"Listen, Understand and Translate": Triple Supervision Decouples End-to-end Speech-to-text Translation

Submission ID: 10343

Abstract

An end-to-end speech-to-text translation (ST) takes audio in a source language and outputs the text in a target language. Inspired by neuroscience, humans have perception systems and cognitive systems to process different information. We propose **LUT**, **L**isten-**U**nderstand-**T**ranslate, a unified framework with triple supervision to decouple the end-to-end speech-to-text translation task. In addition to the target sentence translation loss, LUT includes two auxiliary supervising signals to guide the acoustic encoder to extracts acoustic features from the input, and the semantic encoder to extract semantic features relevant to the source transcription text. We do experiments on both English-French and English-German speech translation benchmarks and the results demonstrate the reasonability of LUT. Our code and models will be released.

1 Introduction

Processing audio in one language and translating it into another language has been requested in many applications. Traditional speech translation (ST) systems are cascaded by connecting separately trained automatic speech recognition (ASR) and machine translation (MT) subsystems (Sperber et al. 2017, 2019b; Zhang et al. 2019; Beck, Cohn, and Haffari 2019; Cheng et al. 2019). However, such cascaded ST systems have drawbacks including higher latency, larger memory footprint, and potential error propagation in its subsystems. In contrast, an end-to-end ST system has a single unified model, which is beneficial in deployment. While very promising, existing end-to-end ST models still cannot outperform cascaded systems in terms of translation accuracy.

Cascaded ST systems usually have intermediate stages which extract acoustic features and source-text semantic features, before translating to the target text, like humans with perception systems and cognitive systems to process different information (Gazzaniga 2000). Ideally, a neural encoder-decoder network should also benefit from imitating these intermediate steps. The challenges are: *a)* there is no sufficient supervision to guide the internals of an encoder-decoder to process the audio input and obtain acoustic and semantic information properly, *b)* the training corpus for ST with pairs of source audio and target text is much smaller than those typically used for ASR and MT. Previous works attempt to relieve these challenges using pre-training and fine-tuning approaches. They usually initialize the ST model with the encoder trained on ASR data to mimic the speech transducing process and then fine-tune on a speech translation dataset to make the cross-lingual translation. However, pre-training and fine-tuning are still not sufficient enough to train an effective ST system, for the following reasons: *a)* the encoder for speech recognition is mainly used to extract acoustic information, while the ST model requires to encode both acoustic and semantic information. *b)* previous studies (Battenberg et al. 2017) have proved that the learned alignments between input and output units in ASR models are local and monotonic, which is not conducive to modeling long-distance dependencies for translation models. *c)* the gap of length between the input audio signals (typically $\sim$ 1000 frames) and target sentences (typically $\sim$ 20 tokens) renders the association from the encoder to decoder difficult to learn.

Based on the above analysis, we explore decoupled model structures, LUT, with an acoustic encoder (Listen), a semantic encoder (Understand), and a translation decoder (Translate) to imitate the intermediate steps for effective end-to-end speech translation. In addition to the normal translation loss with cross-entropy, we propose two additional auxiliary supervising signals. We introduce connectionist temporal classification (CTC) (Graves et al. 2006) loss to ensure the acoustic encoder capture necessary acoustic information from the input audio spectrum sequence. In this way, the local relations among nearby audio frames are preserved. We utilize the pre-trained embedding to guide the semantic encoder to capture a proper semantic representation. Specifically, we use the pre-trained feature extracted from BERT in this work. Notice that neither of the two auxiliary supervision is required during the model inference time and therefore our method is efficient.

The contributions of the paper can be summarized as follows:

- We design LUT, a unified framework augmented with additional components and supervision to decouple and guide the ST task.
- Our proposed method can extract semantic knowledge from the pre-trained language model (BERT) and utilize

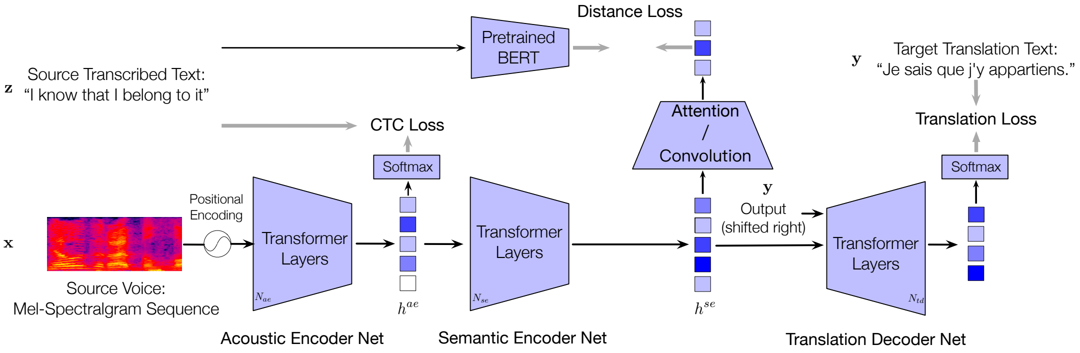

Figure 1: The architecture of LUT. It contains three modules, an acoustic encoder, a semantic encoder, and a translation decoder.

external ASR corpus to enhance acoustic modeling more effectively benefiting from the flexibly designed structure.

- We conduct experiments and do analysis on two mainstream speech translation datasets, LibriSpeech (English-French) and IWSLT2018 (English-German), to verify the effectiveness of our model.

2 Methodology

In this section, we illustrate how we design the speech-to-text translation model. The LUT architecture allows a flexible configuration of the backbone network structure in each module. One can freely choose convolutional layers, recurrent neural networks, or *Transformer* network as the main building structure. Figure 1 illustrates the overall architecture of the LUT, using *Transformer* as the backbone network. Our proposed LUT consists of three modules: *a)* an *acoustic encoder* network that encodes the audio input sequence into hidden features corresponding to the source text; *b)* a *semantic encoder* network that extracts hidden semantic representation for translation, which behaves like a normal machine translation encoder; *c)* a *translation decoder* network that emits sentence tokens in the target language. Notice an input sequence typically has a length of more than 1000, while a target sentence has tens of tokens. We have specially designed layers to cope with such a big discrepancy in lengths.

