: What criteria should be used to evaluate both the overall translation performance and the accuracy of rare words? Furthermore, how can we assess the impact of the retrieval and demonstration of example sentences on these two criteria?

1.3. Thesis Outline

The rest of this work is structured as follows.

Chapter 2 presents the foundational context and research pertinent to this thesis, encompassing models of speech translation, existing approaches that are used to improve rare word translation accuracy in both machine translation and speech translation, as well as the concept of in-context learning and retrieval-augmented translation.

In Chapter 3, we explain the approach we proposed. We suggest a new approach of retrieving and demonstrating examples to help get better at translating rare words in speech translation.

In Chapter 4, we introduce the setup of our experiments. We propose a way to synthesize the dataset for a better evaluation of rare word translation performance. Then we introduces the model configuration during training and inference, and our evaluation metrics we used.

Chapter 5 details the results and findings derived from our experiments and engages in a discussion of these outcomes.

Lastly, Chapter 6 summarizes the principal conclusions drawn from this research, addresses the posited research questions, and outlines the future research directions to extend this work.

2. Background and Related Work

This chapter formally explains the concepts and works related to ST. Firstly, Section 2.1 introduces the background knowledge of ST systems, from cascade ST systems to direct ST systems. After that, sequence-to-sequence learning approaches and various common encoder-decoder models for ST encoder-decoder models are presented in Section 2.2, including the RNN-based encoder-decoder model, transformer model, conformer model, and toolkits in end-to-end speech translation. In Section 2.3, we present the previous works focusing on rare words in ASR, MT, and ST. Afterward, Section 2.4 introduces the in-context learning approach and its application in ASR, MT, and ST. Last not the least, section 2.5 presents the retrieval approach and retrieval-augmented recognition and translation methods.

2.1. Speech Translation

Speech Translation is a process that converts spoken language into another target language. Speech translation systems can be broadly categorized into two types: cascade ST systems and direct ST systems. This section will focus on introducing these two types of systems and explain why this thesis focuses on the speech translation of end-to-end systems.

2.1.1. Cascade ST System

Cascade ST systems, also known as pipeline systems, are the traditional approach to speech translation, which was introduced in [154]. They consist of two components: an ASR and an MT system. As shown in Figure 2.1, the process involves two main steps: ASR and MT.

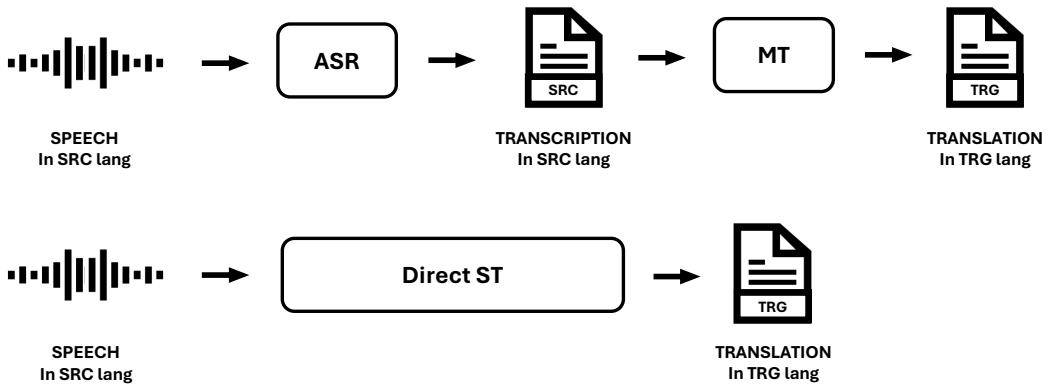

Figure 2.1.: Comparison between cascade system and end-to-end system

During the ASR, the spoken input in the source language is first transcribed into text. This step converts the audio signals into a textual representation, capturing the content of the spoken language as accurately as possible. Then, the output from the ASR system is fed into the MT component, which translates it into the target language. This step focuses on understanding the source language text semantically and generating a text translation in the target language.

However, the cascade system has several challenges. For example, errors from the ASR component (e.g., misrecognition or misinterpretations) can propagate into the MT component, affecting the overall translation quality. Also, the MT component may lack access to the speech's prosodic features (intonation, stress, rhythm), which can lead to misinterpretations of context or sentiment.

2.1.2. Direct ST System

In contrast to the two-layer architecture of the cascade system, the direct ST system, in another way, the end-to-end(E2E) ST systems, aims to directly translate spoken language into text or speech in another language, bypassing the intermediate textual representation used in cascade systems as shown in Figure 2.1. This approach was explored by [11, 17, 43] and involves a single model or a tightly integrated system that performs both recognition and translation simultaneously. Although the end-to-end system requires large amounts of parallel speech-to-text data in multiple languages, which is harder to obtain than the training data of cascade system and requires a more complex model to capture the nuances of both speech recognition and language translation within a single model, it still has grown popular significantly. The end-to-end system can significantly reduce error propagation and compounding problems of the cascade system, eliminating the intermediate transcription step. Also, the end-to-end models have the potential to better utilize the acoustic features of the speech, such as tone and pause, which can provide additional context for more accurate translation. Last but not least, end-to-end systems can be more efficient, as they require only one processing step rather than two, potentially leading to faster translation times[145]. End-to-end speech translation has gained more and more use in various areas, and there are lots of works on end-to-end systems that have shown excellent performance in speech translation[146]. However, at the same time, the lack of domain-specific training data for terminology translation has exposed challenges for further end-to-end speech translation performance enhancement. Thus, in this thesis, we also focus on enhancing the terminology translation end-to-end speech translation.

2.2. Sequence-to-Sequence Learning

Speech translation task involves transforming an input sequence into a desired target sequence. Similar tasks like automatic speech recognition, automatic summarization, and machine translation also fall into this category. These tasks can be best expressed as sequence-to-sequence(Seq2Seq) learning problems, which can be formalized in finding a mapping f from an input sequence of n vectors $X_{1:n}$ to a sequence of m target vectors $Y_{1:m}$, whereas the number of target vectors m is unknown apriori and depends on the input

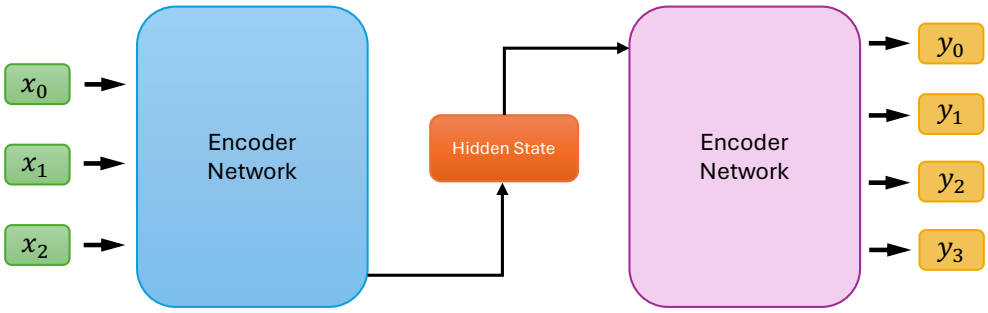

Figure 2.2.: Encoder-Decoder Models

sequence, as shown in Equation 2.1:

$$f : X_{1:n} \rightarrow Y_{1:m} \tag{2.1}$$

Seq2Seq Learning can be realized in several ways, with the encoder-decoder model being the most common. The encoder-decoder architecture was first proposed by [147] in 2014 using recurrent neural networks (RNNs) in tandem as encoder and decoder. In this approach, the encoder processes the input sequence into a fixed-length vector representation, which the decoder then uses to generate the output sequence as shown in Figure 2.2. This architecture laid the foundation for many subsequent advancements. In 2017, Vaswani et al. introduced the Transformer and thereby gave birth to transformer-based encoder-decoder models that utilize attention mechanisms to process input[151]. Later in 2020, Gulati et al.[55] introduced the Conformer model which integrates the strengths of both convolutional neural networks (CNNs) and Transformers to effectively process sequential data. This section will delve into a more detailed exploration of these models.

2.2.1. RNN-based Encoder-Decoder Model

The encoder-decoder model is an architectural framework often used to implement Seq2Seq models. The encoder neural network reads and encodes a source sentence into a fixed-length vector in the basic encoder processes. The decoder then generates a translation based on this encoded vector. The entire encoder-decoder system, which includes both the encoder and the decoder for a language pair, is jointly trained to maximize the probability of producing the correct translation given the source sentence [147, 30]. The RNN-based encoder-decoder model's structure can be shown in Figure 2.3

A potential issue with the basic RNN-based encoder-decoder approach is that the neural network must compress all the necessary information of a source sentence into a fixed-length vector. This can make it challenging for the network to handle long sentences, especially those longer than the sentences in the training corpus. To address this, attention-based encoder-decoder models have been proposed for constructing encoder-decoder models. Unlike the basic encoder-decoder model, which encodes an entire input sentence into a single fixed-length vector, the attention-based encoder-decoder model encodes the input sentence into a sequence of vectors and adaptively selects a subset of these vectors while decoding the translation. This approach allows the neural translation model to avoid

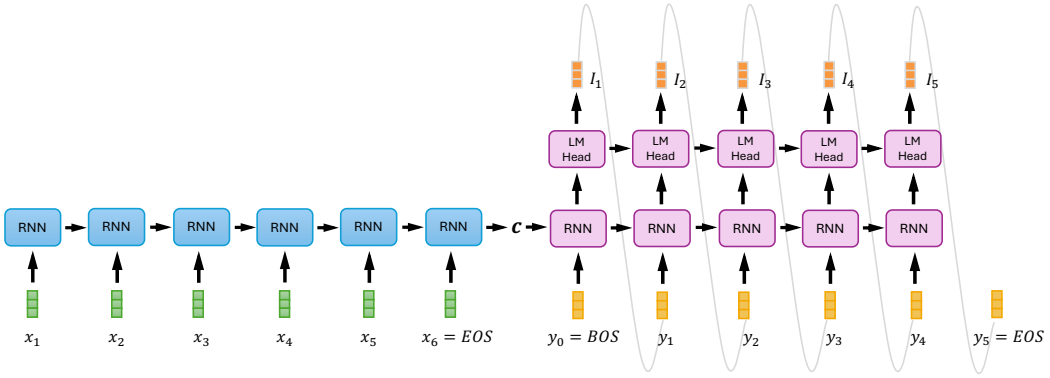

Figure 2.3.: Structure of basic Encoder-Decoder RNN

compressing all the information of a source sentence into a single fixed-length vector, regardless of its length.[98, 10]

Both the encoder and decoder are implemented using Recurrent Neural Networks (RNNs) or one of their more advanced variants, such as Long Short-Term Memory (LSTM) networks or Gated Recurrent Units (GRUs). An end-to-end speech translation system using an attention-based encoder-decoder was first proposed in [17]. In this system, both the encoder and decoder utilize LSTM networks. The encoder is constructed as a multi-layered bidirectional LSTM network, which processes the input sequence and produces a sequence of outputs. The decoder state is initialized with the last state of the encoder, and subsequent states are computed using LSTM units.

2.2.2. Attention Mechanism and Transformer-based Encoder-Decoder Model

In 2017, Vaswani et al.[151] introduced the Transformer, giving birth to transformer-based encoder-decoder models. Analogous to RNN-based encoder-decoder models, transformer-based encoder-decoder models consist of an encoder and a decoder, which are both stacks of residual attention blocks. The key innovation of transformer-based encoder-decoder models is that such residual attention blocks can process an input sequence X1:n of variable length n without exhibiting a recurrent structure. Not relying on a recurrent structure allows transformer-based encoder-decoders to be highly parallelizable, which makes the model orders of magnitude more computationally efficient than RNN-based encoder-decoder models on modern hardware.

The architecture of the transformer-based encoder-decoder model is shown in Figure 2.4

A closer look at the architecture shows that the transformer-based encoder is a stack of residual encoder blocks. Each encoder block contains a bi-directional self-attention layer; the bi-directional self-attention layer puts each input vector $x_j^{'}$ into relation with all input vectors $x_i^{'}$, ..., $x_n^{'}$ and by doing so transforms the input vector to a more "refined" contextual representation of itself, defined as $x_j^{''}$. Thereby, the first encoder block transforms each input vector of the input sequence $X_{1:n}$ from a context-independent vector representation to a context-dependent vector representation $x_{i:n}^{'}$, and the following encoder blocks further refine this contextual representation until the last encoder block outputs the final contextual

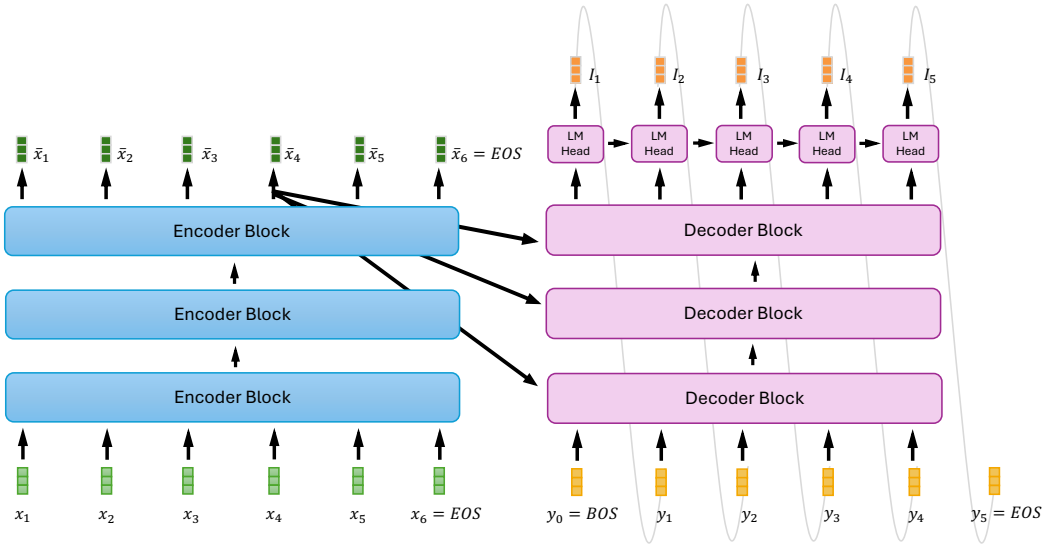

Figure 2.4.: Transformer-based Encoder-decoder

encoding $x^{'''}_{i:n}$. The bi-directional self-attention works as follows: Each input vector $x^{'}_{i}$ of an input sequence $X^{'}_{i:n}$ of an encoder block is projected to a key vector k_i, value vector v_i and query vector q_i through three trainable weight matrics W_q, W_v, W_k:

$$Q_i = W_q x^{'}_i \tag{2.2}$$

$$V_i = W_v x^{'}_i \tag{2.3}$$

$$K_i = W_k x^{'}_i \tag{2.4}$$

After the projection, each query vector q_j is compared to all key vectors k_1, ..., k_n. The more similar one of the key vectors k_1, ..., k_n is to query q_j, the more important is the corresponding value vector v_j for the output vectors $x^{''}_j$. More specifically, an output vector $x^{''}_j$ is defined as the weighted sum of all value vectors plus the input vector $x^{'}_j$. Thereby, the weights, or the so-called attention score, are proportional to the cosine similarity between q_j and the respective key vectors k_1, ..., k_n, which is mathematically expressed by softmax $(K^T_{1:n} q_j)$.

Instead of having a single set of queries, keys, and values(single-head attention), the transformer enables the creation of multiple sets (multi-head attention), each projecting the inputs into different representation spaces. By having different linear projections of the inputs, the model learns a richer representation and achieves much stronger performance. Importantly, owing to parallelization, the total computation cost of multi-headed attention remains the same as that of single-head attention.

The decoder part looks similar to the encoder. The transformer-based decoder is a stack of decoder blocks followed by a dense layer, the "LM head". The stack of decoder blocks maps the contextualized encoding sequence X1:n and a target vector sequence prepended by the BOS vector and cut to the last target vector, i.e., $Y_{0:j-1}$, in order to predict the distribution $P(y_i|Y_{0:i-1}, X_{1:n})$. In contrast to transformer-based encoders, in transformer-based decoders, the encoded output vector y_i should be a good representation

of the next target vector y_{i+1} and not of the input vector itself. To meet these requirements, each decoder block contains a uni-directional self-attention layer, followed by a cross-attention layer. The uni-directional self-attention layer puts each of its input vectors $y_j^{'}$ only into relation with all previous input vectors $y_i^{'}$, with $i \leq j$ for all $j \in \{1, \ldots, n\}$ to model the probability distribution of the next target vectors. The cross-attention layer puts each of its input vectors $y_j^{''}$ into relation with all contextualized encoding vectors $x_{1:n}$ to condition the probability distribution of the next target vectors on the input of the encoder as well. Each decoder block consists of a uni-directional self-attention layer, followed by a cross-attention layer and two feed-forward layers. The uni-directional self-attention layer puts each of its input vectors only in relation to all previous input vectors. This helps model the probability of the next target vector. The cross-attention layer uses the contextualized encoding vectors to condition the probability distribution of the next target vector. So, the uni-directional self-attention layer is responsible for conditioning each output vector on all previous decoder input vectors and the current input vector, and the cross-attention layer is responsible for further conditioning each output vector on all encoded input vectors. In summary, the Transformer-based encoder-decoder model excels at its ability to parallel processing and manage long-range dependencies. However, it still struggles with capturing local, fine-grained patterns and can become computationally costly for long sequences due to the quadratic scaling of their self-attention mechanism, as illustrated in Figure 2.5.

