DuTongChuan: Context-aware Translation Model for Simultaneous Interpreting

Abstract

In this paper, we present DuTongChuan, a novel context-aware translation model for simultaneous interpreting. This model allows to constantly read streaming text from the Automatic Speech Recognition (ASR) model and simultaneously determine the boundaries of Information Units (IUs) one after another. The detected IU is then translated into a fluent translation with two simple yet effective decoding strategies: partial decoding and context-aware decoding. In practice, by controlling the granularity of IUs and the size of the context, we can get a good trade-off between latency and translation quality easily. Elaborate evaluation from human translators reveals that our system achieves promising translation quality (85.71% for Chinese-English, and 86.36% for English-Chinese), specially in the sense of surprisingly good discourse coherence. According to an End-to-End (speech-to-speech simultaneous interpreting) evaluation, this model presents impressive performance in reducing latency (to less than 3 seconds at most times). Furthermore, we successfully deploy this model in a variety of Baidu's products which have hundreds of millions of users, and we release it as a service in our AI platform.

Introduction

Recent progress in Automatic Speech Recognition (ASR) and Neural Machine Translation (NMT), has facilitated the research on automatic speech translation with applications to live and streaming scenarios such as Simultaneous Interpreting (SI). In contrast to non-real time speech translation, simultaneous interpreting involves starting translating source speech, before the speaker finishes speaking (translating the on-going speech while listening to it). Because of this distinguishing feature, simultaneous interpreting is widely used by multilateral organizations (UN/EU), international summits (APEC/G-20), legal proceedings, and press conferences. Despite of recent advance BIBREF0 , BIBREF1 , the research on

simultaneous interpreting is notoriously difficult BIBREF0 due to well known challenging requirements: high-quality translation and low latency.

Many studies present methods to improve the translation quality by enhancing the robustness of translation model against ASR errors BIBREF2 , BIBREF3 , BIBREF4 , BIBREF5 , BIBREF6 , BIBREF7 . On the other hand, to reduce latency, some researchers propose models that start translating after reading a few source tokens BIBREF8 , BIBREF9 , BIBREF10 , BIBREF11 , BIBREF12 , BIBREF1 . As one representative work related to this topic, recently, we present a translation model using prefix-to-prefix framework with INLINEFORM0 policy BIBREF0 . This model is simple yet effective in practice, achieving impressive performance both on translation quality and latency.

However, existing work pays less attention to the fluency of translation, which is extremely important in the context of simultaneous translation. For example, we have a sub-sentence NMT model that starts to translate after reading a sub-sentence rather than waiting until the end of a sentence like the full-sentence models does. This will definitely reduce the time waiting for the source language speech. However, as shown in the Figure FIGREF2 , the translation for each sub-sentence is barely adequate, whereas the translation of the entire source sentence lacks coherence and fluency. Moreover, it is clear that the model produces an inappropriate translation “your own” for the source token “自己” due to the absence of the preceding sub-sentence.

To make the simultaneous machine translation more accessible and producible, we borrow SI strategies used by human interpreters to create our model. As shown in Figure FIGREF3 , this model is able to constantly read streaming text from the ASR model, and simultaneously determine the boundaries of Information Units (IUs) one after another. Each detected IU is then translated into a fluent translation with two simple yet effective decoding strategies: partial decoding and context-aware decoding. Specifically, IUs at the beginning of each sentence are sent to the partial decoding module. Other information units,

either appearing in the middle or at the end of a sentence, are translated into target language by the context-aware decoding module. Notice that this module is able to exploit additional context from the history so that the model can generate coherent translation. This method is derived from the “salami technique” BIBREF13 , BIBREF14 , or “chunking”, one of the most commonly used strategies by human interpreters to cope with the linearity constraint in simultaneous interpreting. Having severely limited access to source speech structure in SI, interpreters tend to slice up the incoming speech into smaller meaningful pieces that can be directly rendered or locally reformulated without having to wait for the entire sentence to unfold.

In general, there are several remarkable novel advantages that differ our model from the previous work:

For a comprehensive evaluation of our system, we use two evaluation metrics: translation quality and latency. According to the automatic evaluation metric, our system presents excellent performance both in translation quality and latency. In the speech-to-speech scenario, our model achieves an acceptability of 85.71% for Chinese-English translation, and 86.36% for English-Chinese translation in human evaluation. Moreover, the output speech lags behind the source speech by an average of less than 3 seconds, which presents surprisingly good experience for machine translation users BIBREF15 , BIBREF16 , BIBREF17 . We also ask three interpreters with SI experience to simultaneously interpret the test speech in a mock conference setting. However, the target texts transcribed from human SI obtain worse BLEU scores as the reference in the test set are actually from written translating rather than simultaneous interpreting. More importantly, when evaluated by human translators, the performance of NMT model is comparable to the professional human interpreter.