Problem Formulation The training corpus for speech translation contains speech-transcription-translation triples, denoted as $\mathcal{S} = \{(\mathbf{x}, \mathbf{z}, \mathbf{y})\}$. Specially, $\mathbf{x} = (x_1, ..., x_{T_x})$ is a sequence of acoustic features. $\mathbf{z} = (z_1, ..., z_{T_z})$ and $\mathbf{y} = (y_1, ..., y_{T_y})$ represents the corresponding text sequence in source language and target language, respectively. Meanwhile, $\mathcal{A} = \{(\mathbf{x}', \mathbf{z}')\}$ represents the external ASR corpus. Usually, the amount of end-to-end speech translation corpus is much smaller than that of ASR, i.e. $|\mathcal{S}| \ll |\mathcal{A}|$.

2.1 Acoustic Encoder

The acoustic encoder of LUT takes the input of low-level audio features and outputs a series of vectors corresponding to the transcribed text in the source language. The original audio signal is transformed into mel-frequency cepstrum (Mermelstein 1976), which is the standard preprocessing in speech recognition. The sequence of frames ($\mathbf{x}$) are processed by a feed-forward linear layer, and N_{ae} layers of *Transformer* sub-network, which includes a multi-head attention layer, a feed-forward layer, normalization layers, and residual connections. The output of acoustic encoder is denoted as $\mathbf{h}^{ae}$. They are further projected linearly with a softmax layer to obtain auxiliary output probability p for each token in the vocabulary. Note here the vocabulary is augmented with one extra blank symbol "␣". The transcribed source sentence is also split into sub-word tokens in this vocabulary. Since the length of the frame sequence, $\mathbf{x}$ is much larger than that of the transcribed source sentence $\mathbf{z}$, we employ the CTC loss to align the acoustic encoder output and the expected supervision sequence $\mathbf{z}$.

Given the ground truth transcribed token sequence $\mathbf{z}$, there can be multiple raw predicted label sequences from the acoustic encoder. Let g denote the mapping from the raw label sequence to the ground truth, which is based on a deterministic rule by removing the blank symbols and consecutive duplicate tokens. For example, $g(aa\text{␣}ab\text{␣}) = g(a\text{␣}abb\text{␣}) = aab$. We denote the set of all raw label sequences corresponding to a ground truth transcription as $g^{-1}(\mathbf{z})$. Then the conditional probability of a ground truth token sequence $\mathbf{z}$ can be modeled by marginalizing over all raw label sequences:

$$P(\mathbf{z}|\mathbf{x}) = \sum_{s \in g^{-1}(\mathbf{z})} P(s|\mathbf{x}) \tag{1}$$

Where each raw label probability $p(s|\mathbf{z})$ for a sequence s is calculated from the acoustic encoder using the following equation:

$$P(s|\mathbf{x}) = \prod_{i=1}^{T_x} p(s_i|\mathbf{x}) = \prod_{i=1}^{T_x} \mathrm{Softmax}(\mathbf{h}_i^{ae})^{s_i} \tag{2}$$

Where $\mathbf{h}^{ae}$ is the output of the acoustic encoder.

Finally, the acoustic encoder loss is defined as

$$\mathcal{L}_{ae}(\theta; \mathbf{x}, \mathbf{z}) = -\log P(\mathbf{z}|\mathbf{x}) \tag{3}$$

2.2 Semantic Encoder

The second module of LUT is motivated by the commonly used encoder for a neural machine translation model. LUT's semantic encoder aims to extract semantic and contextual information for translation. However, unlike the normal encoder in the MT model taking the input of source sentence tokens, LUT's semantic encoder takes the hidden representation $\mathbf{h}_{ae}$ computed from the acoustic encoder as the input. Since we do not have explicit supervision of the semantic representation, we utilize a pre-trained BERT model to calculate sentence embeddings for the source sentence $\mathbf{z}$ and then further employ these embeddings to supervise the training of this encoder module. This approach of self-supervision is advantageous because it enables training using a very large independent monolingual corpus in the source language. It proves to be beneficial in our experiments.

The semantic encoder contains N_{se} *Transformer* layers at the core and then connects to two branches. The output of this module is denoted as $\mathbf{h}^{se}$. One branch is to compute an overall semantic vector of the input, marked as "Seq-level Distance". It is realized using a 2D convolutional layer to reduce dimension, a normalization layer, and finally an average pooling layer to shrink the vectors into one. The output of this branch is denoted as v_0^{se}, which is a single vector. This is to be compared with the class-label representation h_c^{BERT} calculated by a BERT model. Another branch is aimed to match the semantic representation of the transcribed source sentence, marked as "Word-level Distance". This branch is connected to an auxiliary layer to calculate the length-synchronized semantic representation, which is a sequence of vectors of the size T_z, equivalent to that of the transcribed source sentence. To this end, we first use a separately pre-trained BERT model to calculate the sentence embedding vectors, excluding the class-label vector h_c^{BERT}. These vectors are organized into T_z time steps, denoted as $\mathbf{h}^{\text{BERT}}$. Suppose the N_{se}-layer transformer outputs a sequence of vectors at length T_x, denoted as $\mathbf{v} = \mathbf{h}^{se}$. Note each of these vectors are split into J heads, i.e. $\mathbf{h}^{\text{BERT}} = (\mathbf{h}_1^{\text{BERT}}, \ldots, \mathbf{h}_J^{\text{BERT}})$ and $\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_J)$. These BERT vectors are used as queries to compute the attention weights for the branch input hidden vectors.

$$\text{head}_i = \text{Attn}(\mathbf{h}_i^{\text{BERT}} W_i^Q, \mathbf{v}_i W_i^K, \mathbf{v}_i W_i^V) \tag{4}$$

Where the W_i^Q,W_i^K,W_i^V are parameters for the attention of i-th head. The attention is calculated by scaled dot-product layer, as follows:

$$\text{Attn}(Q, K, V) = \text{Softmax}(\frac{Q \cdot K^T}{\sqrt{d_k}})V \tag{5}$$

where d_k is the dimension of the key K. With this layer, the output can be reduced to the same length as source text by concatenating the heads.

$$\mathbf{v}_1^{se} = \text{Concat}(\text{head}_1, \ldots, \text{head}_J) \tag{6}$$

Finally, the semantic encoder loss is defined as the distance between the calculated hidden representations and the BERT embeddings.

$$\mathcal{L}_{se}(\theta; \mathbf{z}) = \begin{cases} |v_0^{se} - h_c^{\text{BERT}}|, & \text{Seq-level} \\ |\mathbf{v}_1^{se} - \mathbf{h}^{\text{BERT}}|, & \text{Word-level} \end{cases} \tag{7}$$

The key insight of our formulation is that the semantic encoder needs to behave like a text encoder of a neural machine translation model, with only source language text data in the training. The specifically designed loss ensures that the semantic encoder could produce similar semantic embeddings close to the BERT representation trained on a separate large text corpus. During the inference time, the output of this module is $\mathbf{h}^{se}$, therefore no additional source transcription text is needed and the BERT calculation is saved.