2.2.3. Conformer-based Encoder-Decoder Model

The Conformer(Convolution-augmented Transformer) model is proposed by Gulati et. al.[55]. It was introduced to address specific limitations of the Transformer model, particularly in handling the nuanced, local contexts of audio signals. Transformer models are good at capturing content-based global interactions, while CNNs exploit local features effectively. Conformer combines convolution neural networks and transformers to model both local and global dependencies of an audio sequence in a parameter-efficient way. The convolutional layers are integrated into a conformer to capture local dependencies and fine-grained patterns in the data. This is particularly useful for speech recognition, where such local patterns (e.g., phonetic nuances) are crucial. This combination of convolution module and transformer allows the Conformer to process sequences with a level of detail and context awareness.

2.2.4. Applications in Speech Translation

The transformer has been a successful strategy for getting state-of-the-art (SOTA) results in many NLP tasks as well as in other areas[52]. Below, we provide the efforts made to handle ST tasks using the attention mechanism and transformer within the Seq2Seq framework. This subsection will delve into recent works and toolkits on the seq2seq speech translation method.

The Transformer model, originally designed for text-based tasks, can face challenges when directly applied to acoustic inputs, primarily due to the inherent differences between language text and audio signals. Unlike text, where tokens (words or characters) are discrete

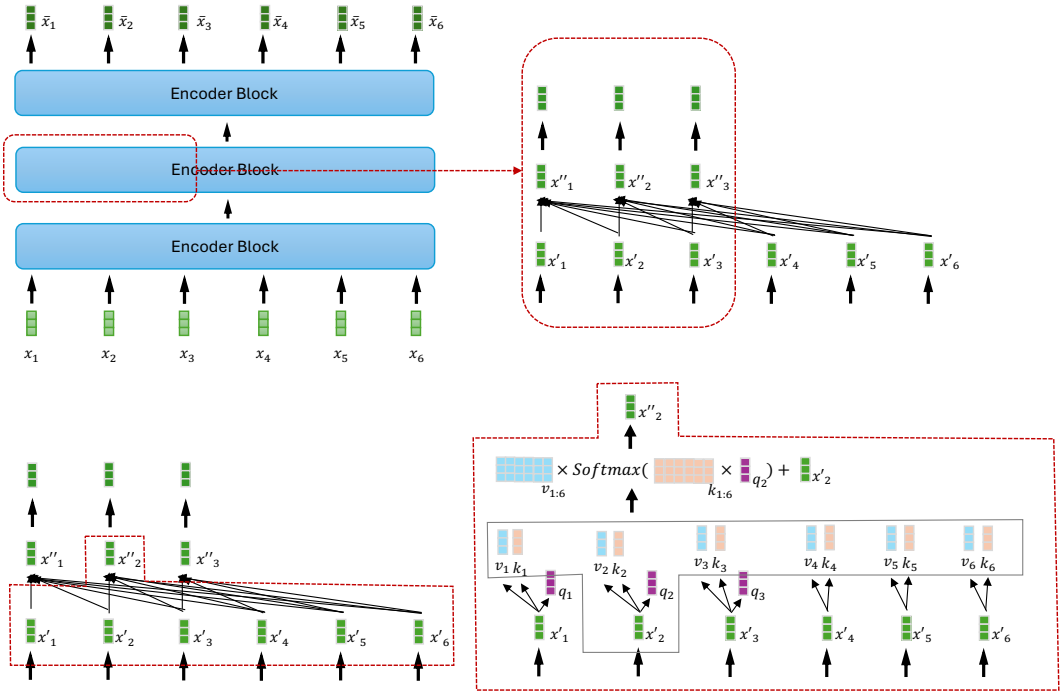

Figure 2.5.: Attention Mechanism

and generally uniform, acoustic signals are continuous, have long-range dependencies, and can vary greatly in length. Therefore, a lot of work has been done on the speech transformer. The Deep Transformer Networks for End-to-End Speech Recognition are proposed by [119]. The paper uses very deep Transformer networks (up to 48 layers) with stochastic residual connections to improve generalizability and training efficiency for ASR. Based on this, [118] combined the relative position encoding[35] with the Deep Transformer[119] to better accommodate the variable distributions in speech data and achieve improved results on the MuST-C[38] speech translation benchmark. Considering the quadratic complexity of the Transformer's self-attention mechanism when dealing with long audio sequences, an adapted transformer for End-to-end Spoken Language Translation is proposed in[37] and reducing the memory complexity by 1) down-sampling of input with convolutional neural networks, 2) modeling of bidimensional nature of spectrograms, and 3) a distance penalty in the attention mechanism to focus more on local context. In [2], the weight of some attention is avoided for speech tasks, hence decreasing the size of the attention matrix. The transformer encodes the speech features, thereby introducing local self-attention with a suitable window size for each layer to reduce the computational complexity.

The gap between speech and text modalities challenges the E2E speech-to-text translation. In contrast to the traditional cascade pipeline, the end-to-end model necessitates a moderate amount of paired speech-to-text data, which is not easy to obtain. Different methods of existing E2E models have been proposed to mitigate the lack of ST parallel training data. Based on their training strategies, they can be classified into either 1)pre-training automatic speech recognition (ASR) and machine translation (MT) components

of the model, 2)employing multi-task learning frameworks that integrate various aspects of speech and language processing, and 3) generating synthetic data [9, 141].

ASR and MT component pretraining Low-resource speech-to-text Translation proposed by [11] explores methods for speech-to-text translation in scenarios where transcripts to train a speech recognizer are not available for low-resource languages, and [16] also focused on developing a method for translating spoken content in audiobooks directly without intermediate text transcription in the source language. However, these two methods that train directly without pretraining are limited in performance. Adapted from prior, [12] the authors proposed to use an encoder that was pre-trained on high-resource automatic speech recognition (ASR) data, which is robust in capturing linguistically meaningful features across languages and thus significantly improved in the ST task. The adapter tuning proposed by [83] explored using a pre-trained ASR model or an mBART pre-trained model [95] as backbones to perform the multilingual ST task, demonstrating the flexibility of adapters in leveraging different types of pre-trained systems. [158] presented an innovative approach to improving speech-to-text translation (ST) performance using a Curriculum pre-training method, Transcription Learning, including the first phase of pre-training the model learns to transcribe the speech into text, and the second phase of understanding the utterance (Frame-based Masked Language Model) and mapping words between languages (Frame-based Bilingual Lexicon Translation).

Multi-Task Learning Tandem Connectionist Encoding Network (TCEN) proposed by [157] combined a speech encoder and a text encoder in a tandem setup, followed by a target text decoder. The architecture allows the model to separately handle acoustic feature extraction (via the speech encoder) and linguistic feature extraction (via the text encoder). The TCEN was trained using a multi-task learning approach that involves simultaneous training on AST, MT, and ST tasks, which leverages the strengths of each individual task to improve overall model performance. Unlike the TDEN focusing on consistency across different training phases, [3] introduces a unique multitask learning approach where task-specific decoders are directly informed by other tasks' decoders, enhancing inter-task relationships, where the decoder of the second task (translation) receives inputs not only from the speech encoder but also from the decoder of the first task (transcription), which allows the translation task to leverage higher-level representations produced by the transcription task. The paper proposed by [32] integrated multitask learning involving speech recognition and translation by leveraging word embeddings as an intermediate semantic layer between the AST task and ST task, to enrich the semantic information transmitted, arguing that maintaining semantic integrity is more crucial than textual accuracy. The dual-decoder transformer [82] introduced a dual-decoder architecture within a Transformer model to handle both ASR and ST tasks simultaneously, which enhanced overall model flexibility and task interaction. To leverage text data for ST tasks, the paper [149] proposed a multi-task learning framework that integrates auxiliary tasks like MT and a denoising autoencoder alongside primary tasks of automatic speech recognition ASR and speech translation ST, where phoneme representation of text is used to align text data more closely with speech data, improving the efficacy of the multitask

training. The ConST proposed by [172] used a cross-modal contrastive learning method to bridge the representation gap between speech and text modalities by learning similar representations for semantically similar speech and text. All these works contributed to the field by leveraging multitask learning, enhancing semantic understanding, and aligning different data modalities to improve overall speech translation performance.

Generating Synthetic Data The Leveraging Weakly Supervised Data method proposed by [67] discusses an innovative method to enhance speech-to-text (ST) translation quality by utilizing synthetic training data. This paper proposed two ways of synthetic training data: 1) Using Text-to-Speech (TTS) systems to generate synthetic speech from the text; and 2) Employing Machine Translation (MT) models to generate synthetic translated text from the transcribed outputs of ASR systems. In this way, the model can train on a broader range of examples without the need for a corresponding increase in manually labeled ST datasets.

2.2.5. Toolkits for Speech Translation

To support the development and training of speech translation (ST) models, several researchers have introduced various toolkits. These ST toolkits provide a framework for pre-processing datasets for ST tasks, as well as training, fine-tuning, and evaluating models.

The EspNet-ST toolkit[65], was created in response to the absence of a toolkit capable of handling the sub-tasks of ST ahead of it. EspNet-ST includes functionalities for ASR, LM, E2E-ST, Cascade-ST, MT, and text-to-speech (TTS), accompanied by practical examples. Additionally, it offers pre-trained transformer-based models trained on various datasets such as MuST-C[38], Libri-trans[74], Fisher[33] and CALL-HOME corpus[121].

FairSeq S2T[156] toolkit extends the original FairSeq framework by[112] to include all features of EspNet-ST[65]. It provides the Non-Autoregressive MT, Online ST, and Speech Pretraining. The toolkit also provides state-of-the-art ST models based on RNN, transformers, and conformers. It also has an in-built data loader for MuST-C[38], Librispeech[114], and CoVoST datasets[155].

NeurST[184] is a streamlined toolkit that does not depend on Kaldi[122]. It enhances computational efficiency through the use of mixed precision and accelerated linear algebra.

SLT.KIT[176] provides models for ASR, MT, and ST. It includes unique features such as CTC and attention-based ASR, ASR with punctuation recognition, and a neural MT system.

In Table 2.1, we present the performance scores of various ST models or toolkits we mentioned here.

2.3. Rare words Recognition and Translation

Rare word translation plays a crucial role in ST, significantly impacting the overall quality and accuracy of the translation output. Our focus on rare word translation in ST involves translating rare words directly from spoken language to text. This process goes beyond merely detecting or recognizing rare words and also beyond translating rare words found

Models/Toolkits	Dataset	Language Pair	Speech(h)	Metric(BLEU)
EspNet-ST[65]	LibriTrans	En-Fr	960	17.8
	MuST-C	En-De	271	25.05
FairSeq S2T[156]	MuST-C	En-Xx	452	23.39
	Librispeech	En-Fr	960	9.0
	CoVoST-2	Xx-En	427	21.24
NeurST[184]	Libri-trans	En-Fr	960	18.7
	MuST-C	En-Xx	452	24.9
SLT.KIT[176]	IWSLT	En-De	80	14.08

Table 2.1.: Dataset statistics. Performance scores of the toolkits in the Seq2Seq framework with the datasets, language pairs, duration of speech, and metric(BLEU).

in text transcripts in machine translation (MT) systems; it entails translating rare words present in spoken utterances. However, we can still draw valuable insights from the recognition of rare words in ASR systems and the translation of rare words that are contained in text inputs in MT systems. This section will review previous research and methodologies used for recognizing rare words in ASR systems, translating rare words from text inputs in MT systems, and exploring the current research landscape of rare word translation in ST systems.

2.3.1. Rare Words in ASR

Due to the difficulty of recognizing rare words solely through fine-tuning pre-trained speech models, there are many recent works on the methods of improving rare word recognition in ASR. Current approaches to tackle the problem mainly involve 1) *Post Processing with Language Model, for instance, user-dependent language models, LM rescoring*; 2) *Generating Synthetic Audio*; 3) *Context awareness and memory enhancement.*

Post Processing with Language Model The second pass rescoring in ASR was proposed in [111], which is to run the ASR decoding in two passes, where the second pass model is used to improve the initial outputs from first-pass models by rescoring with a stronger language model. This approach has been used in various ASR systems [80, 94]. In [170], the second-pass model incorporates the multi-task language model trained not just to predict the next word but also to predict intents and slots related to the words. While this kind of method can improve performance on tasks it was directly trained on, it may reduce the model's flexibility to adapt to new, unseen tasks or domains without retraining of LM. The work of [135] explores enhancing the second-pass component with the Listen, Attend, and Spell (LAS) model proposed by [25], whose decoder serves similarly as an LM. Furthermore, [129] also proposed to use the Neural language model (NLM) as a second pass component. The work [134] employed a Hybrid Autoregressive Transducer (HAT) that combines LM trained solely on text data with an ASR system for rare words and phrases recognition. The work of [161] introduced a method of integrating language models more effectively

during the training of E2E models with LM-aware Minimum Word Error Rate (MWER) training. The approach uses language models to generate hypotheses and compute the MWER loss. Another method proposed by [100] used a retrieval-augmented language model (PersonaLM) to improve ASR personalization and enhance the recognition of rare words and domain-specific terminology. During ASR decoding, the initial predictions made by the acoustic model are rescored using the probabilities provided by PersonaLM. While the retrieval augmented mechanism allows the system to adapt its predictions to other domains without requiring retraining of the entire model, its effectiveness is contingent upon the availability and quality of domain-specific n-gram frequencies in the external database.

Finetuning with Generated Synthetic Audio The paper [56] incorporated synthetic audio generated from text-to-speech systems(TTS) to create training examples that include Out-Of-Vocabulary(OOV) words. This helps in fine-tuning the ASR model to better recognize these words without the need for extensive real-world data that includes the OOV terms. Another method proposed in [126] used TTS to generate synthetic audio of rare words and employed techniques like L2 regularization and Elastic Weight Consolidation (EWC) to prevent catastrophic forgetting. The effectiveness of the method relies heavily on the quality and representativeness of the synthetic audio, which may not perfectly mimic real-world speech variations and complexities.

Context Awareness And Memory Enhancement [23] proposes an approach named Phoebe, which extends the contextualized Listen, Attend, and Spell (CLAS) model by injecting pronunciations obtained by the grapheme-to-phoneme (G2P) model into the context module via using bias phrases converted into fixed-length embeddings, which include both textual and phonetic forms, enhancing the model's ability to interpret rare and context-specific words. The term "bias phrases" here refers to specific words or phrases, such as named entities that are provided to the model to influence or "bias" its predictions. Besides that, the two-step memory-enhanced model (2MEM) proposed in [64] proposed to enhance the ASR system with a memory module in a combination of a memory-attention and memory-entry-attention mechanism that allows for instant integration and recognition of new words or phrases extracted from context without retraining the system. Further, the context-aware Confidence Estimation Model [124] used a multi-head attention mechanism to adjust the encoder features based on contextual clues from an external memory bank (Neural Associative Memory) that contains relevant phrases or words, enhancing the model's ability to integrate contextual information for rare word recognition. Another approach of Continuous Learning of New Words proposed in [63] used a combination of factorization-based model adaptation and a memory-enhanced model to bias the ASR system towards decoding new words learned from slide content in lecture talks, iteratively improving word recall. This kind of method is limited by its dependency on the quality and applicability of memory information, the complexity and computational demands are also increased due to the integration of the memory module.

2.3.2. Rare Words in MT

Neural Machine Translation (NMT) [30, 147, 151] has recently demonstrated impressive advancements over Statistical Machine Translation (SMT)[167, 73]. However, NMT systems continue to face significant challenges[75], in that addressing rare words is one of them. Due to NMT, which has tended to bias high-frequency words, low-frequency words have little chance of being considered in the inference process. To tackle this limitation, previous works have proposed various strategies for representing rare words to enhance the translation of low-frequency words. Typically, [99] improved NMT systems by substituting rare words with special symbols such as unk_1, unk_2, ..., unk_i in the sentence. They then used an aligned dictionary to establish mappings between an unk_i in the source sentence and an unk_j in the target sentence. However, this method introduces potential ambiguities in the sentence context, as observed by [138]. In addition, [139] suggests applying Byte Pair Encoding (BPE) [47] to NMT systems, which effectively reduces vocabulary size and enhances translation performance—a technique now commonly employed in most translation systems. This approach allows a rare word to be divided into sub-words, preserving the sentence context but potentially creating new rare sub-words. However, this segmentation process may complicate the identification of original rare words from their sub-words. The methods of machine translation of rare words can be classified into 3 categories: 1) *Constrained decoding*; 2) *Copy mechanism*; 3) *Retrieval augmented generation.*

Constrained Decoding To address the challenges of translating terminology, a variety of terminology constraints (TC) approaches have been developed. TC approach requires the model to translate the following pre-provided terminology pairs, and they have been broadly implemented in commercial translation systems [180]. There are primarily two TC methods: PlaceHolder (PH) and Code-Switch (CS). The PH method substitutes terminology terms in both the source and target texts with sequential labels (e.g., "T_1", "T_2"), and during inference, the model predicts these labels instead of the actual terms [34, 101]. However, a significant limitation of PH is that these labels strip away the original semantic content, leading to translations that lack coherence. Unlike PH methods, CS methods follow the standard model and generate term translations word by word by injecting target constraints directly into the source sequence [144]. However, the source constraints' semantics are still constrained due to the direct replacement of target constraints. In [39], a variant approach that retains the source constraints but uses a tag to distinguish them from the replacement marks was proposed. [1] further improved performance by masking the source constraints. [18] used target lemma to make the model learn morphology knowledge. [58] utilized finite-state machines (FSMs) and multi-stack decoding strategy for structured enforcement of terminology constraints while translating. [93] proposed to extract a bilingual dictionary in a non-supervised manner from the parallel corpora provided for training, which helped to ensure the translation model is well-acquainted with the rare and technical terms it needs to handle. A soft constraint method was proposed in [110], which inserted target terms in the source sentences during training. [116] presented two Transformer-based encoder-decoder models for improving the translation consistency of terminologies: 1) Terminology Self-selection Neural Machine Translation (TSSNMT) which incorporated the gating mechanism to dynamically determine the relevance of source sentence and the

terminology during the translation process; 2) ForceGen Transformer (ForceGen-T) which incorporated a force decoding mechanism along with a copy mechanism ensuring the generation of pre-defined terminology.