The contributions of this paper can be concluded into the following aspects:

Context-aware Translation Model

As shown in Figure FIGREF7 , our model consists of two key modules: an information unit boundary detector and a tailored NMT model. In the process of translation, the IU detector will determine the boundary for each IU while constantly reading the steaming input from the ASR model. Then, different decoding strategies are applied to translate IUs at the different positions.

In this section, we use “IU” to denote one sub-sentence for better description. But in effect, our translation model is a general solution for simultaneous interpreting, and is compatible to IUs at arbitrary granularity, i.e., clause-level, phrase-level, and word-level, etc.

For example, by treating a full-sentence as an IU, the model is reduced to the standard translation model. When the IU is one segment, it is reduced to the segment-to-segment translation model BIBREF18 , BIBREF12 . Moreover, if we treat one token as an IU, it is reduced to our previous wait-k model BIBREF0 . The key point of our model is to train the IU detector to recognize the IU boundary at the corresponding granularity.

In the remain of this section, we will introduce above two components in details.

Dynamic Context Based Information Unit Boundary Detector

Recent success on pre-training indicates that a pre-trained language representation is beneficial to downstream natural language processing tasks including classification and sequence labeling problems BIBREF19 , BIBREF20 , BIBREF21 . We thus formulate the IU boundary detection as a classification problem, and fine-tune the pre-trained model on a small size training corpus. Fine-tuned in several iterations, the model learns to recognize the boundaries of information units correctly.

As shown in Figure FIGREF13 , the model tries to predict the potential class for the current position.

Once the position is assigned to a definitely positive class, its preceding sequence is labeled as one information unit. One distinguishing feature of this model is that we allow it to wait for more context so that it can make a reliable prediction. We call this model a dynamic context based information unit boundary detector.

Definition 1 Assuming the model has already read a sequence INLINEFORM0 with INLINEFORM1 tokens, we denote INLINEFORM2 as the anchor, and the subsequence INLINEFORM3 with INLINEFORM4 tokens as dynamic context.

For example, in Figure FIGREF13 , the anchor in both cases is “姬”, and the dynamic context in the left side case is “这”, and in the right side case is “这个”.

Definition 2 If the normalized probability INLINEFORM0 for the prediction of the current anchor INLINEFORM1 is larger than a threshold INLINEFORM2 , then the sequence INLINEFORM3 is a complete sequence, and if INLINEFORM4 is smaller than a threshold INLINEFORM5 (INLINEFORM6), it is an incomplete sequence, otherwise it is an undetermined sequence.

For a complete sequence INLINEFORM0 , we will send it to the corresponding translation model . Afterwards, the detector will continue to recognize boundaries in the rest of the sequence (INLINEFORM1). For an incomplete sequence, we will take the INLINEFORM2 as the new anchor for further detection. For an undetermined sequence, which is as shown in Figure FIGREF13 , the model will wait for a new token INLINEFORM3 , and take (INLINEFORM4) as dynamic context for further prediction.

In the training stage, for one common sentence including two sub-sequences, INLINEFORM0 and INLINEFORM1 . We collect INLINEFORM2 plus any token in INLINEFORM3 as positive training

samples, and the other sub-sequences in INLINEFORM4 as negative training samples. We refer readers to Appendix for more details.

In the decoding stage, we begin with setting the size of the dynamic context to 0, and then determine whether to read more context according to the principle defined in definition SECREF15 .

Partial Decoding

Traditional NMT models are usually trained on bilingual corpora containing only complete sentences. However in our context-aware translation model, information units usually are sub-sentences. Intuitively, the discrepancy between the training and the decoding will lead to a problematic translation, if we use the conventional NMT model to translate such information units. On the other hand, conventional NMT models rarely do anticipation. Whereas in simultaneous interpreting, human interpreters often have to anticipate the up-coming input and render a constituent at the same time or even before it is uttered by the speaker.

In our previous work BIBREF0 , training a wait-k policy slightly differs from the traditional method. When predicting the first target token, we mask the source content behind the INLINEFORM0 token, in order to make the model learn to anticipate. The prediction of other tokens can also be obtained by moving the mask-window token-by-token from position INLINEFORM1 to the end of the line. According to our practical experiments, this training strategy do help the model anticipate correctly most of the time.

Following our previous work, we propose the partial decoding model, a tailored NMT model for translating the IUs that appear at the beginning of each sentence. As depicted in Figure FIGREF17 , in the training stage, we mask the second sub-sentence both in the source and target side. While translating the first sub-sentence, the model learns to anticipate the content after the comma, and produces a temporary

translation that can be further completed with more source context. Clearly, this method relies on the associated sub-sentence pairs in the training data (black text in Figure FIGREF17). In this paper, we propose an automatic method to acquire such sub-sentence pairs.

