Spoken Translation

Adarsh Jaju, Preethi Jyothi, and Pushpak Bhattacharyya
Indian Institute of Technology, Bombay
{adarshj, pjyothi, pb} @cse.iitb.ac.in

Abstract

Spoken language translation system is receiving a lot of attention these days. It enables in translation of speech signals in a source language A to text in target language B. This problem mainly deals with Machine translation (MT), Automatic Speech Recognition (ASR), Machine Learning (ML) and End to End models. There are two approaches to perform this task. First is the standard traditional pipeline. The spoken utterances are first recognized and converted to text in source language and later this source language text is translated to target language. In the second model end to end models are trained to directly translate speech in source language to text in target language. In this paper, we start with looking into the whole flow of speech translation by going via Automatic Speech Recognition and its techniques and neural machine translation. We study the coupling of speech recognition system and the machine translation system. The end to end models are discussed along with network architectures followed by some multitask networks to effectively make use of the auxiliary data.

1 Introduction

Spoken language translation is the process by which conversational spoken phrases are converted to second language. This enables the speakers of different languages to communicate. Translation is a process of changing the language that is written or spoken in one language to another language. Without changing the meaning of the language. The speech translation system integrates two technologies : Automatic Speech Recognition, Machine Translation. The speaker of language A speaks and the speech recognizer recognizes the utterance. The input is then converted into a string of words, using dictionary and grammar of language A, by using the massive corpus of text of language A. The machine translation part takes care of translating the text into another language. In end to end translation model directly translates speech in language to text in another language. In this paper, we will first look into the ASR approaches, NMT approaches and the coupling of the systems.

2 Automatic Speech Recognition

Speech recognition is an inter-disciplinary field of computational linguistics that develops method for recognition and translation of spoken language to text. Speech recognition applications include voice user interfaces such as voice dialing, call routing, simple data entry, preparation of structured documents, speech-to-text processing. Some Sr systems use training where individual speaker reads text or isolated vocabulary into the system. The system analyzes the persons specific voice and uses it to fine-tune the recognition of that persons speech, resulting in increased accuracy. Such SR systems that use the training are called speaker dependent else it is called speaker independent systems. The term speaker identification refers to identifying the speaker, rather than what they are saying. The voice of any person can be translated and stored as data on which we can train persons voice and it will be useful for speaker identification, helpful for security purposes. ASR uses mainly two models acoustic model and language model to recognize the speech. Acoustic model gives a relationship between the phonemes and the audio signals. Language model gives idea of identifying correct words from the data looking at the context as well. There are few methods like HMM, Neural networks(Peddinti et al., 2015) etc that are used in speech recognition.

3 Neural Machine Translation

Neural machine translation(Verma and Bhattacharyya) is end-to-end translation process for automated translation and is designed to remove all the weaknesses that was because of the phrase based machine translation(Chu et al., 2017). NMT is an asset as it has ability to learn directly, as end-to-end sequence and mapping the input sequence to the output sequence. NMT generally consists of two RNNs, with one RNN taking input text sequence and the other one giving the output sequence. NMT can be made efficient by making use of attention.

4 Attention Mechanisms

The need of attention mechanism is to find a way to associate the decoder state and the every input word. Based on this result we can find the input words that how important are these to get the output words or we can set the weights accordingly.(Luong et al., 2015)

4.1 Global Attention

In case of Global attention, while predicting the next target word, each source word is taken into consideration. Each source sentence word is assigned some weight which shows the importance of that word.

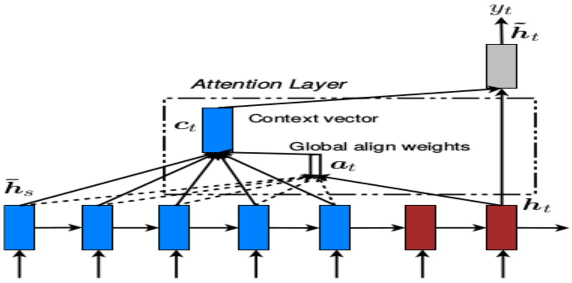


Figure 1: Global Attention Architecture

4.2 Local Attention

In case of Local attention, while predicting the next target word, we first predict a main source word aligned to the target word. Now words near this is given more importance. Words which are away from this aligned word is not given any weight. The weights are assigned using a Gaussian function.