2.3 Translation Decoder

As with the normal machine translation model, our proposed LUT uses N_{td} layers of *Transformer* network as the decoder. Additional attention from the decoder to the semantic encoder output $\mathbf{h}^{se}$ is added. We use the cross entropy loss to measure the translation decoding performance.

$$\mathcal{L}_{td}(\theta; \mathbf{y}) = -\sum_{i=1}^{T_y} \log p_\theta(y_i^t | y_{<i}^t, \mathbf{h}^{se}) \tag{8}$$

As usual, the decoder probability is calculated from the final softmax layer based on the output of the decoder.

The overall objective function for end-to-end training is the sum from three supervision modules:

$$\begin{aligned} &\mathcal{L}(\theta; \mathbf{x}, \mathbf{z}, \mathbf{y}) \\ &= \alpha\mathcal{L}_{ae}(\theta; \mathbf{x}, \mathbf{z}) + \beta\mathcal{L}_{se}(\theta; \mathbf{x}, \mathbf{z}) + \gamma\mathcal{L}_{td}(\theta; \mathbf{x}, \mathbf{y}) \end{aligned} \tag{9}$$

where θ is the model parameter. α, β and γ are hyper-parameters to balance among the acoustic transducer loss $\mathcal{L}_{ae}$, the semantic encoder loss $\mathcal{L}_{se}$, and the translation decoder loss $\mathcal{L}_{td}$.

3 Experiments

3.1 Data

Augmented LibriSpeech Dataset Augmented LibriSpeech (Kocabiyikoglu, Besacier, and Kraif 2018) is built by automatically aligning e-books in French with English utterances of LibriSpeech. The dataset includes four types of information: English speech signal, English transcription, French text translations from the alignment of e-books with augmented references via Google Translate. Following the previous work (Liu et al. 2019a), we also conduct experiments on the 100 hours clean train set for training, with 2 hours development set and 4 hours test set, corresponding to 47271, 1071, and 2048 utterances respectively.

IWSLT2018 English-German Dataset IWSLT2018 English-German (Jan et al. 2018) is the KIT end-to-end speech translation corpus, which is built automatically by aligning English audios with SRT transcripts for English and German from lectures online. The raw data, including long wav files, English transcriptions, and the corresponding German translations, is segmented into chunks with the attached time stamps and made forced alignments using the gentle toolkit[1], according to the officially released

[1] https://github.com/lowerquality/gentle

version. It should be noted that some transcriptions are not aligned with their corresponding audio well. Noisy data is harmful to models' performance, which can be avoided by data filtering, re-alignment, and re-segmentation (Liu et al. 2018). In this paper, the original data is used directly as training data to verify our method, with a size of 272 hours and 171121 segmentations. We use *dev2010* as validation set, and *tst2010, tst2013, tst2014, tst2015* as test set, corresponding to 653, 1337, 793, 957 and 1177 utterances respectively.

TED English-Chinese Dataset English-Chinese TED is crawled from TED website[2] and released by (Liu et al. 2019a) as a benchmark for speech translation from English audio to Chinese text. Following the previous work (Liu et al. 2019a), we use dev2010 as development set and tst2015 as test set. The raw long audio is segmented based on timestamps for complete semantic information. Finally, we get 524 hours train set, 1.5 hours test set and 2.5 hours test set, corresponding to 308,660, 835, 1223 utterances respectively.

LIUM2 Dataset We use LIUM2 as the external ASR parallel corpus ($\in \mathcal{A}$) used in the expanded experimental setting for broad reproducibility. LIUM2 (Rousseau, Deléglise, and Esteve 2014) is composed of segments of public talks extracted from the lecture website[3] with 207 hours of speech data. Speed perturbation is performed on the raw signals with speed factors 0.9 and 1.1.

Data Preprocessing Following the efforts of (Liu et al. 2019a; Wang et al. 2020), we introduce acoustic features that are 80 dimensional log Mel filterbanks. The features are extracted with a step size of 10ms and a window size of 25ms and extended with mean subtraction and variance normalization. The features are stacked with 5 frames to the right and downsampled to a 30ms frame rate. For target language text data, we lowercase all the texts, tokenize and apply normalize punctuations with the Moses scripts[4]. For source language text data, we lowercase all the texts, tokenize and remove the punctuation to make the data more consistent with the output of ASR. We apply BPE[5] on the combination of source and target text to obtain shared subword units. The number of merge operations in BPE for ASR and MT systems is set to 8k and 30k, respectively. For strategies using BERT features, we apply the same preprocessing tool as BERT does to text data for ST models and regenerate the vocabulary. For English-French and English-German corpora, we report case-insensitive BLEU scores by `multi-bleu.pl`[6] script for the evaluation of ST and MT tasks. And for English-Chinese corpus, we report character-level BLEU scores. We use word error rates (WER) to evaluate ASR tasks.

3.2 Baselines

We conduct our experiments in three following settings.

Base Setting with only Speech-translation Data Our main purpose is to compare our method with conventional end-to-end speech translation models. In the experiment, the base setting is restricted to only the triple data. To pre-train the encoder, only the audio-transcription pair of the triple data can be leveraged to train an ASR model.

Expanded Setting with External Data In the context of expanded setting, Bahar et al. (2019) apply the SpecAugment (Park et al. 2019) on Librispeech English-French ST task, where his team uses a total of 236h of speech for ASR pre-training. Inaguma et al. (2019) combine three ST datasets of 472h training data to train a multilingual ST model for both Librispeech English-French ST task and IWSLT2013 English-German ST task. And (Wang et al. 2019) introduce an additional 272h ASR corpus and 41M parallel data from WMT18 to enhance the ST performance.