Copy Mechanism The pointer networks [153, 137] automatically copy rare words from the source sentence into the target sentence by integrating a copy probability to the output distribution with a copy coefficient learned during the training process. The pointer network provided an effective way for implementing the TC approach [144] and other variants of rare word translation in MT. For example, [117] proposed to improve rare word translation by integrating an external expert model to annotate source sentences with possible translations for rare word translations. The model learns to focus on and potentially copy annotated inputs using a pointer network and reinforcement learning during training. In inference, similar annotations guide the translation of new source sentences to replicate training success with rare words. The paper by [54] explored the approach to handle rare words, which extends the basic Pointer-Generator model by integrating bilingual lexicons with the neural translation model, which allows the model to directly incorporate translations from the dictionary into the MT process, which is particularly useful for translating rare words that may not be adequately represented in the training data. The paper [181] proposed incorporating bilingual dictionaries effectively using a model composed of Pointer, Disambiguator, and Copier modules (PDC model), enhancing the translation of rare and OOV words. Instead of a direct decision between copying from the input and generating from a vocabulary based on contextual probabilities in basic pointer networks, PDC used the pointer for identifying source words for which dictionary translations are applicable, disambiguator for Choosing the correct translation among potentially multiple dictionary entries based on the context, the copier for integrating the selected translation into the target text output.

Retrieval Augmented Generation Besides, the kNN-MT approach proposed in [71] augmented MT model with a k-nearest-neighbor (kNN) classifier that retrieves k candidate target tokens from a separate datastore at inference time. By switching out the datastore or modifying its contents, kNN-MT can adapt to different domains without retraining. Although the work by [92] showed that kNN-MT can effectively help rare word translation performance, kNN-MT hasn't been used in E2E ST.

2.3.3. Rare Words in ST

In the field of speech translation (ST), the translation of rare words poses a critical challenge. This challenge is particularly intensified in ST, as it inherently compounds the complexities associated with rare word processing observed in both automatic speech recognition (ASR) and machine translation (MT) systems. Existing strategies for rare word recognition in ASR, such as language model (LM) rescoring, fine-tuning with synthetic audio data, and context enhancement, primarily aim to improve the fluency of rare word transcription. However, these approaches are limited by their lack of direct mapping from audio to translation. Additionally, methods employed in MT for addressing rare word translation,

including constrained decoding and copy mechanisms, focus on enhancing translation adequacy and quality semantically. Nonetheless, these techniques are restricted by their modality, functioning only with text inputs.

Speech translation, which merges the capabilities of both ASR and MT, necessitates a tradeoff between the fluency targeted by ASR techniques and the adequacy addressed by MT strategies. This integration adds complexity to the translation of rare words in speech translation, rendering the direct application of existing previous rare word processing methods for ASR or MT unsuitable.

As regards previous works on rare word processing in ST, there are only limited research works. A benchmark NEuRoparl-ST was created by [49], tailored for evaluating ST systems, particularly their performance in translating named entities (NEs) and terminology. The author of [50] proposed a method to detect NEs from the audio using a specialized detection module that compares the encoded representations of utterances with those of NEs from a contextual dictionary of NEs known to likely appear in given contexts. However, this work only focuses on detecting rare words rather than directly teaching the model how to translate them. The authors of [48] found that the nationality of the person being referred to is a critical factor leading to inaccuracies in person name translation. They then proposed to implement multilingual models to enhance the system's ability to handle diverse pronunciations effectively. Nevertheless, this work focused on the model's capability to recognize and adapt to pronunciation diversity, not directly on rare word translation. [42] used KNN to construct a datastore from the in-domain text translation corpus. This datastore is used during inference to perform a similarity search. The final translation probability is an interpolation between the E2E-ST model's prediction and the kNN retrieved neighbors. This method leverages in-domain text translation data to adapt E2E-ST models to new domains without needing in-domain speech translation data.

In light of the above strategies, beyond merely enhancing detection capabilities or adapting to linguistic variations to indirectly improve translation outcomes, there is a significant demand for direct speech translation models that are excellent at translating rare words and are capable of adapting to new, unseen domains.

2.4. In-Context Learning

For enabling speech translation models to adapt to new, unseen words or rare words where contextual information is sparse, In-Context Learning (ICL) is a particularly effective method. In-context learning, which involves directing the model's predictions by presenting relevant examples as a demonstration context during the inference phase [40].

In this chapter, we will provide a detailed introduction to in-context learning and review related works about their role in enhancing machine translation (MT), automatic speech recognition (ASR), and speech translation (ST) systems. Moreover, we will investigate how these approaches can be applied to develop direct speech translation systems that excel in handling translations of rare words. Then, we will discuss the potentiality of applying in-context learning to build direct speech translation systems that are excellent in rare word translation.

In-context learning is a method where LLMs make predictions based solely on provided contexts that include a few examples, without the need for explicit retraining or parameter updates[40, 22]. ICL works in the following way: Firstly, A small set of examples relevant to the task is selected, and these examples are formulated in a consistent format and serve as a demonstration context. Secondly, the demonstration context is concatenated with a new query to form a complete prompt. This prompt is designed to guide the model in understanding the task it needs to perform based on the examples provided. Unlike traditional machine learning, which adjusts model parameters through training, ICL makes predictions based solely on the context provided, without additional training or updating of model parameters. This process allows LLMs to adapt quickly to new tasks with minimal data in a computationally efficient way.

The performance of ICL relies on two stages: (1) the warmup training stage that cultivates the ICL ability of LLMs, and (2) the inference stage where LLMs predict according to task-specific demonstrations.

The training stage is to train the LLMs before ICL inference. The key idea is to bridge the gap between pretraining and downstream ICL formats by introducing objectives close to in-context learning. Since LLMs have shown promising ICL capability[22], existing works taking well-trained LLMs as back-bones exhibit ICL ability, even if not specifically trained. However, many studies also show that the ICL capability can be further improved through a continual training stage between pretraining and ICL inference, which we call model warmup for short. Previous training approaches can be classified into two categories: 1) Supervised In-context Training, where LLM is continually trained on a broad range of tasks with demonstration examples, which boosts its few-shot abilities[103, 150, 163, 164]; and 2) Self-supervised In-context Training, which leverages raw corpora for training[28, 53]. As for the inference stage, many studies have highlighted that the effectiveness of ICL is heavily influenced by the demonstration surface [40, 104], including the format and the order of demonstration examples, and so on. Given the critical role that demonstrations play in ICL, this section will delve deeper into exploring demonstration design strategies that are specifically tailored for demonstration strategies for 1)*ASR*, 2)*MT*, and 3)*ST* tasks.

ASR task The SALM model by[29] proposed a keyword boosting demonstration approach by providing a list of keywords to the model in the format of optional text context as a prompt in the inference stage, for example, 'Following word may occur in audio:['x','y','z',...]', x,y,z are keywords. The keywords list was built by selecting words and phrases with high occurrences in the test set and low recognition accuracy. The keyword boosting aims to bias the model to recognize particular words of interest without any backpropagation. The demonstration design in [61] utilized in both the training and inference phases includes speech utterances paired with their corresponding labels. A special separation token ('$\langle s \rangle$') is incorporated to segment various components of the input sequence, aiding the model in distinguishing between utterances and labels. This arrangement of input data ensures that the model comprehends and processes the demonstrations correctly, facilitating the model's ability to predict the label of a target utterance based on the provided examples. [89] proposed to integrate LLM into the ASR system as second-pass rescoring by demonstrating domain-specific prompts that encapsulate the

key elements or topics of the speech in the inference stage, enhancing its ability to adapt to the domain-specific vocabulary and style.

MT task There are many works working on using Large Language Models (LLMs) for machine translation [59, 68, 169, 188]. For example, [105] explored improving Machine Translation (MT) accuracy and adherence to domain-specific terminology by leveraging the in-context learning capabilities of LLMs. Their method involves feeding LLMs with prompts containing previously approved translation pairs or predefined terms at inference time. LLMs were also used to generate synthetic bilingual data that includes rare or domain-specific terms in [106] for finetuning the model to learn and predict rare terms more effectively during translation tasks. The terminology-aware translation approach was proposed in [19], which used LLM to refine translations by providing it with terminology constraints through curated prompts.

When selecting relevant examples to include in a prompt, the connection between the examples and the input sentences plays a crucial role in influencing the quality of translation. Various methods have been used to determine this relevance, including (a) n-gram word overlap between input sentences and examples(Agrawal et al., 2022). (b) embedding similarity[177, 59, 152] using LaBSE or RoBERTa[96]. Moreover, the quality of the examples is also an important factor. To ensure quality, examples are either selected from a known high-quality pool[152] or based on Language-agnostic BERT Sentence Embedding (LaBSE)[46] or COMET-QE scores[132] between the pairs[59, 89]. The work of [142] shows that the coherency of prompt examples with respect to the test sentence is an important factor for translation performance.

[79] introduced CTQ scorer, which is a method of regression-based scoring function designed to select in-context examples based on multiple features, such as similarity metrics, translation quality metrics, and other lexical or syntactic features and so on, to enhance the quality of translation. The typical way of demonstration examples was formatted as "[source] sentence: [X_1]; [target] sentence: [Y_1]" [X_1] and [Y_1] represent the source and target sentences from the selected in-context examples [79, 89, 152].

ST task Given the translation capabilities of Large Language Models (LLMs)[59, 68, 169, 188], and the speech recognition proficiency of Speech Foundation Models (SFMs)[13, 123, 127], there were many researches working on accomplishing ST tasks by integrating these technologies[29, 31, 45, 179]. LST, a large multimodal model designed for E2E-ST proposed in [179], consists of a speech frontend, an adapter, and an LLM backend. Through two-stage training for modality adjustment and downstream task fine-tuning, respectively, the LST achieves state-of-the-art BLEU scores on the MuST-C benchmark. [31] proposed a multi-task training framework Qwen-Audio to handle various audio types and over 30 tasks, including ST. The AudioChatLlama proposed by [45] extends the instruction-tuned Llama-2 model to handle E2E speech processing while maintaining the LLM capabilities by integrating Llama model with a small conformer audio encoder followed by a projection layer to match Llama-2-chat dimensions. Speech-Augmented Language Model(SALM) proposed in [29] integrates a frozen text LLM with a fast conformer speech model using modality adapter modules and Low-Rank Adaptation (LoRA) layers(Hu et al., 2021) to

handle speech inputs for tasks like ASR and AST. SALM utilizes keyword boosting and supervised in-context training to enhance speech-to-text tasks. This architecture for ST, in general combining SFM and LLM, has been formalized in research by [51]. This architecture consists of five components: SFM, Length Adapter (LA), Modality Adapter (MA), LLM, and Prompt-Speech Mixer (PSMix). The SFM is responsible for deriving semantic representations from the audio using a transformer or conformer encoder. The LA helps compress the timeline of audio embeddings, while the MA, a relatively smaller trained network, transforms these embeddings to be compatible with the LLM to bridge the gap between the speech input modality and text modality. The PSMix integrates these speech representations with textual prompts, which are then processed by the LLM to produce the final textual translation.

As for concatenating speech representation with textual embedding, there are mainly three concatenation solutions: 1)prepending the speech representation to the prompt embeddings[29, 31, 62, 113, 148]; 2)appending it to the prompt embeddings [160, 166]; 3) interleaving the speech representation with a prompt prefix and suffix [45]. Only one work [179] completely omits the prompt and directly feeds the LLM with the speech representations.

2.5. Retrieval

As we explore demonstrating examples to the ST model to facilitate its adaptation to open domains, the ability to retrieve relevant examples from other databases and the quality of these examples is critical to this task[79]. In this scenario, the retrieval system plays a crucial role in identifying and retrieving relevant examples from external corpus. In this section, we begin by discussing prior research on text retrieval approaches. Given our focus on spoken language, we will then delve into the most recent studies concerning the retrieval of spoken queries, and further, we will introduce recent works on retrieval-augmented speech/language models.

2.5.1. Text Retriever

The Retriever is usually regarded as an information retrieval system aiming to retrieve specific passages or text documents based on certain criteria from a larger corpus. It is especially important in open-domain text Question Answering(QA) systems where the system must find and extract information that probably contains correct answers from a vast, unstructured dataset to answer questions accurately[125], which reduces the search space for answer extraction and identifies the support context for users to verify the answer. It is also used in tasks such as summarization[173], where relevant sections of text are identified for further processing, and in machine translation, where context from similar texts can aid translation accuracy[71]. Broadly, current approaches to Retriever can be classified into three categories[187], i.e., 1)*Sparse Retriever*, 2)*Dense Retriever*, and 3)*Iterative Retriever*, which will be detailed in the following.

Sparse Retriever refers to the systems that search for the relevant documents by adopting classical IR methods such as TF-IDF[27, 76, 77, 85] and BM25[162, 171]. TF-IDF (Term Frequency-Inverse Document Frequency) is a numerical statistic that reflects how frequently a term occurs in a document. BM25[133] is an evolution of the TF-IDF concept, and it provides a ranking to estimate the relevance of documents to search queries based on the probabilistic model that is sensitive to term frequency and document length. Retrieval methods based on TF-IDF and BM25 use sparse representations to measure term matches. However, the effectiveness of the retriever may be impacted by the fact that the same term can have different semantic meanings, and different terms can share the same semantic meaning in both the questions and the documents [187].

Dense Retrievers are the systems that encode questions and documents into dense latent vector space where text semantics beyond term match can be measured. Compared with the limited ability to understand synonyms or semantically similar phrases of Sparse Retrieval, Dense Retrieval models show better performance in understanding and matching queries and documents based on their meaning rather than just exact word matches[187]. Along with the success of deep learning that offers remarkable semantic representation, various deep retrieval models have been developed in the past few years, greatly enhancing retrieval effectiveness. According to the different ways of encoding the question and document as well as of scoring their similarity, dense retrievers in existing OpenQA systems can be roughly divided into three types: (1)*Representation-based Retriever*, (2)*Interaction-based Retriever* and (3)*Representation-interaction Retriever*.

- The Representation-based Retriever, also called Dual-encoder or Two-tower retriever, employs two independent encoders like BERT [36] to encode the question and the document respectively, and estimates their relevance by computing a single similarity score between two representations[86, 70, 57, 140]. For example, ORQA [86] adopts two independent BERT-based encoders to encode a question and a document, respectively, and the relevance score between them is computed by the inner product of their vectors. In order to obtain a sufficiently powerful retriever, they pretrain the retriever using an Inverse Cloze Task (ICT), i.e., to predict its context given a sentence. The DPR[70] employs two independent BERT encoders like ORQA but avoids the necessity of the expensive pre-training stage. Instead, it focuses on training a strong retriever using pairwise questions and answers sorely. DPR strategically selects negative samples for a question, which include random documents from the corpus, the top documents identified by BM25 that lack the correct answer, and in-batch negatives, which are the correct documents associated with other questions within the same batch. Additionally, DPR demonstrates that the inner product function is optimal for calculating similarity scores in a dual-encoder retriever. However,[72, 97] pointed out that dual encoders have a limitation where the final relevance score between a query and a document is calculated through a straightforward dot-product of their embeddings. This simplicity can restrict their effectiveness, particularly in their ability to generalize across diverse domains. [107] challenged the notion and proposed to scale up the model capacity of the dual encoder using the T5 architecture [128] while keeping the bottleneck layer fixed to a

single dot-product operation and keeping the fundamental architecture unchanged. This study found that by scaling up the model size, dual encoders can overcome their inherent limitations and lead to significant improvements in the generalizability and data efficiency of the retrieval model.

- The Interaction-based Retriever takes a question together with a document at the same time as an input and is powerful by usually modeling the token-level interactions between them, such as a transformer-based encoder. For example, [108] They developed both a paragraph-level and a sentence-level dense Retriever, each based on BERT [36]. They view dense retrieval as a binary classification issue, where each question and document pair is input, and the embedding of the [CLS] token is used to determine their relevance. Interaction-based method is powerful as it allows for very rich interactions between question and document; however, it is computationally expensive.

- Representation-interaction Retriever: In order to achieve both high accuracy and efficiency, some recent systems [72, 183, 185] combine representation-based and interaction-based methods. For example, SPARTA [185] develops a neural ranker to calculate the token-level matching score using dot-product between a non-contextualized encoded (e.g., BERT word embedding) question and a contextualized encoded (e.g., BERT encoder) document. Concretely, given the representations of the question and document, the matching score between the question and passage is computed via dot product, max-pooling, ReLU, and log sequentially. The representation-interaction method is a promising approach to dense retrieval due to its good trade-off between effectiveness and accuracy.

Iterative Retriever aims to search for the relevant documents from a large collection in multiple steps given a question, which is also called Multi-step Retriever. It was explored especially when answering complex questions like those requiring multi-hop reasoning [6, 102, 182, 168].In order to obtain a sufficient amount of relevant documents, the search queries need to vary for different steps and be reformulated based on the context information in the previous step[165].