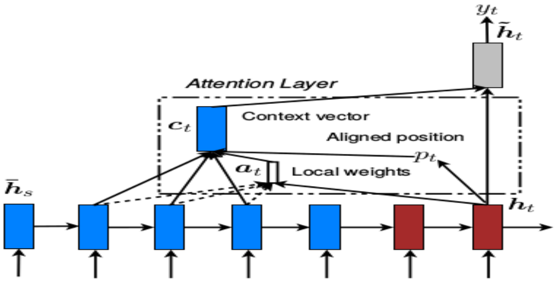


Figure 2: Local Attention Architecture

5 Coupling of ASR and MT

Speech translation is conventionally carried out by cascading an **Automatic Speech Recognition System** and **Machine Translation system**. Generally the factors that are optimized are the language models and the acoustic models along with the **word error rate** for the ASR system and the **bleu** score for the MT system. The process of spoken translation is a three step pipeline. Step one involves transcribing the speech to the text format using ASR system. Step two ensures that the output from the ASR is in the format in which the MT system expects the input to be in. The third step includes the SMT part which translates the text from source language to the target language. The basic case will be to use the ASR 1-best output that can be used as an input by the MT system. The other output options from the ASR system that can be fed to the MT system are **N-best** or **lattices** and **confusion networks**. These can be useful for the tuning and decoding in the MT system, however, it increases the complexity because of the number of alternatives present are exponential(Collins, 2002).

5.1 Maximum Spanning Phrases Model

In SMT phrase base translation, the translation is produced for each possible span of input sentence as allowed by the phrase table. If the **phrases** are **longer**, it gives **lesser options** and the translation is reliable as there may be sufficient occurrences of the phrases in the model. So, the longer the phrases, better the translation. We want the **ASR hypothesis** that requires least number of hypothesis to cover. We use the phrase lattice which is the composition of word lattice and phrase segmentation transducer for achieving this. The phrase lattice use the weights from the phrase segmentation transducer and these weights are the number of phrases used to cover the path. The shortest path will give us the phrase we were looking for. Thus, this feature of SMT of source path length can be used for the phrase selection.

5.2 Training and Feature selection

The training of the hypothesis selection is based on the standard methods of log linear model on the held-out set. For this we decode the **N-Best** derived from the held out set. Our main **objective** is to **maximize the translation quality** on the basis on some sentence level scores. Each time we get the translation, it can be compared to the N-Best and whenever the weights are updated it tells us how much importance needs to be given to the ASR and MT. The following features can be used for the model :

1. **ASR Scores** : We combine the ASR acoustic model and the language model scores as the combined feature.

   $$f_{ASR} = LM + \alpha AM \tag{1}$$

   where the AM and the LM are the negative log probabilities and α is the acoustic scaling factor chosen to minimize the word error rate.

2. **Source Phrase Count** : This feature gives the intuition that using the fewer number of phrases for covering the input sentence will give better result.

3. **Length normalized phrase uni-gram probability** : We can use a phrase Language Model feature using the n-gram probabilities normalized by the length.

   $$f_{uni}(f_j) = \left[\frac{count(f_j)}{\sum_k count(f_k)}\right]^{len(f+i)} \tag{2}$$

4. **Phrase Translation Entropy** : For each of the source phrase p_j there can be many translations e_i with different translation probabilities($P(e_i/f_j)$). A simple metric to get the correct translation will be to use the entropy measure to get the confidence of which translation is the best for the SMT.

   $$H_{tr}(E|p_j) = -\sum_i p_{tr}(e_i|f_j)log(p_{tr}(e_i|f_j)) \tag{3}$$

In this section we look into the coupling of the ASR and the MT system by fetching the output from the ASR system and giving it as input to the MT system and hence coupling them. We have considered the coarse to fine speech translation. We look into few featurized models for hypothesis selection. Also, we looked into the maximum spanning phrases model and also describe the training and feature selection. We wind up by looking into the decoding and few related techniques.

6 Sub-Word Based Translation

Sub-word based translation address the problem of sparsity using a fixed size vocabulary. The paper(Sennrich et al., 2015) discusses the algorithms and effectiveness using Byte Pair Encoding as a sub word unit. The main idea is to identify frequent consecutive set of bytes in the corpus and replace these frequent bytes with a new byte which is not in the data-set, and this continues. This is also called as the merge operations. The words that we consider while calculating the frequent bytes are termed as work list. Morpheme based translation technique also plays an important role where a word is broken down into its root word along with a suffix. These individual words are called as morphemes. Morpheme based translation performs better in language pairs involving related languages. In contrast to Byte Pair Encoding which perform good in general.