MT Systems The input of MT systems is the manual transcribed text which can be regarded as the upper bound of ST models.

3.3 Details of Our Model and Experiments

For ST tasks, we use a similar hyper-parameter setting with the *Transformer* base model (Vaswani et al. 2017) for the stack of *Transformer* layers, in which we set the hidden size $d_{model} = 768$ to match the output of BERT. Learning from speech-transformer (Dong, Xu, and Xu 2018), one third of the layer is used for the decoder ($N_td = 4$). N_{ae} and N_{se} are both set to 4 for our best performance, discussed in Section 4.1. For ASR and MT tasks, the standard *Transformer* base model is adopted. All samples are batched together with 20000-frame features by approximate feature sequence length during training. We train our models on 1 NVIDIA V100 GPUs with a maximum number of training steps 400k. We use a greedy search and beam search (as default) with a beam size of 8 for our experimental settings. The maximum decoding length for ASR and ST (MT) is set to 200 and 250, respectively. The hyper-parameters in Equation 9, α, β and γ are set to 0.5, 0.05, 0.45 (details in Table 11 in Appendix). For experiments in the base setting, the ST model is trained from scratch. For experiments in the expanded setting, the ST model is trained as the following two steps: *a)* pre-training the acoustic encoder with CTC loss with $(\mathbf{x}', \mathbf{z}') \in \mathcal{A}$, as Section 2.1. *b)* fine-tuning the overall ST model with $(\mathbf{x}, \mathbf{z}, \mathbf{y}) \in \mathcal{S}$, as Equation 9.

[2] https://www.ted.com

[3] http://www.ted.com

[4] https://github.com/moses-smt/mosesdecoder

[5] https://github.com/rsennrich/subword-nmt

[6] https://github.com/moses-smt/mosesdecoder/scripts/generic/multi-bleu.perl

Method	Enc Pre-train (speech data)	Dec Pre-train (text data)	greedy	beam
MT system				
Transformer MT	-	-	20.98	21.51
Base ST setting				
LSTM ST (Bérard et al. 2018)	✗	✗	12.30	12.90
+pre-train+multitask (Bérard et al. 2018)	✓	✓	12.60	13.40
LSTM ST+pre-train (Inaguma et al. 2020)	✓	✓	-	16.68
Transformer+pre-train (Liu et al. 2019a)	✓	✓	13.89	14.30
+knowledge distillation (Liu et al. 2019a)	✓	✓	14.96	17.02
TCEN-LSTM (Wang et al. 2019)	✓	✓	-	17.05
LUT	✗	✗	**16.70**	**17.75**
Expanded ST setting				
LSTM+pre-train+SpecAugment (Bahar et al. 2019)	✓(236h)	✓	-	17.00
Multilingual ST+PT (Inaguma et al. 2019)	✓(472h)	✗	-	17.60
LUT	✓(207h)	✗	**17.55**	**18.34**

Table 1: Performance (BLEU) on Augmented Librispeech English-French test set. MT model only translates from the transcribed source text, which serves as an upper limit. Our proposed LUT achieves the best performance.

Method	Enc Pre-train (speech data)	Dec Pre-train (text data)	tst2010	tst2013	tst2014	tst2015	Avg
MT system							
RNN MT (Inaguma et al. 2020)	-	-	23.80	24.90	21.17	22.33	23.05
Base ST setting							
ESPnet (Inaguma et al. 2020)	✗	✗	13.77	12.50	11.50	12.68	12.61
+enc pre-train	✓	✗	14.46	13.12	11.62	11.30	12.63
+enc dec pre-train	✓	✓	14.98	13.54	12.33	11.67	13.13
LUT	✗	✗	**16.60**	**16.35**	**13.25**	**14.37**	**15.16**
Expanded ST setting							
Multilingual ST (Inaguma et al. 2019)	✓(472h)	✗	-	14.6	-	-	-
CL-fast* (Kano, Sakti, and Nakamura 2018)	✓(479h)	✗	-	14.33	-	-	-
TCEN-LSTM (Wang et al. 2019)	✓(479h)	✓(40M)	15.49	15.50	13.21	13.02	14.31
LUT	✓(207h)	✗	**17.07**	**16.42**	**13.63**	**14.97**	**15.52**

Table 2: Performance (BLEU) on IWSLT2018 English-German test set. MT model only translates from the transcribed source text, which serves as an upper limit. Our proposed LUT achieves the best performance.

4 Results

4.1 Main Results

Librispeech English-French For En-Fr experiments, we compare the performance with existing end-to-end methods in Table 1. Clearly, LUT outperforms the previous best results by more than 0.7 BLEU in base setting and 0.74 BLEU in expanded setting respectively. Specifically, in the base setting, the model we propose outperforms ESPnet, which is equipped with both a well-trained encoder and decoder. We also achieve better results than a knowledge distillation baseline in which an MT model is introduced to teach the ST model (Liu et al. 2019a). Different from previous work, our work focuses on reducing the modeling burdens of the encoder by suggesting that auxilliary supervision signals make it easier to learn both the acoustic and semantic information. This proposal promises great potential for the application of the double supervised encoder. Compared to the TCEN baseline which includes two encoders, LUT is simple and flexible, without the need to introduce additional computational cost for inference. Simple yet effective, LUT achieves the best performance in this benchmark dataset in terms of BLEU.

IWSLT2018 English-German For En-De experiments, we compare the performance with existing end-to-end methods in Table 2. Unlike that of Librispeech English-French, this dataset is noisy, and the transcriptions do not align well with the corresponding audios. As a result, there is a wide gap between the performance of the end to end ST and the upper bound of the ST. Overall, our method outperforms ESPnet on all test sets by averaged 2.03 bleu in the base setting and has an advantage of averaged 1.21 bleu compared with TCEN (Wang et al. 2019). To be notice, our LUT does not include any pretraining tricks, and achieves the state-of-the-art performance in the base ST setting. This trend is consistent with that in the Librispeech dataset.

Method	Enc Pre-train (speech data)	Dec Pre-train (text data)	BLEU
MT system			
Transformer MT (Liu et al. 2019a)	-	-	27.08
Base setting			
Transformer+pre-train (Liu et al. 2019a)	✓	✓	16.80
+knowledge distillation (Liu et al. 2019a)	✓	✓	19.55
LUT	✗	✗	**20.84**

Table 3: Performance for MT, ST tasks on English-Chinese TED test set. *: re-implemented. Our proposed LUT achieves the best results.