2.5.2. Speech Retriever

With our focus on end-to-end speech translation, we will explore retrieval systems that operate with spoken queries. We define a Spoken Query Retriever as a system capable of retrieving both spoken and textual content from extensive datasets based on queries in spoken form. This requires modifying traditional retrieval systems to comprehend and process queries delivered in spoken language.

The concept of Spoken Query Retrieval aligns closely with Spoken Content Retrieval. The latter specifically involves indexing and retrieving spoken content from large collections of spoken audio data, where queries may be submitted in either text or spoken formats [66]. Given the scarcity of research focused exclusively on spoken query retrieval,

we will primarily concentrate on spoken content retrieval to gain insights that may apply to both domains.

Considering the achievements and effectiveness of text-based retrieval systems as we discussed in the last section, a spoken query retrieval system could intuitively be implemented by integrating an Automatic Speech Recognition (ASR) module ahead of a text-based retriever[87]. This setup converts spoken queries into text, which is then processed by text retrieval methods. The cascade approach of retrieving passages from spoken archives has been extensively investigated and reported by [87, 81, 26, 143].

However, simply cascading ASR module ahead of text retriever will cause several challenges[174]: First, transforming speech signals into ASR transcriptions is inevitably associated with ASR errors; Secondly, previous work [84] shows that directly feeding the ASR output as the input for the following down-stream modules usually cause significant performance loss, for example, modules for speech translation; Moreover, additional information, such as audio recordings, contains potentially valuable information in spoken form. Last but not least, the rare words, such as terminologies, named entities, or out-of-vocabulary (OOV) words that we emphasize can also be hard to be recognized by ASR. Therefore, implementing spoken content retrieval in an end-to-end approach directly rather than over ASR transcriptions is highly desired. The end-to-end spoken content retrieval approaches have mainly two directions:

Query-by-example spoken-term detection [130, 4]: The Retrieval from query to the passage, where the query typically consists of a spoken short phrase and the gold passages must contain this phrase. The proposed end-to-end Query-by-Example Spoken Term Detection system by [130] consists of three interconnected modules: feature extraction, similarity matrix computation, and CNN-based matching. Initially, features are extracted from both the query and the test utterances. Subsequently, these features are used to compute a frame-level similarity matrix, which quantifies the similarity between the query and the passage over time. This matrix is then treated as an image and fed into a convolutional neural network (CNN). The CNN is tasked with classifying whether the test utterance contains the query, effectively determining the presence of the query within the test based on the patterns observed in the similarity matrix.

Semantic Search The Retrieval from query to passage, where the query is a spoken sentence that may not necessarily share overlapping content with the gold passage. The SpeechDPR (Speech Dense Passage Retriever) by [90] is proposed to tackle the challenge of untranscribed spoken passage semantic retrieval. SpeechDPR adopts the bi-encoder retriever framework and learns a sentence-level semantic representation space by distilling knowledge from the cascading model of unsupervised ASR (UASR)[7, 91] and text dense retriever (TDR). SpeechDPR assesses the similarity between the question and each passage in the spoken archive by calculating the inner product of their sentence representations and thus can find the most semantically relevant passage. However, Speech DPR is limited to cross-modality retrieval.

2.5.3. Retrieval-Augmented Recognition and Translation

Given the retrieval techniques discussed above, in this section, we will dive into previous works in retrieval augmented 1)*ASR*; 2)*MT*; and 3)*ST*.

Retrieval augmented ASR Persona LM proposed by [100] used a retrieval-augmented language model to improve ASR personalization. It constructs n-gram frequency matrices for retrieval and a Span Aggregated Group-Contrastive Neural (SCAN) retriever to rank external domains/users based on semantic similarity. The retrieved n-gram probabilities are used to augment the ASR's predictions with personalized context. In Retrieval Augmented SLM (ReSLM) by [159], a dual-encoder architecture where one encoder processes audio inputs to retrieve relevant text entities was used. Retrieved entities are prepended to the inputs of the language model, providing additional context for better recognizing rare entities in speech dialogs. The retriever uses one encoder to encode the query (audio of the user's utterance), and the other encodes the candidate (text of entity names) to two fixed-length embeddings. An average pooling was then applied on the encoder output (generally varied length) to map it to the fixed length embedding. The relevance of an audio/entity pair is scored by the cosine distance. [14] introduced k-PAT (k-nearest neighbors based Phone Augmented Transformer) for ASR slot error correction. [186] introduces a method called kNN-CTC that enhances CTC-based ASR systems using k-nearest neighbors (kNN) retrieval. The model is trained to produce Connectionist Temporal Classification (CTC) pseudo labels, which are used to create frame-level audio-text key-value pairs. During decoding, the intermediate representations of the input are used as queries to retrieve the k-nearest neighbors from the database. A probability distribution is computed over these neighbors, which is then interpolated with the original CTC output to improve recognition accuracy. [24] proposes a method to adapt ASR models using external text-to-speech (TTS) generated key-value stores. An external knowledge store is created by mapping TTS-generated audio representations to semantic text embeddings. The ASR model incorporates a k-nearest neighbors (kNN) based attentive fusion step during fine-tuning or training. This mechanism biases the ASR model using the retrieved key-value pairs from the external knowledge store, improving the model's adaptability to new domains. The approach is designed to handle large catalogs of specialized words or phrases, enabling efficient domain adaptation and reducing the need for extensive fine-tuning. The method demonstrates improved performance in ASR tasks, particularly in zero and few-shot scenarios, by leveraging external text data catalogs for contextual biasing.

Retrieval Augmented MT kNN-MT approach proposed in [71] augmented MT model with a k-nearest-neighbor (kNN) classifier that retrieves k candidate target tokens from a separate datastore at inference time. The kNN distribution is computed by aggregating the distances of the retrieved neighbors, which is then interpolated with the base model's output distribution to generate the final translation. Although the work by[92] (D. Liu et al., 2023) showed that kNN-MT can effectively help rare word translation performance, kNN-MT hasn't been used in E2E ST.

Retrieval Augmented ST Decoupled Non-Parametric Knowledge Distillation for End-to-End Speech Translation (DNKD) proposed in [178] utilizes retrieval to augment speech translation through knowledge distillation. A datastore containing key-value pairs, with key as high-dimensional representations of the translation context and value as corresponding ground truth tokens, is first constructed. During training, for a given translation context, k-nearest neighbors are retrieved from the datastore. The retrieved neighbors extend the training data by providing multiple target tokens for each context. The teacher distribution is constructed from retrieved examples to provide soft targets that guide the model toward better translation performance. [42] used KNN to construct a datastore from the in-domain text translation corpus. This datastore is used during inference to perform a similarity search. The final translation probability is an interpolation between the E2E-ST model's prediction and the kNN retrieved neighbors. This method leverages in-domain text translation data to adapt E2E-ST models to new domains without needing in-domain speech translation data

3. Method

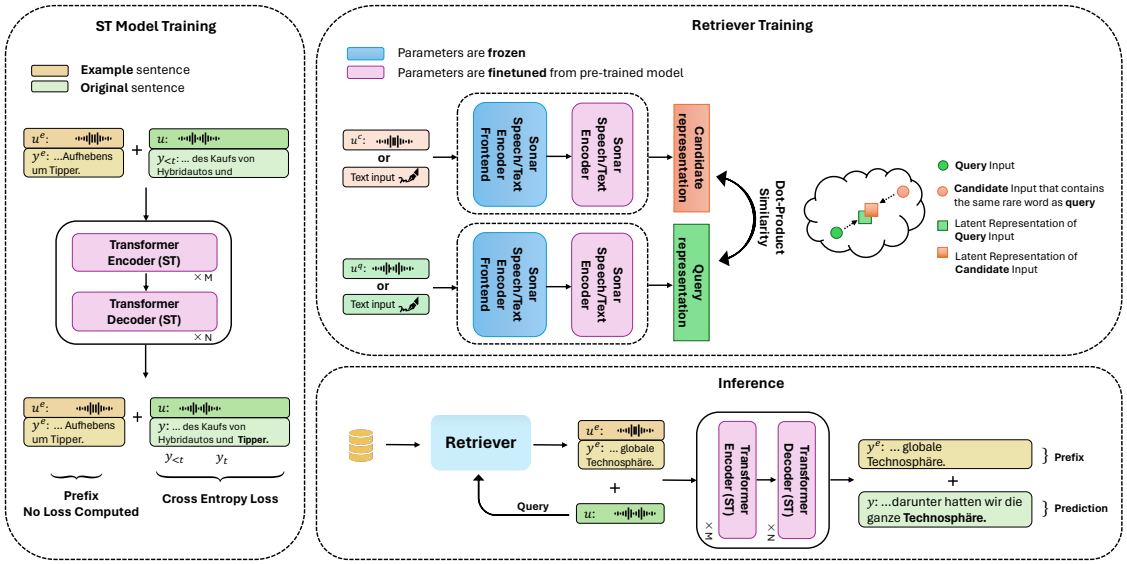

Figure 3.1.: Proposed retrieval-and-demonstration framework: At the ST model training stage (§3.1), example-prepended training data is used to instill in-context learning abilities in the S2T model. At the retriever training stage (§3.2), SONAR encoders are fine-tuned within the DPR architecture for our rare word task. At the inference stage (§3.3), retrieved examples are used as demonstrations to facilitate the translation of rare words.

Our retrieval-and-demonstration framework is illustrated in Figure 3.1. First at the left side of Figure 3.1, a trained direct ST model is finetuned to ingest examples (§3.1), which serve as demonstrations of correctly translating the rare words in question. Next, as shown at the upper right of Figure 3.1, we trained a multi-modal retriever (§3.2) to retrieve relevant examples for demonstration. During inference, as shown at the lower right of Figure 3.1, given an utterance containing rare words, we retrieve a relevant utterance and its translation with our trained retriever and demonstrate them to our finetuned ST model to guide its inference (§3.3).

3.1. Adapting ST Models to Ingest Example

3.1.1. Motivation

The example translation, which is known as *translation memory* [20], is widely leveraged by human translators for domain-specific translations with terminologies [21]. Similarly, we aim to apply it to direct ST models to enhance its rare word translation ability. The underlying idea mirrors that of in-context learning (ICL) [22], where providing models with task-specific examples during inference improves the quality of the generated output.

While ICL has been primarily observed on text-based LLMs [22, 104, 152], we here first explored whether small- or medium-sized encoder-decoder-based speech translation models can also be instilled to exhibit this capability.

3.1.2. Training

The task of adapting standard ST models to ingest examples can be defined as follows: Given u the utterance to translation, let $\hat{y}$ be the target translation and y of predicted translation. Let (u^e, y^e) be an example utterance-translation pair retrieved from the retriever model. We finetune the ST model so that the model maximizes the probability of generating the correct translation $\hat{y}$, given input utterance u and example (u^e, y^e), as shown in Equation 3.1:

$$y = \arg\max_{\hat{y}} P(\hat{y}|u^e, y^e, u) \tag{3.1}$$

The difference to the standard training is that the example (u^e, y^e) is included as context when generating the target translation. For the training data, for the i-th training utterance u_i, an example utterance u_i^e is prepended to it, forming a concatenated input $u_i^e + u_i$.[1] The targets are also concatenated as y_i^e + <SEP> + y_i, where <SEP> is a special token indicating the separator between sentences.

During the training process, the loss is only calculated on y_i to prioritize the translation of the utterance after the example. Including the loss of the prefix leads the finetuning step to end prematurely in preliminary experiments. Therefore, the translation of the example sentence is used as a prefix and masked during loss calculation. The cross-entropy loss function we use for training can be expressed as Equation 3.2:

$$\mathcal{L} = -\sum_{t=1}^{T} M_t log P(y_t|y_{<t}, u^e, y^e, u) \tag{3.2}$$

With M_t as a mask function Equation 3.3:

$$M_t = \begin{cases} 0 & \text{if position } t \text{ is part of } y^e \\ 1 & \text{if position } t \text{ is part of } y \end{cases} \tag{3.3}$$

In doing so, we encourage the model to predict its outputs based on the context provided by the demonstration example.

[1] Details on constructing the dataset is in §4.1.

3.2. Example Retrieval

3.2.1. Formalization and Challenge

Given a query utterance u containing a rare word w: $u = (w_1, w_2, \ldots, w_n)$ and $w \in \{w_1, w_2, \ldots, w_n\}$. We aim to retrieve a relevant example (u^e, y^e) from an example pool $\mathcal{D} = \{(u^1, y^1), \ldots, (u^m, y^m)\}$ with a retrieval model R, such that the rare word w is spoken in utterance u^e. Here u^i indicates the i-th utterance and y^i its translation, as shown in Equation 3.4.

$$(u_i^e, y_i^e) = R(u_i, D),\ where\, R(u_i, D) = \{(u_i^e, y_i^e) \in D \,|\, w \in u_i and w \in u_i^e\} \tag{3.4}$$

Note that here, we only have utterances to translate as queries. Therefore, speech-based retrieval is a must. Compared with text-based retrieval, we face additional complexities in speech-based retrieval as the query u is only in speech.

- First, speech is versatile. Unlike text, which often has a standard writing system, the speaking condition for the same content varies in every recording, which requires a robust retriever that accounts for pronunciation variations.
- Second, speech sequences are magnitudes longer than text. As we have to find the sentence with the same keywords as the sentence we are querying, the retriever must be able to find fine-grained local features corresponding to the keywords in long sequences.
- Third, we cannot implement the retrieval by directly cascading the ASR with a text-based retriever. Transcribing the query utterance first and then using text-based retrieval is suboptimal due to ASR errors, especially on rare words.

3.2.2. Architecture

To build the retriever, we take inspiration from the Dense Passage Retriever (DPR) architecture [70]. DPR is a prominent architecture in *text-to-text*-based information retrieval (IR) approaches.

We adopted the DPR architecture for our retrieval as the nature of our example retrieval task resembles IR, where relevant answers are retrieved given a question.

The DPR has a *dual-encoder* architecture, where one encoder encodes the questions, and the other encodes the passages potentially containing answers to the questions. The retrieval model is trained with a contrastive objective, mapping question-passage (positive) pairs closer to each other in the latent space while pushing irrelevant (negative) pairs further apart. During inference, passages closer to the encoded question by the dot-product similarity are returned as answers.

In our case, the questions are the utterances containing rare words to translate, and the passages are the example sentences we are searching for as our demonstration.

Therefore, the candidate passages containing the same rare words as the query utterances are considered positive pairs, while those not sharing the same rare words are negative pairs. The architecture of our retriever is shown in Figure 3.1.

3.2.3. Speech-to-Speech/Text Retrieval

As we have utterances u as queries and we could retrieve both speech and text, we propose to extend the DPR model to support querying speech/text from speech. We consider the following retrieval modalities:

- Speech→speech retrieval: we retrieve u^e in speech using audio query u.
- Speech→text retrieval: as the example utterance-translation pair (u^e, y^e) to be retrieved often also have text transcripts s^e available, we also consider retrieving y^e directly using audio query u, with s^e as candidates. This requires the retriever to support both modalities (text and speech).
- Naïve text→text retrieval: first transcribing the query utterance u and then text-to-text retrieval for y^e. As discussed before, the risk of ASR errors, especially in rare words, renders this approach suboptimal. The additional inference time for running ASR makes it further unpractical.
  However, here we are supposing we have oracle text transcript of both querying utterances and candidate utterances and do the retrieval with their transcripts.

Given these requirements, instead of initializing the dual encoders with pre-trained BERT [36] as in DPR [70], we leverage recent speech-text joint representation models including SONAR [44] and SpeechT5 [5], more details will be introduces in subsection 4.2.2. Note that our ST model and the retriever are two separate models, and we don't back-propagate the gradient of ST to the retriever.

3.3. Integrating Examples into ST Model

3.3.1. Inference with Retrieved Examples

As shown in Figure 3.1, during inference, the model is provided with a test input u and a retrieved example (u^e, y^e). The example is prepended to test input in the same way as in training. The example input-output pairs are integrated by forced decoding. After the separator token (<SEP>), the model starts to autoregressively generate the output translation, conditioned additionally by the example utterance and translations. The translation of the example y^e is taken as a prefix in the same way as in training.

3.3.2. Practical Considerations

An advantage of our framework is its modularity. The separation of the ST and retrieval modules enables straightforward upgrades to newer models in either component. This also means either our ST models or our retriever can be substituted by other modules to fulfill various needs.

Moreover, the retrieval module can be implemented using highly optimized toolkits like FAISS [69], which ensures efficient retrieval without compromising inference speed.

4. Experimental Setup

4.1. Dataset Construction

Split	# utt.	Avg. utt. duration (s)	Avg. #tokens	# unique rare words
train (original)	250942	6.5	27.1	9512
tst-COMMON	2580	5.8	25.3	157
rare-word pool	9821	9.7	43.1	8679
dev-rare-word	6932	9.9	42.8	6244
tst-rare-word	2500	9.9	43.1	2358
train-reduced	231689	6.2	25.8	3164

Table 4.1.: Dataset statistics. We split the original training set into the example pool with rare words (rare-word pool), dev/test sets for rare words (dev/tst-rare-word), and a reduced training set (train-reduced). The example pool simulates existing resources for querying.

we use the English-to-German subset of the MuST-C dataset [38] for training and evaluation, where the task is to translate from English-public speaking audio to German text. Instead of directly utilizing the MuST-C data, we reorganize its original training set into rare-word pool, dev-rare-word, tst-rare-word, and train-reduced, in order to create a targeted data condition for rare words. This is implemented by extracting sentences containing rare words from the original training set to create train-reduced and other dedicated sets. The statistics of the original dataset and the newly created splits are in Table 4.1. The rare-word sets have higher average token counts due to 1) longer utterance duration and 2) the rare words being segmented into finer-grained subwords. Note that we only re-split the training set, leaving the official validation and test sets (tst-COMMON) unmodified. Below, we describe the dataset construction process in detail.