Definition 3 Given a source sentence INLINEFORM0 with INLINEFORM1 tokens, a target sentence INLINEFORM2 with INLINEFORM3 tokens, and a word alignment set INLINEFORM4 where each alignment INLINEFORM5 is a tuple indicating a word alignment existed between the source token INLINEFORM6 and target token INLINEFORM7 , a sub-sentence pair INLINEFORM8 holds if satisfying the following conditions: DISPLAYFORM0

To acquire the word alignment, we run the open source toolkit fast_align , and use a variety of standard symmetrization heuristics to generate the alignment matrix. In the training stage, we perform training by firstly tuning the model on a normal bilingual corpus, and then fine-tune the model on a special training corpus containing sub-sentence pairs.

Context-aware Decoding

For IUs that have one preceding sub-sentence, the context-aware decoding model is applied to translate them based on the pre-generated translations. The requirements of this model are obvious:

The model is required to exploit more context to continue the translation.

The model is required to generate the coherent translation given partial pre-generated translations.

Intuitively, the above requirements can be easily satisfied using a force decoding strategy. For example, when translating the second sub-sentence in “这点也是以前让我非常地诧异，也是非常纠结的地方”, given

the already-produced translation of the first sub-sentence “It also surprised me very much before .”, the model finishes the translation by adding “It's also a very surprising , tangled place .”. Clearly, translation is not that accurate and fluent with the redundant constituent “surprising”. We ascribe this to the discrepancy between training and decoding. In the training stage, the model learns to predict the translation based on the full source sentence. In the decoding stage, the source contexts for translating the first-subsentence and the second-subsentence are different. Forcing the model to generate identical translation of the first sub-sentence is very likely to cause under-translation or over-translation.

To produce more adequate and coherent translation, we make the following refinements:

During training, we force the model to focus on learning how to continue the translation without over-translation and under-translation.

During decoding, we discard a few previously generated translations, in order to make more fluent translations.

As shown in Figure FIGREF19 , during training, we do not mask the source input, instead we mask the target sequence aligned to the first sub-sentence. This strategy will force the model to learn to complete the half-way done translation, rather than to concentrate on generating a translation of the full sentence.

Moreover, in the decoding stage, as shown in Figure FIGREF28 , we propose to discard the last INLINEFORM0 tokens from the generated partial translation (at most times, discarding the last token brings promising result). Then the context-aware decoding model will complete the rest of the translation. The motivation is that the translation of the tail of a sub-sentence is largely influenced by the content of the succeeding sub-sentence. By discarding a few tokens from previously generated translation, the model is able to generate a more appropriate translation. In the practical experiment, this slight

modification is proved to be effective in generating fluent translation.

Latency Metric: Equilibrium Efficiency

In the work of DBLP:journals/corr/abs-1810-08398 and arivazhagan2019monotonic, they used the average lagging as the metric for evaluating the latency. However, there are two major flaws of this metric:

1) This metric is unsuitable for evaluating the sub-sentence model. Take the sentence in Figure FIGREF3 for example. As the model reads four tokens “她说 我 错了 那个”, and generates six target tokens “She said I was wrong ,”, the lag of the last target token is one negative value (INLINEFORM0) according to its original definition.

2) This metric is unsuitable for evaluating latency in the scenario of speech-to-speech translation. DBLP:journals/corr/abs-1810-08398 considered that the target token generated after the cut-off point doesn't cause any lag. However, this assumption is only supported in the speech-to-text scenario. In the speech-to-speech scenario, it is necessary to consider the time for playing the last synthesized speech.

Therefore, we instead propose a novel metric, Equilibrium Efficiency (EE), which measures the efficiency of equilibrium strategy.

Definition 4 Consider a sentence with INLINEFORM0 subsequences, and let INLINEFORM1 be the length of INLINEFORM2 source subsequence that emits a target subsequence with INLINEFORM3 tokens. Then the equilibrium efficiency is: INLINEFORM4 , where INLINEFORM5 is defined as: DISPLAYFORM0

and INLINEFORM0 , INLINEFORM1 is an empirical factor.

In practice, we set INLINEFORM0 to 0.3 for Chinese-English translation (reading about 200 English tokens in one minute). The motivation of EE is that one good model should equilibrate the time for playing the target speech to the time for listening to the speaker. Assuming playing one word takes one second, the EE actually measures the latency from the audience hearing the final target word to the speaker finishing the speech. For example, the EE of the sentence in Figure FIGREF7 is equal to INLINEFORM1 , since the time for playing the sequence “She said I was wrong” is equilibrated to the time for speaker speaking the second sub-sentence “那个 叫 什么 什么 呃 妖姬”.

Evaluation

We conduct multiple experiments to evaluate the effectiveness of our system in many ways.