TED English-Chinese For En-Zh experiments, we compared the performance with existing end-to-end methods in Table 3. Under the base setting, LUT exceeded the Transformer-based ST model augmented by knowledge distillation with 0.7 bleu, proving the validity of our method.

Comparison with Cascaded Baselines Table 4 shows the comparison with cascaded ST systems on Augmented Librispeech En-Fr test set, En-De TED tst2013 set and En-Zh test set. For a fair comparison, we do the experiments on the base settings of English-French/German/Chinese translation. Results show that LUT either receives the equivalent performance or outperforms with cascaded methods in two datasets, thus displaying great potential for the end-to-end approach. This indicates our flexible structure can make good use of additional ASR corpus and learn valuable linguistic knowledge.

	Method	BLEU
$En \rightarrow Fr$	Pipeline	17.58
	LUT	17.75
$En \rightarrow De$	Pipeline	15.38
	LUT	16.35
$En \rightarrow Zh$	Pipeline	21.36
	LUT	20.84

Table 4: LUT versus cascaded systems on Augmented Librispeech En-Fr test set and En-De TED tst2013 set. "Pipeline" systems consist of separate ASR and MT models trained independently.

4.2 Ablation Study

Effects of Auxiliary Supervision We first study the effects of two auxiliary supervision for LUT. The results in Table 5 show that all the auxiliary supervision indicate positive results that can be superimposed. Models that use supervision only from the acoustic encoder can be regarded as a method of multi-task learning, which has a significant performance improvement compared to the model of direct pre-training and fine-tuning (seen in Table 1). This reflects the catastrophic forgetting problem that occurs in the sequential transfer learning based on the pre-training method.

	Dev Bleu	Test Bleu
LUT	**18.51**	**17.75**
w/o Semantic Encoder Loss	17.72	16.81
w/o Acoustic Encoder Loss *	16.91	15.48
w/o Acoustic Encoder Loss	12.05	11.24

Table 5: Effects of LUT on En-Fr validation and test set. "*" means using ASR pre-training as initialization.

Balance of Acoustic and Semantic Modeling Experimental results, shown in Table 6 prove that the performance is better when the two modules are balanced. In order to determine which module has a more significant impact on performance, we conducted experiments on the layer number allocation of the two modules, in which the total number of layers of the acoustic transducer and semantic encoder is fixed, and the number of layers for one module is adjusted from 2 to 6. As the number of layers decreases, the two modules will both result in worse performance degradation, thus explaining that using enough layers to extract acoustic features and encode semantic representation is equally essential to the speech translation model.

N_{ae}	N_{se}	Dev BLEU	Test BLEU
2	6	14.81	13.09
3	5	17.01	15.50
4	4	**17.93**	**16.70**
5	3	17.07	16.21
6	2	16.47	15.49

Table 6: Performance on En-Fr corpus: LUT with varying layers in its acoustic encoder (N_{ae}) and semantic encoder (N_{se}). Greedy decoding is employed.

Sequence-level Distance v.s. Word-level Distance For this part, we conduct experiments with different branches described in Section 2.2. We conduct an experimental comparison of the performance differences caused by the pre-training features extracted by different layers of BERT for semantic encoder's supervision. We found that the pre-trained features of the higher layers of BERT have similar supervisory effects. We then adopt the pre-trained features from the last layer of BERT as our default setting. The results, as shown in Table 7, prove that the word-level distance benefit more from the BERT pre-trained features because of its finer and grainer regulation.

	Dev BLEU	Test BLEU
Seq-level Distance	17.64	16.61
Word-level Distance	**17.93**	**16.70**

Table 7: Performance on En-Fr corpus: LUT with different losses for semantic encoder. “Seq-level” and “Word-level” losses are described in Eq. (7). Greedy decoding is employed.

4.3 Analysis

	SpeakerVer	IntentIde
AT Output $\mathbf{h}^{at}$	**97.6**	91.0
SE output $\mathbf{h}^{se}$	46.3	**93.1**

Table 8: Classification accuracy on speaker verification and intent identification, using LUT’s acoustic transducer (AT) and semantic encoder (SE) output embeddings.

Acoustic or Semantic In Table 8, we design auxiliary probing tasks to further analyze the learned representation (Lugosch et al. 2019). **SpeakerVer** is designed to identify the speaker, therefore it benefits more from acoustic information. **IntentIde** is focused on intention recognition, so it needs more linguistic knowledge. We use the Fluent Speech Commands dataset (Lugosch et al. 2019) for experiments which contains 30,043 utterances, 97 speakers, and 31 intents. For the train split, we extract the hidden output of each layer of our well-trained LUT encoder and freeze it, followed by a fully connected layer. We then fine-tune for 20,000 steps on the two probing tasks respectively. We report the accuracy of the test split. It can be seen that during the modeling process of ST, acoustic information is modeled at low-level layers and semantic information is captured at high-level layers.

Case Study Table 9 shows our case study analysis, proving that the end-to-end speech translation system can alleviate the problem of error propagation caused by upstream speech recognition errors. LUT can obtain the intermediate results of speech recognition by the way of CTC decoding, so it can perform a certain degree of interpretable diagnosis on the translation results. Benefiting from the ability of the end-to-end system to directly obtain the original audio information, our method is fault-tolerant in the case of incorrect recognition, missing recognition, repeated recognition, and son on during the first stage.

Speech #1	766-144485-0090.wav
Transcription	
reference	it was mister jack maldon
hypothesis	it was mister jack mal
Translation	
reference	c’était m. jack maldon
hypothesis	c’était m. jack maldon
Phenomenon	incorrect recognition
Speech #2	1257-122442-0101.wav
Transcription	
reference	cried the old soldier
hypothesis	cried the soldier
Translation	
reference	s’écria le vieux soldat,
hypothesis	s’écria le vieux soldat,
Phenomenon	missing recognition
Speech #3	1184-121026-0000.wav
Transcription	
reference	chapter seventeen the abbes chamber
hypothesis	chapter seventeen teen the abbey chamber
Translation	
reference	chapitre xvii la chambre de l’abbé.
hypothesis	chapitre xvii la chambre de l’abbé.
Phenomenon	repeated recognition

Table 9: Examples of transcription and translation on En-Fr test set generated by LUT. Text in pink means correct tokens, and text in red represents incorrect tokens.