Rare Word Sets Our data partition step is inspired by [109], which re-splits parallel data based on word frequencies. Specifically, from the English transcript, we find rare words by their *corpus-level frequency*, choosing those whose lemma appears two or three times in the original training set. For rare words occurring twice, we move their corresponding utterances to the rare-word pool and the joint dev/tst set, respectively, which creates a *zero-shot* condition where the rare word is never seen in training. For rare words occurring thrice, We followed the same strategy for two occurrences. The remaining third occurrence is retained in the reduced training set to create a *one-shot* learning scenario, where the rare word is seen once in the training set. We are first moving the rare word to the rare-word

Algorithm 1: Algorithm for Dataset Construction

Input: TrainSet=$\{u_1, u_2, ...\}$, RareWordList=$\{w_1, w_2, ...\}$
Output: TrainReduced, DevRareWord, TstRareWord, RareWordPool
//Initialization
$TrainReduced \leftarrow TrainSet$
$DevRareWord \leftarrow []$
$TstRareWord \leftarrow []$
$RareWordPool \leftarrow []$
$EncounteredRareWord \leftarrow \{\}$
for *each*, $u \in$ *TrainSet* **do**
    //Each Utterance u in TrainSet
    if *u contains word w* $\in$*RareWordList* **then**
        if *w not in EncounteredRareWord.keys()* **then**
            //Word w appears for the first time
            EncounteredRareWord[w].append(u)
            TrainReduced.remove(u)
            RareWordPool.append(u)
        end
        else if *EncounteredRareWord[w].size is 1* **then**
            //Word w appears for the second time
            EncounteredRareWord[w].append(u)
            TrainReduced.remove(u)
            TstRareWord.append(u)
        end
        else if *EncounteredRareWord[w].size is 2* **then**
            //Word w appears for the third time
            EncounteredRareWord[w].append(u)
        end
        else
            Continue
        end
    end
end

pool and then to the tst-rare-word (as shown in algorithm 1) to ensure that for each word in the tst-rare-word, there's a sentence that contains the same rare word in the rare-word pool. Moreover, we are making sure that one sentence only appears once among the train-reduced, tst-rare word, dev-rare word, and rare word pool. Finally, the aggregated dev/tst set is split into individual development and test sets for standard evaluation. The number of unique rare words in Table 4.1 represents those whose lemma only appears two/three times in the original training set, and each lemma is counted only once. The detailed algorithm is shown in algorithm 1.

We then analyze the rare word types in tst-rare-word by a named entity recognition (NER) model[1] with results in Table 4.2. More detailed categorization of words are shown in section A.1.

tst-rare-word	Person	Location	Tech	Food	Company
2358	130	72	29	27	25

Table 4.2.: NER results on rare words in tst-rare-word with the number of unique words in each category.

Training Data with Prepended Examples To adapt the ST model and to train the retriever, we need training data with prepended examples. As most utterances lack rare words by the previously used corpus-level frequency (3164 rare words in 231k utterances in Table 4.1), we propose to use *sentence-level* rare words to choose the prepended examples. Specifically, for each piece of the training data (u^i, s^i, y^i), we identify the word w_s in s^i that has the least corpus-level frequency among all words in its transcript. We then sample another training instance (u^j, s^j, y^j) from the training data, where s^j contains the same sentence-level rare word w_s as example. Note that w_s is not an actual rare word, w_s is just a word that appears less frequently than other words in the same sentence, which is what means the *sentence-level* rare word.

Test Set with Gold Examples We also construct a variant of tst-rare-word set with gold examples, where the rare word in the test utterance is always present in the example. This is finished by doing the word analysis on the transcript of sentences in both that test split and the rare-word pool. The test set with gold examples serves as an oracle condition for evaluating the ST model's ability to learn from perfect demonstrations. As our data splitting procedure ensures that the rare words also occur in the example pool, we select sentences from the rare-word pool containing the same rare words as those in the tst-rare-word set to serve as example sentences. The example sentences are then prepended to test sentences in a way identical to that in the training set with prepended examples.

[1]Huggingface model by [175]

4.2. Model Configuration

4.2.1. ST Model

We use the Transformer architecture S2T_TRANSFORMER_S in FAIRSEQ S2T [156] for all our ST models. To prevent the tokenizer from seeing the rare words during its training, which will cause an unfair test condition, we train the SentencePiece [78] tokenizer on the reduced train set after the utterances containing rare words are moved to dedicated splits (Table 4.1). Based on this vocabulary, we train the base model on the train-reduced set, closely following the hyperparameters from [156]. We then adapt the base model to ingest examples as described in §3.1 using the reduced training set with prepended examples (§4.1). As the prefix tokens do not contribute to the overall loss (Figure 3.1), we double the effective batch size to keep the loss scale comparable to before.

Training Details We use the Transformer architecture S2T_TRANSFORMER_S in FAIRSEQ S2T [156] For all our ST models, the encoder-decoder architecture consists of 12 transformer encoder blocks and 6 transformer decoder blocks, with a model dimension of 256 and an inner dimension (FFN) of 2,048.

We initialized the ST model from a pre-trained ASR model[2]. Subsequently, we fine-tuned the pre-trained model for the ST task with hyperparameters following [156], specifically, we set dropout rate 0.1 and label smoothing 0.1. The ST training used a tokenizer with a vocabulary size of 8,000. To prevent the tokenizer from seeing the rare words during its training, which will cause an unfair test condition, we train the SentencePiece [78] tokenizer on the reduced train set after the utterances containing rare words are moved to other splits as discussed in §4.1.

During the training of the adapted ST model with examples, we doubled the effective batch size to maintain a comparable loss scale since the prefix tokens do not contribute to the overall loss. Additionally, we set dropout rate to 0.2 after doing a search in {0.1, 0.2, 0.3} based on the dev loss during the training of the adapted ST model. The training was stopped after the validation performance did not improve for 30 consecutive epochs (patience 30). The detailed Hyper Parameters are listed in Table 4.4. For evaluation, we averaged the last 10 checkpoints.

Inference Details The inference uses a beam size of 5. Since the rare-word-tst dataset includes example-prepended sentences, the sentences are longer than typical translation sentences. To keep all utterances in the rare-word-tst set, we set a large allowed source size with –max-source-positions 30000. This ensures that even the longest utterances are not excluded from the rare-word-tst set.

4.2.2. Retriever

We use the DPR [70] architecture for the retriever. The encoders are initialized with either SONAR [44] or SpeechT5 [5]. For both models, we use the encoder only and discard

[2] `https://dl.fbaipublicfiles.com/fairseq/s2t/mustc_de_asr_transformer_s.pt`

	Hyper Parameters
Optimizer	Adam
Adam Eps	1e-8
Adam Betas	(0.9, 0.999)
Learning Rate Scheduler	Inverse Sqrt
Learning Rate	0.001
Dropout	0.2
Attention Dropout	0.2
Beam Size	5
Max Tokens	80,000
Clip Norm	10
Patience	30
Warmup Updates	10000

Table 4.3.: Adapted ST Model Training Hyperparameters.

the decoder. DPR requires fixed-size embeddings from its encoders. For SpeechT5, we mean-pool over the sequence length. For SONAR, we use the built-in attention-pooling for the speech encoder and mean-pooling for the text encoder. The dual encoders in DPR are trained on the reduced training set with prepended examples. Each sentence's example serves as a positive example, while examples from other sentences in the batch are in-batch negatives. Only the top layer of the encoders is trained, as the lower layers of the encoders are likely responsible for extracting low-level acoustic features. These features are considered less relevant for our retrieval task, which focuses on word-level information. Another reason is memory efficiency in training.

Training Details Our retriever is based on the DPR [70] architecture, where a dense passage encoder E_P and a question encoder E_Q is constructed to map candidate input c and query input q to latent representation vectors respectively. The similarity between the candidate representation and the query representation is defined as the dot-product of their vectors as shown in Equation 4.1:

$$sim(q, c) = E_Q(q)^T E_P(c) \tag{4.1}$$

The encoders E_P and E_Q of DPR are initialized with SpeechT5 encoder[5] or SONAR encoder [44].

Speech T5 The SpeechT5 speech/text encoder transforms speech or text input into a 768-dimensional embedding vector. It comprises 12 Transformer encoder blocks, each with a model dimension of 768 and an inner feed-forward network (FFN) dimension of 3,072. Before the encoder, a speech/text-encoder pre-net preprocesses the input. The speech-encoder pre-net includes the convolutional feature extractor of wav2vec [8] for waveform downsampling. The text-encoder pre-net applies positional encoding to convert character-level tokenized indices into embedding vectors.

SONAR The SONAR speech/text encoder encodes speech/text input to an embedding vector of 1,024. The encoder consists of 24 transformer encoder blocks with a model

dimension of 1,024 and an inner dimension (FFN) of 8,192. The speech encoder-frontend applies the wav2vec feature extractor [8], while the text encoder-frontend uses a position encoder.

Training The dual encoders in DPR are trained on a reduced training set with prepended examples. Each sentence's example works as a positive example, while examples from other sentences in the batch serve as in-batch negatives. We set a batch size of 4 and a learning rate of 2e-5 for training (Detailed in Table 4.4). Given the large size of the SONAR

	Hyper Parameters
Optimizer	Adam
Adam Eps	1e-8
Adam Betas	(0.9, 0.999)
Learning Rate	2e-5
Dropout	0.1
Weight Decay	0.01
Batch Size	4
Warmup Updates	1,234

Table 4.4.: SONAR-based Retriever Training Hyperparameters.

encoder, for memory efficiency, only the top layer of the SONAR encoder is trained. This approach is not only for memory efficiency but also because the lower layers likely extract low-level acoustic features, which are less relevant for our retrieval task focused on word-level information. We further investigate the retrieval accuracy under different numbers of trainable parameters. As shown in Figure 5.1. We use the settings with the best retrieval accuracy for our ST task. which are:

- For the speech-to-speech retriever, the top 2 layers of both speech encoders are trained, resulting in 205 million trainable parameters.
- For the speech-to-text retriever, the top 8 layers of both the text and speech encoders are trained, with 422 million trainable parameters.
- For the text-to-text retriever, the top 8 layers of both text encoders are trainable, totaling 335 million trainable parameters.

Inference Details During inference time, we apply the passage encoder E_P to all the candidates in the rare-word pool. Given a question q, we can derive its embedding $v_q = E_Q(q)$ and then retrieve the top-1 candidate whose embedding is the closest to v_q from the rare-word pool.

4.3. Evaluation

We evaluate speech translation quality with sacreBLEU [115][3] and COMET [131][4]. For the accuracy of rare word translation, we evaluate how many unique lemmatized rare words in the test set are translated. We use the spaCy toolkit [60] for word lemmatization and used AWESoME Aligner [41] for en-de word-level alignment. For rare word accuracy, we further distinguish between rare words appearing once or never appear in the training set (§4.1)(train-reduced set), which corresponds to the *one-shot* and *zero-shot* accuracy. For the retriever, we use top-1 retrieval accuracy to evaluate the retriever's performance. Only the top retrieved examples are used as demonstrations in the ST model.

4.3.1. SacreBLEU Score

The SacreBLEU score[120] is a metric for evaluating the quality of machine-translated text compared to a reference translation. SacreBLEU aims to provide a standardized and replicable BLEU score calculation. It eliminates the variations that can arise from different tokenization and preprocessing steps by using a fixed, consistent approach. SacreBLEU evaluates the overlap of n-grams (sequences of n words) between the candidate translation and one or more reference translations. It uses a fixed, consistent tokenization and preprocessing pipeline to ensure replicable results.

4.3.2. COMET Score

COMET (Crosslingual Optimized Metric for Evaluation of Translation)[131] is a modern metric designed to evaluate the quality of machine translation outputs. Unlike traditional metrics such as BLEU, which rely on n-gram overlap between the machine translation output and reference translations, COMET employs neural networks and large multilingual pre-trained models to assess translation quality.

COMET is trained using human judgment scores from various machine translation evaluation campaigns. This aligns the metric closely with human perceptions of translation quality. The neural network architecture enables COMET to consider the context of the entire sentence or document, rather than just isolated segments. This helps in better evaluating translations that require understanding of broader context. Compared with BLEU score, COMET is more closely aligned with human judgment, providing more reliable assessments of translation quality from a human perspective.

[3] sacreBLEU [120] signature:
nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.4.2

[4] with `Unbabel/wmt22-comet-da`; ×100 for readability. The COMET models take text transcripts as source.

5. Results and Analysis

Before presenting the results of our proposed framework, We confirm that our baseline model performs on par with those reported in the literature[1] with the results in Table 5.1. Below, we will introduce our experiment results and their analysis.

	BLEU
FAIRSEQ S2T [156]	22.7
Our baseline model	23.6

Table 5.1.: The performance of our baseline model on the tst-COMMON split of MuST-C is comparable to existing baselines. Both models have the identical architecture using S2T_TRANSFORMER_S.

5.1. Impact of Demonstration

5.1.1. Direct ST models can effectively learn from demonstration at inference time.

In this section, we will discuss the ST model's performance in ingest demonstrated examples. To independently analyze the ST model's ability to learn from the prepended examples, we first assume an oracle retrieval model by using gold examples, which always contain the rare words in question. The results of adapted model on gold examples are in row (2) of Table 5.2. Compared to the baseline in row (1), this model achieves substantially higher overall rare word translation accuracy (+17.6% abs.), with a larger gain in zero-shot (+18.8%) than one-shot accuracy (+15.3%). Nonetheless, this gain comes at the cost of overall translation quality (−0.2 BLEU, −2.3 COMET). A potential reason is that the prepended example sentences make the input sequences much longer and, therefore, create more difficulty for learning. Nonetheless, since rare words are often important named entities or terminologies, capturing them correctly is as crucial if not more than the overall translation quality scores. Overall, the results suggest that task-specific demonstrations provided at inference time can effectively enhance the rare word translation accuracy of direct ST models. Our results on the tst-COMMON set (shown in Table 5.3) also show the same trend. However, due to the small number of rare words in the tst-COMMON set (116 out of 2580 utterances), there's a relatively small discrepancy between the results on the tst-COMMON split and the tst-rare-word split.

[1] ST performance of FAIRSEQ S2T[156] toolkit

As a reference, we also implement our approach on the MT model, by using our reduced-train set with gold examples. Our MT model is trained fully identical according to the FAIRSEQ MT tooklit[2]. The results of the baseline model and adapted model on gold examples are in Table 5.4 and Table 5.5. The rare word accuracy in MT is, in general, higher than that in ST. Moreover, compared to the baseline, our adapted model has a higher overall translation accuracy(+13.1% abs.) on the tst-rare-word set.

5.1.2. Quality of the given demonstration matters.

However, in reality, demonstrations do not always contain the rare words to translate. How would the demonstration quality influence the translation? In contrast to the gold examples before, we now use random selected examples that do not contain rare words relevant to the sentence to be translated. The results are in row (3) of Table 5.2. This led to a decline in translation quality (−1.3 BLEU, −2.4 COMET) and rare word accuracy. These results indicate that irrelevant demonstrations are harmful.

ST Model	BLEU	COMET	Overall acc (%)	0-shot acc (%)	1-shot acc (%)
(1) baseline model (on train-reduced)	17.2	57.9	11.8	11.0	13.3
(2) adapted + gold example	17.0	55.6	**29.4**	**29.8**	**28.6**
(3) adapted + random example	15.7	53.2	8.8	8.4	9.7
(4) train on {train-reduced + rare-word pool} (more data)	**17.9**	**59.0**	15.5	14.7	17.2
Using retrieved examples					
(5) adapted + text (gold transcript)→text	15.2	54.4	20.1	19.6	**21.2**
(6) adapted + speech→text	15.3	54.0	18.8	18.2	20.2
(7) adapted + speech→speech	**16.2**	**55.3**	**20.3**	**20.3**	20.2

Table 5.2.: Speech Translation quality (BLEU↑, COMET↑) and rare word accuracy↑ (overall, 0- and 1-shot) of different models on the **tst-rare-word** split. The lower section uses retrieved examples from the retriever (§5.3).

5.1.3. Seeing rare words only in training does not sufficiently improve their translation accuracy.

Instead of retrieving data from the rare-word pool as demonstration, a simple alternative is to add these data in training. Here, we add the rare-word pool into the training set and train an identical model to the baseline. The results on the tst-rare-word split are in row (4) of Table 5.2. Overall, the rare word accuracy only sees a slight increase compared to row (1), with an absolute accuracy improvement of 3.7%, which is far less than using gold example sentences (+17.6% overall). The results on the tst-COMMON set(Table 5.3) also show the same trend, where simply adding rare word to the training can only increase the rare word translation slightly(+0.8%) compared to the translation with gold examples(+6%). This

[2]https://github.com/facebookresearch/fairseq/tree/main/examples/translation

ST Model	BLEU	COMET	Overall acc (%)	0-shot acc (%)	1-shot acc (%)
(1) baseline model (on train-reduced)	23.6	70.5	14.7	13.6	15.8
(2) adapted + gold example	21.8	64.5	**20.7**	**22.0**	**19.3**
(3) adapted + random example	20.8	61.9	12.9	11.9	14.0
(4) train on {train-reduced + rare-word pool} (more data)	**23.9**	**70.9**	15.5	8.5	22.8
Using retrieved examples					
(5) adapted + text (gold transcript)→text	21.6	64	**19.0**	**22.0**	15.8
(6) adapted + speech→text	21.4	64.1	14.7	13.6	15.8
(7) adapted + speech→speech	**21.8**	**64.9**	**19.0**	20.3	**17.5**

Table 5.3.: Speech Translation quality (BLEU↑, COMET↑) and rare word accuracy↑ (overall, 0- and 1-shot) of different models on the **tst-COMMON** split. The lower section uses retrieved examples from the retriever (§5.3).