Data Description

We use a subset of the data available for NIST OpenMT08 task . The parallel training corpus contains approximate 2 million sentence pairs. We choose NIST 2006 (NIST06) dataset as our development set, and the NIST 2002 (NIST02), 2003 (NIST03), 2004 (NIST04) 2005 (NIST05), and 2008 (NIST08) datasets as our test sets. We will use this dataset to evaluate the performance of our partial decoding and context-aware decoding strategy from the perspective of translation quality and latency.

Recently, we release Baidu Speech Translation Corpus (BSTC) for open research . This dataset covers speeches in a wide range of domains, including IT, economy, culture, biology, arts, etc. We transcribe the talks carefully, and have professional translators to produce the English translations. This procedure is extremely difficult due to the large number of domain-specific terminologies, speech redundancies and

speakers' accents. We expect that this dataset will help the researchers to develop robust NMT models on the speech translation. In summary, there are many features that distinguish this dataset to the previously related resources:

Speech irregularities are kept in transcription while omitted in translation (eg. filler words like “嗯, 呃, 啊”, and unconscious repetitions like “这个这个呢”), which can be used to evaluate the robustness of the NMT model dealing with spoken language.

Each talk's transcription is translated into English by a single translator, and then segmented into bilingual sentence pairs according to the sentence boundaries in the English translations. Therefore, every sentence is translated based on the understanding of the entire talk and is translated faithfully and coherently in global sense.

We use the streaming multi-layer truncated attention model (SMLTA) trained on the large-scale speech corpus (more than 10,000 hours) and fine-tuned on a number of talk related corpora (more than 1,000 hours), to generate the 5-best automatic recognized text for each acoustic speech.

The test dataset includes interpretations produced by simultaneous interpreters with professional experience. This dataset contributes an essential resource for the comparison between translation and interpretation.

We randomly extract several talks from the dataset, and divide them into the development and test set. In Table TABREF34 , we summarize the statistics of our dataset. The average number of utterances per talk is 152.6 in the training set, 59.75 in the dev set, and 162.5 in the test set.

We firstly run the standard Transformer model on the NIST dataset. Then we evaluate the quality of the

pre-trained model on our proposed speech translation dataset, and propose effective methods to improve the performance of the baseline. In that the testing data in this dataset contains ASR errors and speech irregularities, it can be used to evaluate the robustness of novel methods.

In the final deployment, we train our model using a corpus containing approximately 200 million bilingual pairs both in Chinese-English and English-Chinese translation tasks.

Data Preprocess

To preprocess the Chinese and the English texts, we use an open source Chinese Segmenter and Moses Tokenizer . After tokenization, we convert all English letters into lower case. And we use the “multi-bleu.pl” script to calculate BLEU scores. Except in the large-scale experiments, we conduct byte-pair encoding (BPE) BIBREF22 for both Chinese and English by setting the vocabulary size to 20K and 18K for Chinese and English, respectively. But in the large-scale experiments, we utilize a joint vocabulary for both Chinese-English and English-Chinese translation tasks, with a vocabulary size of 40K.

Model Settings

We implement our models using PaddlePaddle , an end-to-end open source deep learning platform developed by Baidu. It provides a complete suite of deep learning libraries, tools and service platforms to make the research and development of deep learning simple and reliable. For training our dynamic context sequence boundary detector, we use ERNIE BIBREF20 as our pre-trained model.

For fair comparison, we implement the following models:

baseline: A standard Transformer based model with big version of hyper parameters.

sub-sentence: We split a full sentence into multiple sub-sentences by comma, and translate them using the baseline model. To evaluate the translation quality, we concatenate the translation of each sub-sentence into one sentence.

wait-k: This is our previous work BIBREF0 .

context-aware: This is our proposed model using context-aware decoding strategy, without fine-tuning on partial decoding model.

partial decoding: This is our proposed model using partial decoding.

discard INLINEFORM0 tokens: The previously generated INLINEFORM1 tokens are removed to complete the rest of the translation by the context-aware decoding model.

Experiments

We firstly conduct our experiments on the NIST Chinese-English translation task.

To validate the effectiveness of our translation model, we run two baseline models, baseline and sub-sentence. We also compare the translation quality as well as latency of our models with the wait-k model.

Effectiveness on Translation Quality. As shown in Table TABREF49 , there is a great deal of difference between the sub-sentence and the baseline model. On an average the sub-sentence shows weaker

performance by a 3.08 drop in BLEU score (40.39 INLINEFORM0 37.31). Similarly, the wait-k model also brings an obvious decrease in translation quality, even with the best wait-15 policy, its performance is still worse than the baseline system, with a 2.15 drop, averagely, in BLEU (40.39 INLINEFORM1 38.24). For a machine translation product, a large degradation in translation quality will largely affect the use experience even if it has low latency.