5 Related Work

End-to-end ST Previous works (Bérard et al. 2016; Duong et al. 2016) have proved the potential for end-to-end ST, which has attracted intensive attentions (Vila et al. 2018; Salesky et al. 2018; Salesky, Sperber, and Waibel 2019; Di Gangi, Negri, and Turchi 2019; Bahar, Bieschke, and Ney 2019; Di Gangi et al. 2019; Inaguma et al. 2020). It’s proved that pre-training (Weiss et al. 2017; Bérard et al. 2018; Bansal et al. 2018; Stoian, Bansal, and Goldwater 2020) and multi-task learning (Vydana et al. 2020) can significantly improve the performance. Two-pass decoding (Sung et al. 2019) and attention-passing (Anastasopoulos and Chiang 2018; Sperber et al. 2019a) techniques are proposed to handle deeper relationships and alleviate error propagation in end-to-end models. Data augmentation techniques (Jia et al. 2019; Pino et al. 2019b; Bahar et al. 2019; Pino et al. 2019a) are proposed to utilize ASR and MT corpora to generate fake data. Semi-supervised training (Wang et al. 2019) brings great gains to end-to-end models, such as knowledge distillation (Liu et al. 2019a), modality agnostic meta-learning (Indurthi et al. 2019), model adaptation (Di Gangi et al. 2020) and son on. Curriculum learning (Kano, Sakti, and Nakamura 2018; Wang et al. 2020) is proposed to improve performance of ST. Liu et al. (2019b); Liu, Spanakis, and Niehues (2020) optimize the decoding strategy to achieve low-latency end-to-end ST. (Chuang et al. 2020; Salesky and Black 2020; Salesky, Sperber, and Black 2019) explore additional features to enhance end-to-end models. The most related work may be Wang et al. (2020), however, there are much differences between our method and theirs. Wang et al. (2020) focused on the pre-train method to improve the speech translation performance, while LUT focuses on the neural model. Specifically, LUT can decouple the modeling process to better leverage the speech-transcription-translation triple supervision.

Knowledge Distillation in MT Representations learned by pre-training have been widely applied in the field of machine translation. There're different modeling granularities for representation learning, which can be utilized by fine-tuning, freezing, or distillation. Many knowledge distillation methods have been extended to transfer the rich knowledge using teacher-student architecture (Gou et al. 2020). Kim and Rush (2016) extended word-level knowledge distillation into sequence-level knowledge distillation for directing the sequence distribution of the student model. Sequence-level knowledge distillation was further explained from the perspective of data augmentation and regularization in Gordon and Duh (2019). Zhou, Neubig, and Gu (2019) studied how knowledge distillation affects the non-autoregressive MT models by empirical analysis. Lample and Conneau (2019); Edunov, Baevski, and Auli (2019) feeded the last layer of ELMo or BERT to the encoder of MT model for learning better representation. Yang et al. (2019) firstly leveraged ssymptotic distillation to transfer the pre-training information to MT model.

6 Conclusion

In this paper, we propose LUT, a novel and unified training framework to decouple the end-to-end speech translation task. We empirically validate the effectiveness of our approach as compared to previous methods on two benchmark datasets, and empirical analysis suggests that LUT is capable of capturing both acoustic and semantic information properly.

References

Anastasopoulos, A.; and Chiang, D. 2018. Tied multitask learning for neural speech translation. *arXiv preprint arXiv:1802.06655* .

Bahar, P.; Bieschke, T.; and Ney, H. 2019. A comparative study on end-to-end speech to text translation. *arXiv preprint arXiv:1911.08870* .

Bahar, P.; Zeyer, A.; Schlüter, R.; and Ney, H. 2019. On using specaugment for end-to-end speech translation. *arXiv preprint arXiv:1911.08876* .

Bansal, S.; Kamper, H.; Livescu, K.; Lopez, A.; and Goldwater, S. 2018. Pre-training on high-resource speech recognition improves low-resource speech-to-text translation. *arXiv preprint arXiv:1809.01431* .

Battenberg, E.; Chen, J.; Child, R.; Coates, A.; Li, Y. G. Y.; Liu, H.; Satheesh, S.; Sriram, A.; and Zhu, Z. 2017. Exploring neural transducers for end-to-end speech recognition. In *ASRU*, 206–213. IEEE.

Beck, D.; Cohn, T.; and Haffari, G. 2019. Neural Speech Translation using Lattice Transformations and Graph Networks. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, 26–31.

Bérard, A.; Besacier, L.; Kocabiyikoglu, A. C.; and Pietquin, O. 2018. End-to-end automatic speech translation of audiobooks. In *ICASSP*, 6224–6228. IEEE.

Bérard, A.; Pietquin, O.; Servan, C.; and Besacier, L. 2016. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744* .

Cheng, Q.; Fang, M.; Han, Y.; Huang, J.; and Duan, Y. 2019. Breaking the Data Barrier: Towards Robust Speech Translation via Adversarial Stability Training. *arXiv preprint arXiv:1909.11430* .

Chuang, S.-P.; Sung, T.-W.; Liu, A. H.; and Lee, H.-y. 2020. Worse WER, but Better BLEU? Leveraging Word Embedding as Intermediate in Multitask End-to-End Speech Translation. *arXiv preprint arXiv:2005.10678* .

Di Gangi, M. A.; Negri, M.; Cattoni, R.; Roberto, D.; and Turchi, M. 2019. Enhancing transformer for end-to-end speech-to-text translation. In *Machine Translation Summit XVII*, 21–31. European Association for Machine Translation.

Di Gangi, M. A.; Negri, M.; and Turchi, M. 2019. Adapting transformer to end-to-end spoken language translation. In *INTERSPEECH 2019*, 1133–1137. International Speech Communication Association (ISCA).

Di Gangi, M. A.; Nguyen, V.-N.; Negri, M.; and Turchi, M. 2020. Instance-based Model Adaptation for Direct Speech Translation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7914–7918. IEEE.

Dong, L.; Xu, S.; and Xu, B. 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *ICASSP*, 5884–5888. IEEE.

Duong, L.; Anastasopoulos, A.; Chiang, D.; Bird, S.; and Cohn, T. 2016. An attentional model for speech translation without transcription. In *NAACL*, 949–959.