MT Model	tst-rare-word			
	BLEU	Overall acc (%)	Overall acc (%)	Overall acc (%)
(1) baseline model (on train-reduced)	17.8	46.4	50.9	36.6
(2) adapted + gold example	17.0	**59.5**	**63.6**	**50.7**

Table 5.4.: Machine Translation quality (BLEU↑, COMET↑) and rare word accuracy↑ (overall, 0- and 1-shot) of different models on the **tst-rare-word** split.

indicates that training with rare words alone is insufficient for improving their translation accuracy. This is likely because of the limited training signal for rare words, as each appears only once or twice. Note that the translation quality scores under this data condition also improved, which is likely a result of the additional training data.

5.2. Retrieval Performance

Before integrating retrieved examples into the ST model, we analyze the retrieval performance alone with results in Table 5.6. To establish the upper bounds of retrieval performance, we first use the original DPR model for text-to-text retrieval with gold transcripts of the query utterances and examples. As shown in row (1) of Table 5.6, directly using the pretrained DPR for QA is not sufficient for our task of rare word retrieval, as we are retrieving based on keyword matching, not traditionally semantic matching. Therefore, we fine-tuned the pretrained model for our task by using reduced train-set and their corresponding examples as positive example.The results in row(2) of Table 5.6 shows that fine-tuning DPR's encoders on our task enables effective rare word retrieval in a text-to-text setting (55.8%).

MT Model	tst-COMMON			
	BLEU	Overall acc (%)	Overall acc (%)	Overall acc (%)
(1) baseline model (on train-reduced)	23.8	31.9	37.3	26.3
(2) adapted + gold example	**21.1**	**42.2**	**49.2**	**35.1**

Table 5.5.: Machine Translation quality (BLEU↑, COMET↑) and rare word accuracy↑ (overall, 0- and 1-shot) of different models on the **tst-COMMON** split.

Retrieval Model	T→T	S→T	S→S
(1) Orig. DPR w/ BERT (pretrained)	2.0	–	–
(2) Orig. DPR w/ BERT (finetuned)	**55.8**	–	–
(3) DPR w/ SpeechT5 (finetuned)	0.1	0.0	0.0
(4) DPR w/ SONAR (pretrained)	28.7	22.3	20.6
(5) DPR w/ SONAR (finetuned)	46.6	**33.3**	**41.3**

Table 5.6.: Top-1 retrieval accuracy (%) of different retrievers on 3 modalities of text-to-text (T→T), speech-to-text (S→T), and speech-to-speech (S→S) on the **tst-rare-word** split. T→T retrieval uses gold transcripts as query.

5.2.1. Encoder choice is crucial for successful retrieval.

We proceed by adapting the original DPR to retrieval from speech. Overall, we notice that the choice of the encoder heavily impacts the retrieval performance. With SONAR, using the pretrained encoders already achieves partial success in fulfilling the task, with 28.7% retrieved on text-to-text, 22.3% on speech-to-text, and 20.6 % on speech-to-speech, as shown in (row (4) in Table 5.6). With finetuning further improving the results (row (5)), which reaches 46.6% on text-to-text retrieval, 33.3% on speech-to-text retrieval and 41.3% on speech-to-speech retrieval. However, finetuning SpeechT5 proves insufficient for learning the task (row (3)).

We believe that the discrepancy primarily arises from the models' ability to aggregate information over the sentence length: SONAR is explicitly trained to aggregate it into fixed-size embeddings, while SpeechT5 lacks such a mechanism. Naïve mean-pooling over sequence length fails to create meaningful embeddings over long sequences like speech, as well as character-level text representations used in SpeechT5.

5.2.2. Speech→speech outperforms speech→text retrieval.

While we initially expected speech-to-speech retrieval to be more challenging than speech-to-text retrieval due to the high variability of speech, the finetuned retriever in (5) of Table 5.6 shows stronger performance on speech→speech retrieval than speech→text (41.3% vs. 33.3%). We suppose that the reason is the modality gap between text and speech, which makes it more challenging to bridge the two different types of data.

Retrieval Model	T→T	S→T	S→S
(1) Orig. DPR w/ BERT (pretrained)	2.0	–	–
(2) Orig. DPR w/ BERT (finetuned)	**55.8**	–	–
(3) DPR w/ SpeechT5 (finetuned)	0.1	0.0	0.0
(4) DPR w/ SONAR (pretrained)	28.7	22.3	20.6
(5) DPR w/ SONAR (finetuned)	46.6	**33.3**	**41.3**

Table 5.7.: Top-1 retrieval accuracy (%) of different retrievers on 3 modalities of text-to-text (T→T), speech-to-text (S→T), and speech-to-speech (S→S) on the **tst-COMMON** split. T→T retrieval uses gold transcripts as query.

5.3. ST Performance with Retrieved Examples

5.3.1. Correlation between retrieval accuracy and translation quality:

As the retriever based on finetuned SONAR showed the most promising retrieval results (Table 5.6), We use the examples retrieved from this model to guide the ST. The results are in rows (5), (6), and (7) of Table 5.2. When comparing the performance of the three retrieval modalities, retrieval accuracy does not always translate to improved overall translation quality or rare word accuracy. Although text-to-text retrieval using gold transcripts had the highest retrieval accuracy (Table 5.6), its integration into the ST model resulted in lower translation quality compared to speech-to-speech retrieval. Moreover, in practice, we still need an ASR model to derive the transcripts that likely contain errors, especially on rare words. This introduces additional limitations to the text-to-text retrieval approach. Overall, these results show that speech-speech retrieval is more effective than the other modalities in improving rare word translation accuracy. Despite the improvement in rare word translation accuracy, we also note the drop in translation quality compared to the baseline (row (7) vs. (1); −1.0 BLEU and −2.6 COMET). The results on the tst-COMMON set(Table 5.3) also show the same trend. We expect that increasing the robustness of the ST model to examples containing incorrect rare words, for instance by including such examples in training, could mitigate this negative impact.

5.3.2. Does speech→speech retrieval help by implicit speaker adaptation?

Speech-to-speech retrieval could be particularly effective in finding same-speaker utterances due to the access to acoustic information. This raises the hypothesis that if the prepended example originates from the same speaker as the utterance to be translated, translation quality could be improved by implicit speaker adaptation [136], where the model benefits from adapting to the specific speaker's voice characteristics. To test this, we analyze the proportion of retrieved sentences from the same speaker across different retrieval modalities. The results in Table 5.8 show similar percentages for all three scenarios on the tst-rare-word split, indicating that the gains by speech-to-speech retrieval do not stem from speaker adaptation.

DRP + SONAR finetuned	T→T	S→T	S→S
Examples from same speaker (%)	50.3	53.4	50.2

Table 5.8.: Proportion of retrieved examples from the same speaker as the utterance to be translated for the three retrieval modalities on tst-rare-word split.

5.4. Effects on Unseen Speakers

Now we push the approach further under the challenging scenario of unseen speakers, i.e., the example pool does not contain any utterance from the speaker of the test utterance. Specifically, during retrieval, we ignore utterances from the same speaker as the query utterance. As shown in Table 5.9, this harms retrieval accuracy substantially, losing 14.9% to 23.4% compared to Table 5.6 for the three modalities. This is mainly due to the limited coverage of the rare-word pool, which contains only one sentence for most rare words. Excluding the speaker also excludes the rare word. However, the BLEU scores and overall rare word translation accuracy change only slightly compared to Table 5.2: T→T (−0.6 BLEU, −1.5%), S→T (−0.3 BLEU, −3.2%), S→S (+0.2 BLEU, −1.0%). This demonstrates that our approach, especially when using speech→speech retrieval, is relatively robust to unseen speakers.

Retrieval modality	Retrieval acc (%)	BLEU	Overall acc (%)	0-shot acc (%)	1-shot acc (%)
(5) T→T	23.2	14.6	18.6	18.5	18.7
(6) S→T	18.4	15.0	15.6	15.6	15.7
(7) S→S	**23.5**	**16.4**	**19.3**	**18.8**	**20.2**

Table 5.9.: Retrieval and ST performance on **unseen speakers**. Compared to Table 5.2, S→S retrieval has the least decrease in translation quality and rare word accuracy.

5.5. Analyses of Retrieval Performance

In our main experiments, we partially finetuned the DPR encoders. We now investigate the impact of different numbers of trainable parameters in the retriever. As shown in Figure 5.1, the retrieval performance of the SONAR-based retriever is stable across 100 to 500M trainable parameters out of a total of over 1.3B parameters. This indicates that the retriever can maintain nearly consistent performance despite changes in model capacity. In Table 5.11, we also show the top-5 retrieval performance on the tst-rare-word set under different numbers of trainable parameters. We are using a model with the highest retrieval accuracy of each scenario to do the retrieval during ST inference, that is, text-to-text(335M), speech-to-text (422M), and speech-to-speech(205M).

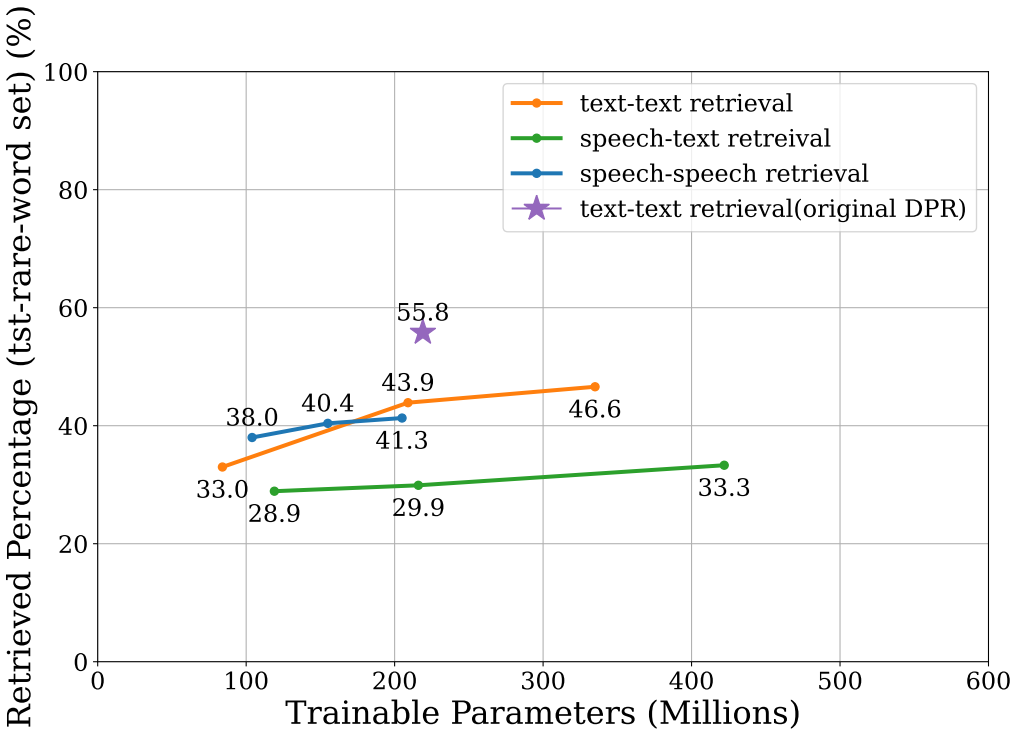

Figure 5.1.: Retrieval performance of the SONAR-based retriever for different numbers of trainable parameters.

5.6. Potential of Using More Examples

DPR + SONAR ft.	T→T	S→T	S→S
Top 1	46.6	33.3	41.3
Top 3	56.1	43.3	52.8
Top 5	60.4	48.0	56.2

Table 5.10.: Top-5 retrieval performance (%) of the SONAR-based retriever on the tst-rare-word set.

Few-shot learning is more often performant than one-shot learning because it provides the model with a broader context and more varied examples. However, as shown in Table 5.10, the increase in retrieval accuracy with top-5 examples is still not substantial compared to the top-1 result. Including multiple examples also makes input sequences significantly longer, especially as audio inputs are factors longer than text. This not only poses a challenge for the model but would also significantly slow down the inference speed, which we aim to avoid. For these reasons, we do not further explore the potential of using more examples.

DPR + SONAR ft.	Text→Text			Speech→Text			Speech→Speech		
# Trainable Params (Millions)	84	209	**335**	119	216	**422**	104	155	**205**
Top 1	33.0	43.9	**46.6**	28.9	29.9	**33.3**	38.0	40.4	**41.3**
Top 3	38.1	52.0	**56.1**	38.1	39.8	**43.3**	47.6	51.3	**52.8**
Top 5	40.8	55.3	**60.4**	42.1	43.9	**48.0**	51.4	54.8	**56.2**

Table 5.11.: Top-5 retrieval performance (%) of the SONAR-based text-to-text, speech-to-text, and speech-to-speech retriever on the tst-rare-word set under various number of trainable parameters.

5.7. Qualitative Example

Table 5.12 shows examples of our retrieval-and-demonstration approach on the translation of rare words. It includes examples of partially correct(top one of Table 5.12), fully correct(middle one of Table 5.12), and false translations(bottom one of Table 5.12) across different methods. For each source sentence, translations produced by different approaches are shown: the baseline model, the model trained with a rare-word pool, the speech-to-speech retriever retrieved example, and the adapted model with the retrieved example. The target translation is also provided for reference.

In the first example, the source sentence contains the rare words "Patrice and Patee." The baseline model fails to translate these names accurately, resulting in a false translation. Adding a rare-word pool improves the translation slightly, but errors remain. The adapted model with speech-to-speech retrieved example is much better but delivers a partially correct translation by capturing the name but adding a small part of unnecessary context "tee".

The second example involves the rare word "Murali Krishna." Here, the baseline model and the model adding the rare-word pool fail to produce accurate translations. Our adapted ST model achieves a fully correct translation, accurately reflecting the source sentence.

In the third example, the rare word "McLaren" is used. The baseline model and the rare-word pool model generate incorrect translations, misinterpreting the term. The speech-to-speech retriever retrieved an irrelevant example that doesn't contain the rare word to translate. Because of this, our adapted ST model fails to provide a correct translation of the rare word.

Source (transcript):	Patrice and Patee set out most days to go out hunting in the forest around their homes.
Baseline (Table 5.2 row (1)):	Die Bäume und Petes (Trees and Petes) setzten die meisten Tage hinaus, um in den Wäldern um ihre Häuser zu pumpen.
Adding rare-word pool to training (Table 5.2 row (4)):	Patrizinpathie (Patrizinpathie) setzte sich in den meisten Tagen um die Jagd in den Wäldern um ihre Häuser.
Speech→speech example (Table 5.6 row (5)):	Sie heißen Patrice und Patee (Their names are Patrice and Patee.).
Adapted ST + speech→speech (Table 5.2 row (7)):	Patrice und Pateetee setzten die meisten Tage, um in den Wäldern um ihre Häuser herum jagen zu können.
Target:	Patrice und Patee (Patrice and Patee) gehen fast jeden Tag jagen in dem Wald rundum ihr Heim.
Source (transcript):	Murali Krishna (Murali Krishna) comes from one of those villages.
Baseline (Table 5.2 row (1)):	Moralische Christen (Moral Christians) sind aus einem dieser Dörfer.
Adding rare-word pool to training (Table 5.2 row (4)):	Das Marate Krishna (Marate Krishna) kommt aus einem dieser Dörfer.
Speech→speech example (Table 5.6 row (5)):	Sie arbeitet mit Leuten wie Murali Krishna. (She works with people like Murali Krishna.).
Adapted ST + speech→speech (Table 5.2 row (7)):	Murali Krishna (Murali Krishna) kommt aus einem dieser Dörfer.
Target:	Murali Krishna (Murali Krishna) kommt aus einer dieser Dörfer.
Source (transcript):	The McLaren (McLaren) just popped off and scratched the side panel.
Baseline (Table 5.2 row (1)):	Und der Klient (client) stoppte ab und kratzte die Seite des Paddels.
Adding rare-word pool to training (Table 5.2 row (4)):	Und der Spieler (player) stürzte einfach ab und kratzte auf den Bürgersteig.
Speech→speech example (Table 5.6 row (5)):	Aber als Nebeneffekt sammelt er Kornette. (But as a sideline, he happens to collect cornets.)
Adapted ST + speech→speech (Table 5.2 row (7)):	Als der Klairner (Klairner) gerade ankam, stopfte er ein Nebenpandel.
Target:	Der McLaren (McLaren) bekam eine Beule und einen Kratzer an der Seitenkarosserie.

Table 5.12.: Examples of our retrieval-and-demonstration approach on the translation of rare words.

6. Conclusion

This thesis introduced a retrieval-and-demonstration approach to improve rare word translation accuracy in direct ST. For real-world applications, e.g., translating scientific talks, we recommend adding utterances from the same topic to the example pool and using speech-to-speech retrieval to identify examples. When feasible, one should consider incorporating an additional verification step to ensure the relevance of the retrieved sentences, by human-in-the-loop or automated techniques.