Unsurprisingly, when treating sub-sentences as IUs, our proposed model significantly improves the translation quality by an average of 2.35 increase in BLEU score (37.31 INLINEFORM0 39.66), and its performance is slightly lower than the baseline system with a 0.73 lower average BLEU score (40.39 INLINEFORM1 39.66). Moreover, as we allow the model to discard a few previously generated tokens, the performance can be further improved to 39.82 (INLINEFORM2 0.16), at a small cost of longer latency (see Figure FIGREF58). It is consistent with our intuition that our novel partial decoding strategy can bring stable improvement on each testing dataset. It achieves an average improvement of 0.44 BLEU score (39.22 INLINEFORM3 39.66) compared to the context-aware system in which we do not fine-tune the trained model when using partial decoding strategy. An interesting finding is that our translation model performs better than the baseline system on the NIST08 testing set. We analyze the translation results and find that the sentences in NIST08 are extremely long, which affect the standard Transformer to learn better representation BIBREF23 . Using context-aware decoding strategy to generate consistent and coherent translation, our model performs better by focusing on generating translation for relatively shorter sub-sentences.

Investigation on Decoding Based on Segment. Intuitively, treating one segment as an IU will reduce the latency in waiting for more input to come. Therefore, we split the testing data into segments according to the principle in Definition SECREF20 (if INLINEFORM0 in Definition SECREF20 is a comma, then the data is sub-sentence pair, otherwise it is a segment-pair.) .

As Table TABREF49 shows, although the translation quality of discard 1 token based on segment is worse than that based on sub-sentence (37.96 vs. 39.66), the performance can be significantly improved by allowing the model discarding more previously generated tokens. Lastly, the discard 6 tokens obtains an impressive result, with an average improvement of 1.76 BLEU score (37.96 INLINEFORM0 39.72).

Effects of Discarding Preceding Generated Tokens. As mentioned and depicted in Figure FIGREF28 , we discard one token in the previously generated translation in our context-aware NMT model. One may be interested in whether discarding more generated translation leads to better translation quality. However, when decoding on the sub-sentence, even the best discard 4 tokens model brings no significant improvement (39.66 INLINEFORM0 39.82) but a slight cost of latency (see in Figure FIGREF58 for visualized latency). While decoding on the segment, even discarding two tokens can bring significant improvement (37.96 INLINEFORM1 39.00). This finding proves that our partial decoding model is able to generate accurate translation by anticipating the future content. It also indicates that the anticipation based on a larger context presents more robust performance than the aggressive anticipation in the wait-k model, as well as in the segment based decoding model.

Effectiveness on latency. As latency in simultaneous machine translation is essential and is worth to be intensively investigated, we compare the latency of our models with that of the previous work using our Equilibrium Efficiency metric. As shown in Figure FIGREF58 , we plot the translation quality and INLINEFORM0 on the NIST06 dev set. Clearly, compared to the baseline system, our model significantly reduce the time delay while remains a competitive translation quality. When treating segments as IUs, the latency can be further reduced by approximate 20% (23.13 INLINEFORM1 18.65), with a slight decrease in BLEU score (47.61 INLINEFORM2 47.27). One interesting finding is that the granularity of information units largely affects both the translation quality and latency. It is clear the decoding based on sub-sentence and based on segment present different performance in two metrics. For the former model, the increase of discarded tokens results in an obvious decrease in translation quality, but no definite

improvement in latency. The latter model can benefit from the increasing of discarding tokens both in translation quality and latency.

The latency of the wait-k models are competitive, their translation quality, however, is still worse than context-aware model. Improving the translation quality for the wait-k will clearly brings a large cost of latency (36.53 INLINEFORM0 46.14 vs. 10.94 INLINEFORM1 22.63). Even with a best k-20 policy, its performance is still worse than most context-aware models. More importantly, the intermediately generated target token in the wait-k policy is unsuitable for TTS due to the fact that the generated token is often a unit in BPE, typically is an incomplete word. One can certainly wait more target tokens to synthesize the target speech, however, this method will reduce to the baseline model. In general, experienced human interpreters lag approximately 5 seconds (15 INLINEFORM2 25 words) behind the speaker BIBREF15 , BIBREF16 , BIBREF17 , which indicates that the latency of our model is accessible and practicable (INLINEFORM3 = 25 indicates lagging 25 words).

In our context-sensitive model, the dynamic context based information unit boundary detector is essential to determine the IU boundaries in the steaming input. To measure the effectiveness of this model, we compare its precision as well as latency against the traditional language model based methods, a 5-gram language model trained by KenLM toolkit , and an in-house implemented RNN based model. Both of two contrastive models are trained on approximate 2 million monolingual Chinese sentences. As shown in Table TABREF60 , it is clear that our model beats the previous work with an absolute improvement of more than 15 points in term of F-score (62.79 INLINEFORM0 78.26) and no obvious burden in latency (average latency). This observation indicates that with bidirectional context, the model can learn better representation to help the downstream tasks. In the next experiments, we will evaluate models given testing data with IU boundaries detected by our detector.