Edunov, S.; Baevski, A.; and Auli, M. 2019. Pre-trained language model representations for language generation. *arXiv preprint arXiv:1903.09722* .

Gazzaniga, M. S. 2000. *Cognitive neuroscience: A reader.* Blackwell Publishing.

Gordon, M. A.; and Duh, K. 2019. Explaining Sequence-Level Knowledge Distillation as Data-Augmentation for Neural Machine Translation. *arXiv preprint arXiv:1912.03334* .

Gou, J.; Yu, B.; Maybank, S. J.; and Tao, D. 2020. Knowledge Distillation: A Survey. *arXiv preprint arXiv:2006.05525* .

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 369–376. ACM.

Inaguma, H.; Duh, K.; Kawahara, T.; and Watanabe, S. 2019. Multilingual end-to-end speech translation. *arXiv preprint arXiv:1910.00254* .

Inaguma, H.; Kiyono, S.; Duh, K.; Karita, S.; Soplin, N. E. Y.; Hayashi, T.; and Watanabe, S. 2020. ESPnet-ST: All-in-One Speech Translation Toolkit. *arXiv preprint arXiv:2004.10234* .

Indurthi, S.; Han, H.; Lakumarapu, N. K.; Lee, B.; Chung, I.; Kim, S.; and Kim, C. 2019. Data Efficient Direct Speech-to-Text Translation with Modality Agnostic Meta-Learning. *arXiv preprint arXiv:1911.04283* .

Jan, N.; Cattoni, R.; Sebastian, S.; Cettolo, M.; Turchi, M.; and Federico, M. 2018. The iwslt 2018 evaluation campaign. In *International Workshop on Spoken Language Translation*, 2–6.

Jia, Y.; Johnson, M.; Macherey, W.; Weiss, R. J.; Cao, Y.; Chiu, C.-C.; Ari, N.; Laurenzo, S.; and Wu, Y. 2019. Leveraging weakly supervised data to improve end-to-end speech-to-text translation. In *ICASSP*, 7180–7184. IEEE.

Kano, T.; Sakti, S.; and Nakamura, S. 2018. Structured-based curriculum learning for end-to-end english-japanese speech translation. *arXiv preprint arXiv:1802.06003* .

Kim, Y.; and Rush, A. M. 2016. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947* .

Kocabiyikoglu, A. C.; Besacier, L.; and Kraif, O. 2018. Augmenting Librispeech with French translations: A multimodal corpus for direct speech translation evaluation. *arXiv preprint arXiv:1802.03142* .

Lample, G.; and Conneau, A. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291* .

Liu, D.; Liu, J.; Guo, W.; Xiong, S.; Ma, Z.; Song, R.; Wu, C.; and Liu, Q. 2018. The USTC-NEL Speech Translation system at IWSLT 2018. *arXiv preprint arXiv:1812.02455* .

Liu, D.; Spanakis, G.; and Niehues, J. 2020. Low-Latency Sequence-to-Sequence Speech Recognition and Translation by Partial Hypothesis Selection. *arXiv preprint arXiv:2005.11185* .

Liu, Y.; Xiong, H.; He, Z.; Zhang, J.; Wu, H.; Wang, H.; and Zong, C. 2019a. End-to-End Speech Translation with Knowledge Distillation. *arXiv preprint arXiv:1904.08075* .

Liu, Y.; Zhang, J.; Xiong, H.; Zhou, L.; He, Z.; Wu, H.; Wang, H.; and Zong, C. 2019b. Synchronous Speech Recognition and Speech-to-Text Translation with Interactive Decoding. *arXiv preprint arXiv:1912.07240* .

Lugosch, L.; Ravanelli, M.; Ignoto, P.; Tomar, V. S.; and Bengio, Y. 2019. Speech Model Pre-training for End-to-End Spoken Language Understanding. *arXiv preprint arXiv:1904.03670* .

Mermelstein, P. 1976. Distance measures for speech recognition, psychological and instrumental. *Pattern Recognition and Artificial Intelligence* 374–388.

Park, D. S.; Chan, W.; Zhang, Y.; Chiu, C.-C.; Zoph, B.; Cubuk, E. D.; and Le, Q. V. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779* .

Pino, J.; Puzon, L.; Gu, J.; Ma, X.; McCarthy, A. D.; and Gopinath, D. 2019a. Harnessing indirect training data for end-to-end automatic speech translation: Tricks of the trade. In *Proceedings of the 16th International Workshop on Spoken Language Translation (IWSLT)*.

Pino, J.; Puzon, L.; Gu, J.; Ma, X.; McCarthy, A. D.; and Gopinath, D. 2019b. Leveraging Out-of-Task Data for End-to-End Automatic Speech Translation. *arXiv preprint arXiv:1909.06515* .

Rousseau, A.; Deléglise, P.; and Esteve, Y. 2014. Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks. In *LREC*, 3935–3939.

Salesky, E.; and Black, A. W. 2020. Phone Features Improve Speech Translation. *arXiv preprint arXiv:2005.13681* .

Salesky, E.; Burger, S.; Niehues, J.; and Waibel, A. 2018. Towards fluent translations from disfluent speech. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, 921–926. IEEE.

Salesky, E.; Sperber, M.; and Black, A. W. 2019. Exploring phoneme-level speech representations for end-to-end speech translation. *arXiv preprint arXiv:1906.01199* .

Salesky, E.; Sperber, M.; and Waibel, A. 2019. Fluent translations from disfluent speech in end-to-end speech translation. *arXiv preprint arXiv:1906.00556* .

Sperber, M.; Neubig, G.; Niehues, J.; and Waibel, A. 2017. Neural lattice-to-sequence models for uncertain inputs. *arXiv preprint arXiv:1704.00559* .

Sperber, M.; Neubig, G.; Niehues, J.; and Waibel, A. 2019a. Attention-Passing Models for Robust and Data-Efficient End-to-End Speech Translation. *TACL* 7: 313–325.

Sperber, M.; Neubig, G.; Pham, N.-Q.; and Waibel, A. 2019b. Self-Attentional Models for Lattice Inputs. *arXiv preprint arXiv:1906.01617* .

Stoian, M. C.; Bansal, S.; and Goldwater, S. 2020. Analyzing ASR pretraining for low-resource speech-to-text translation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7909–7913. IEEE.