6.1. Answers to Research Questions

Based on the proposed approaches and experiment results, we address the research questions formulated in section 6.1 as follow.

Research Question 1: In what ways can the demonstration of sentences containing specific rare words from an external dataset improve the accuracy of rare word translation?

To address this, we adapted the ST model with prepended training data to instill its in-context learning ability, allowing it to ingest rare word information from the prepended examples. During the training of the ST model, each training utterance u was prepended with an example sentence u^e containing the same rare word as the demonstration. The example's target y^e is also prepended with predicted translation y as a prefix. No loss was computed for the prefix during fine-tuning to avoid the premature of the fine-tuning process. Our experiment on inferencing with gold examples(examples that always contain the rare words in question) prepended tst-rare-word split shows that the standard direct ST models can be easily adapted to benefit from prepended examples for rare word translation, in a way similar to in-context learning (§5.1). This improves rare word translation accuracy over the baseline by 17.6% with gold examples and 8.5% with retrieved examples.

Research Question 2: What methodologies can be developed to systematically extract sentences from an external dataset that share rare words with the sentence targeted for translation, thereby serving as a demonstrative example?

We developed a retriever inspired by the Dense Passage Retriever (DPR) [70] architecture. Instead of the BERT-based encoder used in DPR for text-to-text retrieval, we employ speech-text joint representation encoders such as SONAR[44] and SpeechT5[5] to facilitate not only text-to-text retrieval but also speech-to-speech and speech-to-text retrieval. We then fine-tuned our retriever for our example sentence retrieval task. Our experiments showed that the SONAR-based DPR encoder could effectively perform text-to-text,speech-to-speech, and speech-to-text retrieval, yielding 46.6%, 41.3%, and 33.3% top-1 retrieval accuracy, respectively.

Research Question 3: What criteria should be used to evaluate both the overall translation performance and the accuracy of rare words? Furthermore, how can we assess the impact of the retrieval and demonstration of example sentences on these two criteria?

We evaluate translation quality using BLEU and COMET scores and assess translation accuracy by the percentage of correctly translated rare words. We conducted experiments in various settings: translation with text-to-text retrieved examples, speech-to-text retrieved examples, and speech-to-speech retrieved examples. After comparing translation performance and accuracy in various experiment settings, we found that, compared to other modalities, speech-to-speech retrieval leads to higher overall translation quality and rare word translation accuracy (§5.3), as well as more robustness to unseen speakers (§5.4).

6.2. Future work

Language Coverage in Experiments Experiments in this thesis were limited to the English-to-German language pair due to resource constraints. Experiments on additional language pairs, especially distant ones, would further substantiate the findings.

Robustness to Irrelevant Examples The approach proposed in this thesis effectively improves the accuracy of rare word translation. However, as elaborated in the result discussions, we also observed that incorrectly retrieved examples tend to harm translation quality. As a next step, we hope to increase the robustness of the ST models to irrelevant examples. This could for instance be achieved by incorporating incorrect rare words during training to enhance the model's resilience to such errors.

Bibliography

[1] Melissa Ailem, Jinghsu Liu, and Raheel Qader. "Encouraging neural machine translation to satisfy terminology constraints". In: *arXiv preprint arXiv:2106.03730* (2021).

[2] Belen Alastruey et al. "On the locality of attention in direct speech translation". In: *arXiv preprint arXiv:2204.09028* (2022).

[3] Antonios Anastasopoulos and David Chiang. "Tied multitask learning for neural speech translation". In: *arXiv preprint arXiv:1802.06655* (2018).

[4] Chia-Wei Ao and Hung-yi Lee. "Query-by-example spoken term detection using attention-based multi-hop networks". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 6264–6268.

[5] Junyi Ao et al. "SpeechT5: Unified-Modal Encoder-Decoder Pre-Training for Spoken Language Processing". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 5723–5738. DOI: 10.18653/v1/2022.acl-long.393. URL: https://aclanthology.org/2022.acl-long.393.

[6] Akari Asai et al. "Learning to retrieve reasoning paths over wikipedia graph for question answering". In: *arXiv preprint arXiv:1911.10470* (2019).

[7] Alexei Baevski et al. "Unsupervised speech recognition". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 27826–27839.

[8] Alexei Baevski et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations". In: *Advances in neural information processing systems* 33 (2020), pp. 12449–12460.

[9] Parnia Bahar, Tobias Bieschke, and Hermann Ney. "A comparative study on end-to-end speech to text translation". In: *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2019, pp. 792–799.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[11] Sameer Bansal et al. "Low-resource speech-to-text translation". In: *arXiv preprint arXiv:1803.09164* (2018).

[12] Sameer Bansal et al. "Pre-training on high-resource speech recognition improves low-resource speech-to-text translation". In: *arXiv preprint arXiv:1809.01431* (2018).

[13] Loıc Barrault et al. “SeamlessM4T-Massively Multilingual & Multimodal Machine Translation”. In: *arXiv preprint arXiv:2308.11596* (2023).

[14] Dhanush Bekal et al. “Remember the context! ASR slot error correction through memorization”. In: *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2021, pp. 236–243.

[15] Luisa Bentivogli et al. “Cascade versus direct speech translation: Do the differences still make a difference?” In: *arXiv preprint arXiv:2106.01045* (2021).

[16] Alexandre Bérard et al. “End-to-end automatic speech translation of audiobooks”. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 6224–6228.

[17] Alexandre Bérard et al. “Listen and translate: A proof of concept for end-to-end speech-to-text translation”. In: *arXiv preprint arXiv:1612.01744* (2016).

[18] Toms Bergmanis and Mārcis Pinnis. “Facilitating terminology translation with target lemma annotations”. In: *arXiv preprint arXiv:2101.10035* (2021).

[19] Nikolay Bogoychev and Pinzhen Chen. “Terminology-Aware Translation with Constrained Decoding and Large Language Model Prompting”. In: *arXiv preprint arXiv:2310.05824* (2023).

[20] Lynne Bowker. “Productivity vs Quality? A pilot study on the impact of translation memory systems”. In: 2005. URL: https://api.semanticscholar.org/CorpusID:59517742.

[21] Marija Brkić, Sanja Seljan, and Bozena Basic Mikulic. “Using Translation Memory to Speed up Translation Process”. In: 2009. URL: https://api.semanticscholar.org/CorpusID:62878803.

[22] Tom Brown et al. “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[23] Antoine Bruguier et al. “Phoebe: Pronunciation-aware contextualization for end-to-end speech recognition”. In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 6171–6175.

[24] David M Chan et al. “Domain adaptation with external off-policy acoustic catalogs for scalable contextual end-to-end automated speech recognition”. In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2023, pp. 1–5.

[25] William Chan et al. “Listen, attend and spell”. In: *arXiv preprint arXiv:1508.01211* (2015).

[26] Ciprian Chelba, Timothy J Hazen, and Murat Saraclar. “Retrieval and browsing of spoken content”. In: *IEEE Signal Processing Magazine* 25.3 (2008), pp. 39–49.

[27] Danqi Chen et al. “Reading wikipedia to answer open-domain questions”. In: *arXiv preprint arXiv:1704.00051* (2017).

[28] Mingda Chen et al. “Improving in-context few-shot learning via self-supervised training”. In: *arXiv preprint arXiv:2205.01703* (2022).

[29] Zhehuai Chen et al. “Salm: Speech-augmented language model with in-context learning for speech recognition and translation”. In: *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2024, pp. 13521–13525.

[30] Kyunghyun Cho et al. “Learning phrase representations using RNN encoder-decoder for statistical machine translation”. In: *arXiv preprint arXiv:1406.1078* (2014).

[31] Yunfei Chu et al. “Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models”. In: *arXiv preprint arXiv:2311.07919* (2023).

[32] Shun-Po Chuang et al. “Worse wer, but better bleu? leveraging word embedding as intermediate in multitask end-to-end speech translation”. In: *arXiv preprint arXiv:2005.10678* (2020).

[33] Christopher Cieri, David Miller, and Kevin Walker. “The Fisher Corpus: a Resource for the Next Generations of Speech-to-Text”. In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC’04)*. Ed. by Maria Teresa Lino et al. Lisbon, Portugal: European Language Resources Association (ELRA), May 2004. URL: http://www.lrec-conf.org/proceedings/lrec2004/pdf/767.pdf.

[34] Josep Crego et al. “Systran’s pure neural machine translation systems”. In: *arXiv preprint arXiv:1610.05540* (2016).

[35] Zihang Dai et al. “Transformer-xl: Attentive language models beyond a fixed-length context”. In: *arXiv preprint arXiv:1901.02860* (2019).

[36] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[37] Mattia A Di Gangi, Matteo Negri, and Marco Turchi. “Adapting transformer to end-to-end spoken language translation”. In: *Proceedings of INTERSPEECH 2019*. International Speech Communication Association (ISCA), 2019, pp. 1133–1137.

[38] Mattia A. Di Gangi et al. “MuST-C: a Multilingual Speech Translation Corpus”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 2012–2017. DOI: 10.18653/v1/N19-1202. URL: https://aclanthology.org/N19-1202.

[39] Georgiana Dinu et al. “Training neural machine translation to apply terminology constraints”. In: *arXiv preprint arXiv:1906.01105* (2019).

[40] Qingxiu Dong et al. "A survey on in-context learning". In: *arXiv preprint arXiv:2301.00234* (2022).

[41] Zi-Yi Dou and Graham Neubig. "Word Alignment by Fine-tuning Embeddings on Parallel Corpora". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Ed. by Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty. Online: Association for Computational Linguistics, Apr. 2021, pp. 2112–2128. DOI: 10.18653/v1/2021.eacl-main.181. URL: https://aclanthology.org/2021.eacl-main.181.

[42] Yichao Du et al. "Non-parametric domain adaptation for end-to-end speech translation". In: *arXiv preprint arXiv:2205.11211* (2022).

[43] Long Duong et al. "An attentional model for speech translation without transcription". In: *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*. 2016, pp. 949–959.

[44] Paul-Ambroise Duquenne, Holger Schwenk, and Benoıt Sagot. "SONAR: sentence-level multimodal and language-agnostic representations". In: *arXiv e-prints* (2023), arXiv–2308.

[45] Yassir Fathullah et al. "AudioChatLlama: Towards General-Purpose Speech Abilities for LLMs". In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 2024, pp. 5522–5532.

[46] Fangxiaoyu Feng et al. "Language-agnostic BERT sentence embedding". In: *arXiv preprint arXiv:2007.01852* (2020).

[47] Philip Gage. "A new algorithm for data compression". In: *The C Users Journal* 12.2 (1994), pp. 23–38.

[48] Marco Gaido, Matteo Negri, and Marco Turchi. "Who are we talking about? handling person names in speech translation". In: *arXiv preprint arXiv:2205.06755* (2022).

[49] Marco Gaido et al. "Is" moby dick" a whale or a bird? named entities and terminology in speech translation". In: *arXiv preprint arXiv:2109.07439* (2021).

[50] Marco Gaido et al. "Named Entity Detection and Injection for Direct Speech Translation". In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2023, pp. 1–5.

[51] Marco Gaido et al. "Speech Translation with Speech Foundation Models and Large Language Models: What is There and What is Missing?" In: *arXiv preprint arXiv:2402.12025* (2024).

[52] Anthony Gillioz et al. "Overview of the Transformer-based Models for NLP Tasks". In: *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*. IEEE. 2020, pp. 179–183.

[53] Yuxian Gu et al. "Pre-training to learn in context". In: *arXiv preprint arXiv:2305.09137* (2023).

[54] Jetic Gū, Hassan S Shavarani, and Anoop Sarkar. "Pointer-based fusion of bilingual lexicons into neural machine translation". In: *arXiv preprint arXiv:1909.07907* (2019).

[55] Anmol Gulati et al. "Conformer: Convolution-augmented transformer for speech recognition". In: *arXiv preprint arXiv:2005.08100* (2020).

[56] Jinxi Guo, Tara N Sainath, and Ron J Weiss. "A spelling correction model for end-to-end speech recognition". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 5651–5655.

[57] Kelvin Guu et al. "Retrieval augmented language model pre-training". In: *International conference on machine learning*. PMLR. 2020, pp. 3929–3938.

[58] Eva Hasler et al. "Neural machine translation decoding with terminology constraints". In: *arXiv preprint arXiv:1805.03750* (2018).

[59] Amr Hendy et al. "How good are gpt models at machine translation? a comprehensive evaluation". In: *arXiv preprint arXiv:2302.09210* (2023).

[60] Matthew Honnibal et al. "spaCy: Industrial-strength Natural Language Processing in Python". In: (2020). DOI: 10.5281/zenodo.1212303.

[61] Ming-Hao Hsu et al. "An exploration of in-context learning for speech language model". In: *arXiv preprint arXiv:2310.12477* (2023).

[62] Zhichao Huang et al. "Speech Translation with Large Language Models: An Industrial Practice". In: *arXiv preprint arXiv:2312.13585* (2023).

[63] Christian Huber and Alexander Waibel. "Continuously Learning New Words in Automatic Speech Recognition". In: *arXiv preprint arXiv:2401.04482* (2024).

[64] Christian Huber et al. "Instant one-shot word-learning for context-specific neural sequence-to-sequence speech recognition". In: *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2021, pp. 1–7.

[65] Hirofumi Inaguma et al. "ESPnet-ST: All-in-one speech translation toolkit". In: *arXiv preprint arXiv:2004.10234* (2020).

[66] Bernard J Jansen et al. "Real life information retrieval: A study of user queries on the web". In: *Acm sigir forum*. Vol. 32. 1. ACM New York, NY, USA. 1998, pp. 5–17.

[67] Ye Jia et al. "Leveraging weakly supervised data to improve end-to-end speech-to-text translation". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 7180–7184.

[68] Wenxiang Jiao et al. "Is ChatGPT a good translator? Yes with GPT-4 as the engine". In: *arXiv preprint arXiv:2301.08745* (2023).

[69] Jeff Johnson, Matthijs Douze, and Hervé Jégou. "Billion-scale similarity search with GPUs". In: *IEEE Transactions on Big Data* 7.3 (2019), pp. 535–547.

[70] Vladimir Karpukhin et al. "Dense Passage Retrieval for Open-Domain Question Answering". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber et al. Online: Association for Computational Linguistics, Nov. 2020, pp. 6769–6781. DOI: 10.18653/v1/2020.emnlp-main.550. URL: https://aclanthology.org/2020.emnlp-main.550.

[71] Urvashi Khandelwal et al. "Nearest neighbor machine translation". In: *arXiv preprint arXiv:2010.00710* (2020).

[72] Omar Khattab, Christopher Potts, and Matei Zaharia. "Relevance-guided supervision for openqa with colbert". In: *Transactions of the association for computational linguistics* 9 (2021), pp. 929–944.

[73] Guillaume Klein et al. "Opennmt: Open-source toolkit for neural machine translation". In: *arXiv preprint arXiv:1701.02810* (2017).

[74] Ali Can Kocabiyikoglu, Laurent Besacier, and Olivier Kraif. "Augmenting librispeech with french translations: A multimodal corpus for direct speech translation evaluation". In: *arXiv preprint arXiv:1802.03142* (2018).

[75] Philipp Koehn and Rebecca Knowles. "Six challenges for neural machine translation". In: *arXiv preprint arXiv:1706.03872* (2017).

[76] Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. "Rankqa: Neural question answering with answer re-ranking". In: *arXiv preprint arXiv:1906.03008* (2019).

[77] Bernhard Kratzwald and Stefan Feuerriegel. "Adaptive document retrieval for deep question answering". In: *arXiv preprint arXiv:1808.06528* (2018).

[78] Taku Kudo and John Richardson. "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Ed. by Eduardo Blanco and Wei Lu. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 66–71. DOI: 10.18653/v1/D18-2012. URL: https://aclanthology.org/D18-2012.

[79] Aswanth Kumar et al. "CTQScorer: Combining multiple features for in-context example selection for machine translation". In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023, pp. 7736–7752.

[80] Shankar Kumar et al. "Lattice rescoring strategies for long short term memory language models in speech recognition". In: *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2017, pp. 165–172.

[81] Martha Larson, Gareth JF Jones, et al. "Spoken content retrieval: A survey of techniques and technologies". In: *Foundations and Trends® in Information Retrieval* 5.4–5 (2012), pp. 235–422.

[82] Hang Le et al. "Dual-decoder transformer for joint automatic speech recognition and multilingual speech translation". In: *arXiv preprint arXiv:2011.00747* (2020).

[83] Hang Le et al. "Lightweight adapter tuning for multilingual speech translation". In: *arXiv preprint arXiv:2106.01463* (2021).

[84] Chia-Hsuan Lee, Yun-Nung Chen, and Hung-Yi Lee. "Mitigating the impact of speech recognition errors on spoken question answering by adversarial domain adaptation". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 7300–7304.

[85] Jinhyuk Lee et al. “Ranking paragraphs for improving answer recall in open-domain question answering”. In: *arXiv preprint arXiv:1810.00494* (2018).

[86] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. “Latent retrieval for weakly supervised open domain question answering”. In: *arXiv preprint arXiv:1906.00300* (2019).

[87] Lin-shan Lee et al. “Spoken content retrieval—beyond cascading speech recognition with text retrieval”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.9 (2015), pp. 1389–1420.

[88] Haibo Li et al. “Name-aware machine translation”. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2013, pp. 604–614.