To our knowledge, almost all of the previous related work on simultaneous translation evaluate their

models upon the clean testing data without ASR errors and with explicit sentence boundaries annotated by human translators. Certainly, testing data with real ASR errors and without explicit sentence boundaries is beneficial to evaluate the robustness of translation models. To this end, we perform experiments on our proposed BSTC dataset.

The testing data in BSTC corpus consists of six talks. We firstly employ our ASR model to recognize the acoustic waves into Chinese text, which will be further segmented into small pieces of sub-sentences by our IU detector. To evaluate the contribution of our proposed BSTC dataset, we firstly train all models on the NIST dataset, and then check whether the performance can be further improved by fine-tuning them on the BSTC dataset.

From the results shown in Table TABREF64 , we conclude the following observations:

Due to the relatively lower CER in ASR errors (10.32 %), the distinction between the clean input and the noisy input results in a BLEU score difference smaller than 2 points (15.85 vs. 14.60 for pre-train, and 21.98 vs. 19.91 for fine-tune).

Despite the small size of the training data in BSTC, fine-tuning on this data is essential to improve the performance of all models.

In all settings, the best system in context-aware model beats the wait-15 model.

Pre-trained models are not sensitive to errors from Auto IU, while fine-tuned models are.

Another interesting work is to compare machine translation with human interpretation. We request three simultaneous interpreters (S, A and B) with years of interpreting experience ranging from three to seven

years, to interpret the talks in BSTC testing dataset, in a mock conference setting .

We concatenate the translation of each talk into one big sentence, and then evaluate it by BLEU score. From Table TABREF69 , we find that machine translation beats the human interpreters significantly. Moreover, the length of interpretations are relatively short, and results in a high length penalty provided by the evaluation script. The result is unsurprising, because human interpreters often deliberately skip non-primary information to keep a reasonable ear-voice span, which may bring a loss of adequacy and yet a shorter lag time, whereas the machine translation model translates the content adequately. We also use human interpreting results as references. As Table TABREF69 indicates, our model achieves a higher BLEU score, 28.08.

Furthermore, we ask human translators to evaluate the quality between interpreting and machine translation. To evaluate the performance of our final system, we select one Chinese talk as well as one English talk consisting of about 110 sentences, and have human translators to assess the translation from multiple aspects: adequacy, fluency and correctness. The detailed measurements are:

Bad: Typically, the mark Bad indicates that the translation is incorrect and unacceptable.

OK: If a translation is comprehensible and adequate, but with minor errors such as incorrect function words and less fluent phrases, then it will be marked as OK.

Good: A translation will be marked as Good if it contains no obvious errors.

As shown in Table TABREF70 , the performance of our model is comparable to the interpreting. It is worth mentioning that both automatic and human evaluation criteria are designed for evaluating written translation and have a special emphasis on adequacy and faithfulness. But in simultaneous interpreting,

human interpreters routinely omit less-important information to overcome their limitations in working memory. As the last column in Table 6 shows, human interpreters' oral translations have more omissions than machine's and receive lower acceptability. The evaluation results do not mean that machines have exceeded human interpreters in simultaneous interpreting. Instead, it means we need machine translation criteria that suit simultaneous interpreting. We also find that the BSTC dataset is extremely difficult as the best human interpreter obtains a lower Acceptability 73.04%. Although the NMT model obtains impressive translation quality, we do not compare the latency of machine translation and human interpreting in this paper, and leave it to the future work.

To better understand the contribution of our model on generating coherent translation, we select one representative running example for analysis. As the red text in Figure FIGREF73 demonstrates that machine translation model generates coherent translation “its own grid” for the sub-sentence “这个网络”, and “corresponds actually to” for the subsequence “...对应的,就是每个...”. Compared to the human interpretation, our model presents comparable translation quality. In details, our model treats segments as IUs, and generates translation for each IU consecutively. While the human interpreter splits the entire source text into two sub-sentences, and produces the translation respectively.

In the final deployment, we train DuTongChuan on the large-scale training corpus. We also utilize techniques to enhance the robustness of the translation model, such as normalization of the speech irregularities, dealing with abnormal ASR errors, and content censorship, etc (see Appendix). We successfully deploy DuTongChuan in the Baidu Create 2019 (Baidu AI Developer Conference) .

As shown in Table TABREF74 , it is clear that DuTongChuan achieves promising acceptability on both translation tasks (85.71% for Chinese-English, and 86.36 % for English-Chinese). We also elaborately analyze the error types in the final translations, and we find that apart from errors occurring in translation and ASR, a majority of errors come from IU boundary detection, which account for nearly a half of errors.