Sung, T.-W.; Liu, J.-Y.; Lee, H.-y.; and Lee, L.-s. 2019. Towards End-to-end Speech-to-text Translation with Two-pass Decoding. In *ICASSP*, 7175–7179. IEEE.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NIPS*, 5998–6008.

Vila, L. C.; Escolano, C.; Fonollosa, J. A.; and Costa-jussà, M. R. 2018. End-to-End Speech Translation with the Transformer. In *IberSPEECH*, 60–63.

Vydana, H. K.; Karafi'at, M.; Zmolikova, K.; Burget, L.; and Cernocky, H. 2020. Jointly Trained Transformers models for Spoken Language Translation. *arXiv preprint arXiv:2004.12111* .

Wang, C.; Wu, Y.; Liu, S.; Yang, Z.; and Zhou, M. 2019. Bridging the Gap between Pre-Training and Fine-Tuning for End-to-End Speech Translation. *arXiv preprint arXiv:1909.07575* .

Wang, C.; Wu, Y.; Liu, S.; Zhou, M.; and Yang, Z. 2020. Curriculum Pre-training for End-to-End Speech Translation. *arXiv preprint arXiv:2004.10093* .

Weiss, R. J.; Chorowski, J.; Jaitly, N.; Wu, Y.; and Chen, Z. 2017. Sequence-to-sequence models can directly translate foreign speech. *arXiv preprint arXiv:1703.08581* .

Yang, J.; Wang, M.; Zhou, H.; Zhao, C.; Yu, Y.; Zhang, W.; and Li, L. 2019. Towards Making the Most of BERT in Neural Machine Translation. *arXiv preprint arXiv:1908.05672* .

Zhang, P.; Chen, B.; Ge, N.; and Fan, K. 2019. Lattice transformer for speech translation. *arXiv preprint arXiv:1906.05551* .

Zhou, C.; Neubig, G.; and Gu, J. 2019. Understanding knowledge distillation in non-autoregressive machine translation. *arXiv preprint arXiv:1911.02727* .

7 Appendix

7.1 Model Sizes for ST Systems

We make a detailed comparison between the performance of end-to-end systems and cascaded systems with different model parameter sizes. Compared with the original pipeline system, the pipeline (small) system is a cascade of speech recognition models and machine translation models with halved layers. Results in Table 10 prove that end-to-end models have advantages in balancing performance and the size of the total system parameters.

	En-Fr			Parameter
	ASR↓	MT↑	ST↑	
Pipeline (small)	21.3	19.5	15.9	$\approx 129M$
Pipeline	16.6	20.98	16.38	$\approx 209M$
LUT	-	-	**16.70**	$\approx 144M$

Table 10: Performance with greedy search for ASR, MT and ST tasks on En-Fr test set with different model sizes. Pipeline: consists of independently trained ASR and MT systems (see details in Section 3.3).

7.2 Attention Visualization

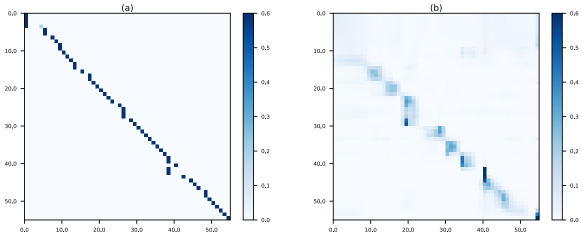

Figure 2: The visualization of attention for different module layers. (a), (b) visualize the attention of the last layer of acoustic transducer and the last layer of semantic encoder respectively. Both the horizontal and vertical coordinates represent the same sequence of speech frames.

We analyze the learned representation through visualizations of the acoustic and semantic modeling's attention between layers. Figure 2 shows an example of the distribution of attention weights. The attention of acoustic modeling is local and monotonous from the first layer to the fourth layer, matching the behavior of ASR. The attention of semantic encoder gradually tends to be smoothed out across the global context, which is beneficial to modeling semantic information. The observation is in line with our hypothesis.

7.3 Correlation Analysis

The quality of the hidden state obtained in our second stage depends largely on the accuracy of the acoustic modeling in the first stage. Using the CTC loss function introduced in the first stage, we can also predict recognition results while predicting translation results. We can diagnose whether the wrong prediction for translation is caused by the wrong acoustic modeling in this way. We use samples from the test set to analyze the relationship between translation quality and acoustic modeling, which are evaluated by BLEU and WER respectively. We draw scatter plots of WER and BLEU on the test set, as can be seen in Figure 3. It can be seen that samples with a higher WER can usually obtain a translation result with a lower BLEU. Statistically, the Pearson correlation coefficient between BLEU and WER is $-0.205 < 0$ (with p-value = $2e^{-16} << 0.05$), which indicates the significant negative relation between them. At the same time, a minority of samples with a higher WER can obtain translation results with a higher BLEU, which indicates that our ST model has a certain degree of robustness to recognition errors.

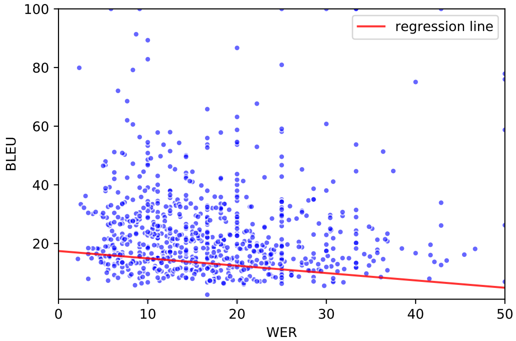

Figure 3: Relationship between WER and BLEU for En-Fr test set.

7.4 Effect of Hyper-parameters

Table 11

7.5 Shallower or Deeper

Table 12

7.6 Semi-supervised Fine-tuning Strategy

For experiments in the

0.80-0.05-0.15	17.31
0.50-0.10-0.40	17.17
0.50-0.05-0.45	**17.55**
0.50-0.01-0.49	16.87
0.40-0.20-0.40	17.00
0.30-0.40-0.30	17.34
0.20-0.05-0.75	17.41

Table 11: Performance with greedy decoding.

6-6-6	-
6-6-4	17.15
5-5-4	**17.58**
4-4-4	17.55
3-3-4	16.39
2-2-4	14.13

Table 12: Performance with greedy decoding. ”-” means failing.