[89] Yuang Li et al. “Prompting large language models for zero-shot domain adaptation in speech recognition”. In: *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2023, pp. 1–8.

[90] Chyi-Jiunn Lin et al. “SpeechDPR: End-to-End Spoken Passage Retrieval for Open-Domain Spoken Question Answering”. In: *arXiv preprint arXiv:2401.13463* (2024).

[91] Alexander H Liu et al. “Towards end-to-end unsupervised speech recognition”. In: *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE. 2023, pp. 221–228.

[92] Danni Liu et al. “KIT’s Multilingual Speech Translation System for IWSLT 2023”. In: *arXiv preprint arXiv:2306.05320* (2023).

[93] Jingshu Liu et al. “Lingua custodia’s participation at the WMT 2023 terminology shared task”. In: *Proceedings of the Eighth Conference on Machine Translation*. 2023, pp. 897–901.

[94] Xunying Liu et al. “Efficient lattice rescoring using recurrent neural network language models”. In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2014, pp. 4908–4912.

[95] Yinhan Liu et al. “Multilingual denoising pre-training for neural machine translation”. In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 726–742.

[96] Yinhan Liu et al. “Roberta: A robustly optimized bert pretraining approach”. In: *arXiv preprint arXiv:1907.11692* (2019).

[97] Jing Lu et al. “Multi-stage training with improved negative contrast for neural passage retrieval”. In: *Proceedings of the 2021 conference on empirical methods in natural language processing*. 2021, pp. 6091–6103.

[98] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. “Effective approaches to attention-based neural machine translation”. In: *arXiv preprint arXiv:1508.04025* (2015).

[99] Minh-Thang Luong et al. “Addressing the rare word problem in neural machine translation”. In: *arXiv preprint arXiv:1410.8206* (2014).

[100] Puneet Mathur et al. "PersonaLM: Language Model Personalization via Domain-distributed Span Aggregated K-Nearest N-gram Retrieval Augmentation". In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023, pp. 11314–11328.

[101] Elise Michon, Josep M Crego, and Jean Senellart. "Integrating domain terminology into neural machine translation". In: *Proceedings of the 28th International Conference on Computational Linguistics*. 2020, pp. 3925–3937.

[102] Sewon Min et al. "Knowledge guided text retrieval and reading for open domain question answering". In: *arXiv preprint arXiv:1911.03868* (2019).

[103] Sewon Min et al. "Metaicl: Learning to learn in context". In: *arXiv preprint arXiv:2110.15943* (2021).

[104] Sewon Min et al. "Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 11048–11064. DOI: 10.18653/v1/2022.emnlp-main.759. URL: https://aclanthology.org/2022.emnlp-main.759.

[105] Yasmin Moslem et al. "Adaptive machine translation with large language models". In: *arXiv preprint arXiv:2301.13294* (2023).

[106] Yasmin Moslem et al. "Domain terminology integration into machine translation: Leveraging large language models". In: *Proceedings of the Eighth Conference on Machine Translation*. 2023, pp. 902–911.

[107] Jianmo Ni et al. "Large dual encoders are generalizable retrievers". In: *arXiv preprint arXiv:2112.07899* (2021).

[108] Yixin Nie, Songhe Wang, and Mohit Bansal. "Revealing the importance of semantic retrieval for machine reading at scale". In: *arXiv preprint arXiv:1909.08041* (2019).

[109] Jan Niehues. "Continuous Learning in Neural Machine Translation using Bilingual Dictionaries". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Ed. by Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty. Online: Association for Computational Linguistics, Apr. 2021, pp. 830–840. DOI: 10.18653/v1/2021.eacl-main.70. URL: https://aclanthology.org/2021.eacl-main.70.

[110] Tommi Nieminen. "Opus-cat terminology systems for the wmt23 terminology shared task". In: *Proceedings of the Eighth Conference on Machine Translation*. 2023, pp. 912–918.

[111] Stefan Ortmanns, Hermann Ney, and Xavier Aubert. "A word graph algorithm for large vocabulary continuous speech recognition". In: *Computer Speech & Language* 11.1 (1997), pp. 43–72.

[112] Myle Ott et al. "fairseq: A fast, extensible toolkit for sequence modeling". In: *arXiv preprint arXiv:1904.01038* (2019).

[113] Jing Pan et al. “Cosmic: Data efficient instruction-tuning for speech in-context learning”. In: *arXiv preprint arXiv:2311.02248* (2023).

[114] Vassil Panayotov et al. “Librispeech: an asr corpus based on public domain audio books”. In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2015, pp. 5206–5210.

[115] Kishore Papineni et al. “Bleu: a method for automatic evaluation of machine translation”. In: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002, pp. 311–318.

[116] Geon Woo Park et al. “VARCO-MT: NCSOFT’s WMT’23 terminology shared task submission”. In: *Proceedings of the Eighth Conference on Machine Translation*. 2023, pp. 919–925.

[117] Ngoc-Quan Pham, Jan Niehues, and Alex Waibel. “Towards one-shot learning for rare-word translation with external experts”. In: *arXiv preprint arXiv:1809.03182* (2018).

[118] Ngoc-Quan Pham et al. “Relative positional encoding for speech recognition and direct translation”. In: *arXiv preprint arXiv:2005.09940* (2020).

[119] Ngoc-Quan Pham et al. “Very deep self-attention networks for end-to-end speech recognition”. In: *arXiv preprint arXiv:1904.13377* (2019).

[120] Matt Post. “A Call for Clarity in Reporting BLEU Scores”. In: *Proceedings of the Third Conference on Machine Translation: Research Papers*. Ed. by Ondřej Bojar et al. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 186–191. DOI: 10.18653/v1/W18-6319. URL: https://aclanthology.org/W18-6319.

[121] Matt Post et al. “Improved speech-to-text translation with the Fisher and Callhome Spanish-English speech translation corpus”. In: *Proceedings of the 10th International Workshop on Spoken Language Translation: Papers*. 2013.

[122] Daniel Povey et al. “The Kaldi speech recognition toolkit”. In: *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society. 2011.

[123] Vineel Pratap et al. “Scaling speech technology to 1,000+ languages”. In: *Journal of Machine Learning Research* 25.97 (2024), pp. 1–52.

[124] David Qiu et al. “Context-aware neural confidence estimation for rare word speech recognition”. In: *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE. 2023, pp. 31–37.

[125] Chen Qu et al. “Open-retrieval conversational question answering”. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 2020, pp. 539–548.

[126] Leyuan Qu, Cornelius Weber, and Stefan Wermter. “Emphasizing unseen words: New vocabulary acquisition for end-to-end speech recognition”. In: *Neural Networks* 161 (2023), pp. 494–504.

[127] Alec Radford et al. "Robust speech recognition via large-scale weak supervision". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 28492–28518.

[128] Colin Raffel et al. "Exploring the limits of transfer learning with a unified text-to-text transformer". In: *Journal of machine learning research* 21.140 (2020), pp. 1–67.

[129] Anirudh Raju et al. "Scalable multi corpora neural language models for asr". In: *arXiv preprint arXiv:1907.01677* (2019).

[130] Dhananjay Ram, Lesly Miculicich, and Hervé Bourlard. "Neural network based end-to-end query by example spoken term detection". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2020), pp. 1416–1427.

[131] Ricardo Rei et al. "COMET: A Neural Framework for MT Evaluation". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber et al. Online: Association for Computational Linguistics, Nov. 2020, pp. 2685–2702. DOI: 10.18653/v1/2020.emnlp-main.213. URL: https://aclanthology.org/2020.emnlp-main.213.

[132] Ricardo Rei et al. "Unbabel's Participation in the WMT20 Metrics Shared Task". In: *arXiv preprint arXiv:2010.15535* (2020).

[133] Stephen Robertson, Hugo Zaragoza, et al. "The probabilistic relevance framework: BM25 and beyond". In: *Foundations and Trends® in Information Retrieval* 3.4 (2009), pp. 333–389.

[134] Tara N Sainath et al. "An Efficient Streaming Non-Recurrent On-Device End-to-End Model with Improvements to Rare-Word Modeling." In: *Interspeech*. Vol. 8. 2021, pp. 1777–1781.

[135] Tara N Sainath et al. "Two-pass end-to-end speech recognition". In: *arXiv preprint arXiv:1908.10992* (2019).

[136] George Saon et al. "Speaker adaptation of neural network acoustic models using i-vectors". In: *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE. 2013, pp. 55–59.

[137] Abigail See, Peter J Liu, and Christopher D Manning. "Get to the point: Summarization with pointer-generator networks". In: *arXiv preprint arXiv:1704.04368* (2017).

[138] Rico Sennrich, Barry Haddow, and Alexandra Birch. "Neural Machine Translation of Rare Words with Subword Units". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Katrin Erk and Noah A. Smith. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725. DOI: 10.18653/v1/P16-1162. URL: https://aclanthology.org/P16-1162.

[139] Rico Sennrich, Barry Haddow, and Alexandra Birch. "Neural machine translation of rare words with subword units". In: *arXiv preprint arXiv:1508.07909* (2015).

[140] Minjoon Seo et al. "Real-time open-domain question answering with dense-sparse phrase index". In: *arXiv preprint arXiv:1906.05807* (2019).

[141] Nivedita Sethiya and Chandresh Kumar Maurya. “End-to-End Speech-to-Text Translation: A Survey”. In: *arXiv preprint arXiv:2312.01053* (2023).

[142] Suzanna Sia and Kevin Duh. “In-context learning as maintaining coherency: A study of on-the-fly machine translation using large language models”. In: *arXiv preprint arXiv:2305.03573* (2023).

[143] MA Siegler et al. “Experiments in spoken document retrieval at CMU”. In: *NIST SPECIAL PUBLICATION SP* (1998), pp. 291–302.

[144] Kai Song et al. “Code-switching for enhancing NMT with pre-specified translation”. In: *arXiv preprint arXiv:1904.09107* (2019).

[145] Matthias Sperber and Matthias Paulik. “Speech translation and the end-to-end promise: Taking stock of where we are”. In: *arXiv preprint arXiv:2004.06358* (2020).

[146] Matthias Sperber et al. “Attention-passing models for robust and data-efficient end-to-end speech translation”. In: *Transactions of the Association for Computational Linguistics* 7 (2019), pp. 313–325.

[147] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. “Sequence to sequence learning with neural networks”. In: *Advances in neural information processing systems* 27 (2014).

[148] Changli Tang et al. “Salmonn: Towards generic hearing abilities for large language models”. In: *arXiv preprint arXiv:2310.13289* (2023).

[149] Yun Tang et al. “A general multi-task learning framework to leverage text data for speech to text tasks”. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 6209–6213.

[150] Romal Thoppilan et al. “Lamda: Language models for dialog applications”. In: *arXiv preprint arXiv:2201.08239* (2022).

[151] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[152] David Vilar et al. “Prompting PaLM for Translation: Assessing Strategies and Performance”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 15406–15427. DOI: 10.18653/v1/2023.acl-long.859. URL: https://aclanthology.org/2023.acl-long.859.

[153] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. “Pointer networks”. In: *Advances in neural information processing systems* 28 (2015).

[154] Alex Waibel et al. “JANUS: a speech-to-speech translation system using connectionist and symbolic processing strategies”. In: *Acoustics, speech, and signal processing, IEEE international conference on*. IEEE Computer Society. 1991, pp. 793–796.

[155] Changhan Wang et al. “Covost: A diverse multilingual speech-to-text translation corpus”. In: *arXiv preprint arXiv:2002.01320* (2020).

[156] Changhan Wang et al. “Fairseq S2T: Fast Speech-to-Text Modeling with Fairseq”. In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations*. Ed. by Derek Wong and Douwe Kiela. Suzhou, China: Association for Computational Linguistics, Dec. 2020, pp. 33–39. URL: https://aclanthology.org/2020.aacl-demo.6.

[157] Chengyi Wang et al. “Bridging the gap between pre-training and fine-tuning for end-to-end speech translation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 05. 2020, pp. 9161–9168.

[158] Chengyi Wang et al. “Curriculum pre-training for end-to-end speech translation”. In: *arXiv preprint arXiv:2004.10093* (2020).

[159] Mingqiu Wang et al. “Retrieval Augmented End-to-End Spoken Dialog Models”. In: *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2024, pp. 12056–12060.

[160] Mingqiu Wang et al. “Slm: Bridge the thin gap between speech and text foundation models”. In: *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2023, pp. 1–8.

[161] Weiran Wang et al. “Improving rare word recognition with lm-aware mwer training”. In: *arXiv preprint arXiv:2204.07553* (2022).

[162] Zhiguo Wang et al. “Multi-passage bert: A globally normalized bert model for open-domain question answering”. In: *arXiv preprint arXiv:1908.08167* (2019).

[163] Jason Wei et al. “Finetuned language models are zero-shot learners”. In: *arXiv preprint arXiv:2109.01652* (2021).

[164] Jerry Wei et al. “Symbol tuning improves in-context learning in language models”. In: *arXiv preprint arXiv:2305.08298* (2023).

[165] Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. “Constructing datasets for multi-hop reading comprehension across documents”. In: *Transactions of the Association for Computational Linguistics* 6 (2018), pp. 287–302.

[166] Jian Wu et al. “On decoder-only architecture for speech-to-text and large language model integration”. In: *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2023, pp. 1–8.

[167] Yonghui Wu et al. “Google’s neural machine translation system: Bridging the gap between human and machine translation”. In: *arXiv preprint arXiv:1609.08144* (2016).

[168] Wenhan Xiong et al. “Answering complex open-domain questions with multi-hop dense retrieval”. In: *arXiv preprint arXiv:2009.12756* (2020).

[169] Haoran Xu et al. “A paradigm shift in machine translation: Boosting translation performance of large language models”. In: *arXiv preprint arXiv:2309.11674* (2023).

[170] Chao-Han Huck Yang et al. “Multi-task language modeling for improving speech recognition of rare words”. In: *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 2021, pp. 1087–1093.

[171] Wei Yang et al. “End-to-end open-domain question answering with bertserini”. In: *arXiv preprint arXiv:1902.01718* (2019).

[172] Rong Ye, Mingxuan Wang, and Lei Li. “Cross-modal contrastive learning for speech translation”. In: *arXiv preprint arXiv:2205.02444* (2022).

[173] JC Ying et al. “Language model passage retrieval for question-oriented multi document summarization”. In: *Proc. of Document Understanding Conference*. 2007.

[174] Chenyu You et al. “Towards data distillation for end-to-end spoken conversational question answering”. In: *arXiv preprint arXiv:2010.08923* (2020).

[175] Urchade Zaratiana et al. *GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer*. 2023. arXiv: 2311.08526 [cs.CL].

[176] Thomas Zenkel et al. “Open Source Toolkit for Speech to Text Translation.” In: *Prague Bull. Math. Linguistics* 111 (2018), pp. 125–135.

[177] Biao Zhang, Barry Haddow, and Alexandra Birch. “Prompting large language model for machine translation: A case study”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 41092–41110.

[178] Hao Zhang et al. “Decoupled Non-Parametric Knowledge Distillation for end-to-End Speech Translation”. In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2023, pp. 1–5.

[179] Hao Zhang et al. “Tuning Large language model for End-to-end Speech Translation”. In: *arXiv preprint arXiv:2310.02050* (2023).

[180] Huaao Zhang et al. “Understanding and Improving the Robustness of Terminology Constraints in Neural Machine Translation”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 6029–6042. DOI: 10.18653/v1/2023.acl-long.332. URL: https://aclanthology.org/2023.acl-long.332.

[181] Tong Zhang et al. “Point, disambiguate and copy: Incorporating bilingual dictionaries for neural machine translation”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 3970–3979.

[182] Yuyu Zhang et al. “Answering any-hop open-domain questions with iterative document reranking”. In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2021, pp. 481–490.

[183] Yuyu Zhang et al. “Dc-bert: Decoupling question and document for efficient contextual encoding”. In: *arXiv preprint arXiv:2002.12591* (2020).

[184] Chengqi Zhao et al. “NeurST: Neural speech translation toolkit”. In: *arXiv preprint arXiv:2012.10018* (2020).

[185] Tiancheng Zhao, Xiaopeng Lu, and Kyusong Lee. “SPARTA: Efficient open-domain question answering via sparse transformer matching retrieval”. In: *arXiv preprint arXiv:2009.13013* (2020).

[186] Jiaming Zhou et al. “knn-ctc: Enhancing asr via retrieval of ctc pseudo labels”. In: *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2024, pp. 11006–11010.

[187] Fengbin Zhu et al. “Retrieving and reading: A comprehensive survey on open-domain question answering”. In: *arXiv preprint arXiv:2101.00774* (2021).

[188] Wenhao Zhu et al. “Multilingual machine translation with large language models: Empirical results and analysis”. In: *arXiv preprint arXiv:2304.04675* (2023).

A. Appendix

A.1. Details of Rare Word Types

The detailed rare word analysis results for Table 4.2 are in Table A.1.

Rare Word Type	Frequency
Person	130
Location	72
Technology	29
Food	27
Company	25
Biology	23
Organization	18
Health	18
Culture	14
Transport	14
Religion	14
Fashion	13
Medicine	12
Science	12
Geography	11
Chemics	11
Language	11
History	10
Politics	9
Architecture	9
Military	9
Environment	8
Education	7
Sport	7
Law	6
Society	4
Data	4
Book	4
Physics	4
Game	3
Economy	3
Literature	2
Art	2
Music	1
Entertainment	1
Award	1

Table A.1.: Detailed NER results on rare words in tst-rare-word with the number of unique words in each category.