In the future, we should concentrate on improving the translation quality by enhancing the robustness of our IU boundary detector. We also evaluate the latency of our model in an End-to-End manner (speech-to-speech), and we find that the target speech slightly lags behind the source speech in less than 3 seconds at most times. The overall performance both on translation quality and latency reveals that DuTongChuan is accessible and practicable in an industrial scenario.

Related Work

The existing research on speech translation can be divided into two types: the End-to-End model BIBREF24 , BIBREF25 , BIBREF26 , BIBREF27 , BIBREF28 and the cascaded model. The former approach directly translates the acoustic speech in one language, into text in another language without generating the intermediate transcription for the source language. Depending on the complexity of the translation task as well as the scarce training data, previous literatures explore effective techniques to boost the performance. For example pre-training BIBREF29 , multi-task learning BIBREF24 , BIBREF27 , attention-passing, BIBREF30 , and knowledge distillation BIBREF28 etc.,. However, the cascaded model remains the dominant approach and presents superior performance practically, since the ASR and NMT model can be optimized separately training on the large-scale corpus.

Many studies have proposed to synthesize realistic ASR errors, and augment them with translation training data, to enhance the robustness of the NMT model towards ASR errors BIBREF2 , BIBREF3 , BIBREF4 . However, most of these approaches depend on simple heuristic rules and only evaluate on artificially noisy test set, which do not always reflect the real noises distribution on training and inference BIBREF5 , BIBREF6 , BIBREF7 .

Beyond the research on translation models, there are many research on the other relevant problems, such as sentence boundary detection for realtime speech translation BIBREF31 , BIBREF18 , BIBREF32

, BIBREF33 , BIBREF34 , low-latency simultaneous interpreting BIBREF8 , BIBREF9 , BIBREF10 , BIBREF11 , BIBREF12 , BIBREF35 , BIBREF36 , automatic punctuation annotation for speech transcription BIBREF37 , BIBREF38 , and discussion about human and machine in simultaneous interpreting BIBREF39 .

Focus on the simultaneous translation task, there are some work referring to the construction of the simultaneous interpreting corpus BIBREF40 , BIBREF41 , BIBREF42 . Particularly, BIBREF42 deliver a collection of a simultaneous translation corpus for comparative analysis on Japanese-English and English-Japanese speech translation. This work analyze the difference between the translation and the interpretations, using the interpretations from human simultaneous interpreters.

For better generation of coherent translations, gong2011cache propose a memory based approach to capture contextual information to make the statistical translation model generate discourse coherent translations. kuang2017cache,tu2018learning,P18-1118 extend similar memory based approach to the NMT framework. wang2017exploiting present a novel document RNN to learn the representation of the entire text, and treated the external context as the auxiliary context which will be retrieved by the hidden state in the decoder. tiedemann2017neural and P18-1117 propose to encode global context through extending the current sentence with one preceding adjacent sentence. Notably, the former is conducted on the recurrent based models while the latter is implemented on the Transformer model. Recently, we also propose a reinforcement learning strategy to deliberate the translation so that the model can generate more coherent translations BIBREF43 .

Conclusion and Future Work

In this paper, we propose DuTongChuan, a novel context-aware translation model for simultaneous interpreting. This model is able to constantly read streaming text from the ASR model, and simultaneously

determine the boundaries of information units one after another. The detected IU is then translated into a fluent translation with two simple yet effective decoding strategies: partial decoding and context-aware decoding. We also release a novel speech translation corpus, BSTC, to boost the research on robust speech translation task.

With elaborate comparison, our model obtains superior translation quality against the wait-k model, but also presents competitive performance in latency. Assessment from human translators reveals that our system achieves promising translation quality (85.71% for Chinese-English, and 86.36% for English-Chinese), specially in the sense of surprisingly good discourse coherence. Our system also presents superior performance in latency (delayed in less 3 seconds at most times) in a speech-to-speech simultaneous translation. We also deploy our simultaneous machine translation model in our AI platform, and welcome the other users to enjoy it.

In the future, we will conduct research on novel method to evaluate the interpreting.

Acknowledgement

We thank Ying Chen for improving the written of this paper. We thank Yutao Qu for developing partial modules of DuTongChuan. We thank colleagues in Baidu for their efforts on construction of the BSTC. They are Zhi Li, Ying Chen, Xuesi Song, Na Chen, Qingfei Li, Xin Hua, Can Jin, Lin Su, Lin Gao, Yang Luo, Xing Wan, Qiaoqiao She, Jingxuan Zhao, Can Jin, Wei Jin, Xiao Yang, Shuo Liu, Yang Zhang, Jing Ma, Junjin Zhao, Yan Xie, Minyang Zhang, Niandong Du, etc.

We also thank tndao.com and zaojiu.com for contributing their speech corpora.

Training Samples for Information Unit Detector

For example, for a sentence “她说我错了，那个叫什么什么呃妖姬。”, there are some representative training samples:

Techniques for Robust Translation

To develop an industrial simultaneous machine translation system, it is necessary to deal with problems that affect the translation quality in practice such as large number of speech irregularities, ASR errors, and topics that allude to violence, religion, sex and politics.

Speech Irregularities Normalization

In the real talk, the speaker tends to express his opinion using irregularities rather than regular written language utilized to train prevalent machine translation relevant models. For example, as depicted in Figure FIGREF3 , the spoken language in the real talk often contains unconscious repetitions (i.e., “什么(shénmē) 什么(shénmē)), and filler words (“呃”, “啊”), which will inevitably affects the downstream models, especially the NMT model. The discrepancy between training and decoding is not only existed in the corpus, but also occurs due to the error propagation from ASR model (e.g. recognize the “饿 (è)” into filler word “呃 (è) ” erroneously), which is related to the field of robust speech NMT research.

In the study of robust speech translation, there are many methods can be applied to alleviate the discrepancy mostly arising from the ASR errors such as disfluency detection, fine-tuning on the noisy training data BIBREF2 , BIBREF3 , complex lattice input BIBREF4 , etc. For spoken language normalization, it is mostly related to the work of sentence simplification. However, the traditional methods for sentence simplification rely large-scale training corpus and will enhance the model complexity by incorporating an End-to-End model to transform the original input.

In our system, to resolve problems both on speech irregularities and ASR errors, we propose a simple rule heuristic method to normalize both spoken language and ASR errors, mostly focus on removing noisy inputs, including filler words, unconscious repetitions, and ASR error that is easy to be detected. Although faithfulness and adequacy is essential in the period of the simultaneous interpreting, however, in a conference, users can understand the majority of the content by discarding some unimportant words.

To remove unconscious repetitions, the problem can be formulated as the Longest Continuous Substring (LCS) problem, which can be solved by an efficient suffix-array based algorithm in INLINEFORM0 time complexity empirically. Unfortunately, this simple solution is problematic in some cases. For example, “他 必须 分成 很多 个 小格 ， 一个 小格 一个 小格 完成”, in this case, the unconscious repetitions “一个 小格 一个 小格” can not be normalized to “一个 小格”. To resolve this drawback, we collect unconscious repetitions appearing more than 5 times in a large-scale corpus consisting of written expressions, resulting in a white list containing more than 7,000 unconscious repetitions. In practice, we will firstly retrieve this white list and prevent the candidates existed in it from being normalized.

According to our previous study, many ASR errors are caused by disambiguating homophone. In some cases, such error will lead to serious problem. For example, both “食油 (cooking oil)” and “石油 (oil)” have similar Chinese phonetic alphabet (shí yóu), but with distinct semantics. The simplest method to resolve this problem is to enhance the ASR model by utilizing a domain-specific language model to generate the correct sequence. However, this method requires an insatiably difficult requirement, a customized ASR model. To reduce the cost of deploying a customized ASR model, as well as to alleviate the propagation of ASR errors, we propose a language model based identifier to remove the abnormal contents.

Definition 5 For a given sequence INLINEFORM0 , if the value of INLINEFORM1 is lower than a threshold INLINEFORM2 , then we denote the token INLINEFORM3 as an abnormal content.

In the above definition, the value of INLINEFORM0 and INLINEFORM1 can be efficiently computed by a language model. In our final system, we firstly train a language model on the domain-specific monolingual corpus, and then identify the abnormal content before the context-aware translation model. For the detected abnormal content, we simply discard it rather than finding an alternative, which will lead to additional errors potentially. Actually, human interpreters often routinely omit source content due to the limited memory.

Constrained Decoding and Content Censorship

For an industrial product, it is extremely important to control the content that will be presented to the audience. Additionally, it is also important to make a consistent translation for the domain-specific entities and terminologies. This two demands lead to two associate problems: content censorship and constrained decoding, where the former aims to avoid producing some translation while the latter has the opposite target, generating pre-specified translation.

Recently, post2018fast proposed a Dynamic Beam Allocation (DBA) strategy, a beam search algorithm that forces the inclusion of pre-specified words and phrases in the output. In the DBA strategy, there are many manually annotated constraints, to force the beam search generating the pre-specified translation. To satisfy the requirement of content censorship, we extend this algorithm to prevent the model from generating the pre-specified forbidden content, a collection that contains words and phrases alluding to violence, religion, sex and politics. Specially, during the beam search, we punish the candidate beam that matches a constraint of pre-specified forbidden content, to prevent it from being selected as the final translation.