7 Attention Passing End to End Models

The paper(Sperber et al., 2019) describes how to efficiently use auxiliary data for training end to end models, and shows that direct speech translation models requires more data to perform better than the cascaded models. The paper comes with a novel architecture to train end to end model.

7.1 Model Architecture

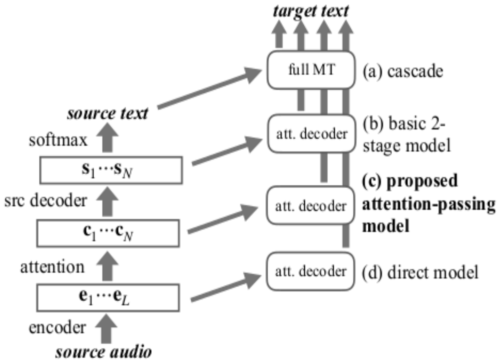


Figure 3: Architecture

The first part is the cascaded model where the speech is translated to text in the source language and then this is given as an input to the Machine Translation model. The last part is the standard end to end model where the speech in source language is directly translated to text in target language. It is found that this model does not effectively makes the use of auxiliary data to train the model parameters. The second part is the two stage model where the first decoder states are passed as an input to the second decoder. It effectively used the auxiliary data to train the model parameters. But the disadvantage is the erroneous decoder states are passed as input to the second decoder. To avoid that the paper proposes a model where the context vectors are passed as an input to the second decoder instead of first decoder states because the context vectors are not affected by the erroneous decoder states.

7.2 Using Auxiliary Data

It is seen that generally cascaded models outperforms end to end models in Spoken translation task. But given more data, multitask setup helps end to end models to perform better than cascaded models. In the given architecture, the Spoken Translation task uses the source speech encoder and target text decoder for training the model. The Automatic Speech Recognition task uses source speech encoder and source text decoder parameters for training. And the MT task uses the source text encoder and target text decoder parameters for training. Overall the ASR and ST task shares the speech encoder and the ST and MT task shares the target text decoder. Auto encoders are also used to train the parameters for source text encoder and source text decoder. The multitask training architecture is slightly different for other models. But in other models are the parameters trained in multitask training are used compared to direct end to end model where only half of the parameters are used.

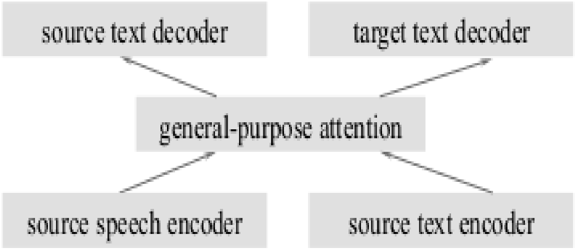


Figure 4: Multitask Setup

7.3 Results and Analysis

The paper shows that given enough data the third architecture (the one that involves context vectors) outperforms all the remaining architectures and it also effectively makes use of the auxiliary data to train the model parameters. The same architecture is also robust against the errors encountered in the first decoder states. This was checked by forcing the first decoder sates to predict output with some percentage of errors incorporated in it. As the context vectors are robust against the errors made by decoder states, there is no propagation of error to the next decoder states.

8 Tied Multitask Learning for Neural Speech Translation

The paper(Anastasopoulos and Chiang, 2018) introduces two intuitive notions to train the ST model. First is that in a two decoder architecture to achieve the task the second decoder receives input from both the first decoder and the encoder, this way we can provide high level intermediate information to the second decoder. And concepts such as transitivity and invertibility are achieved with the help of regularization over the Attention matrices.

8.1 Architectures

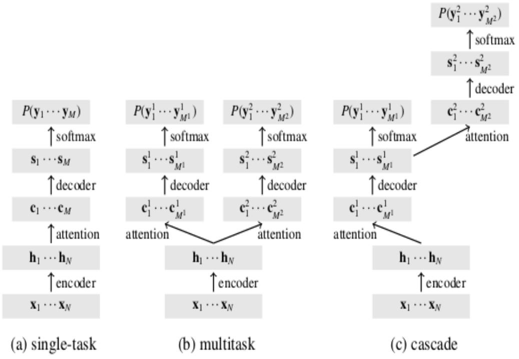
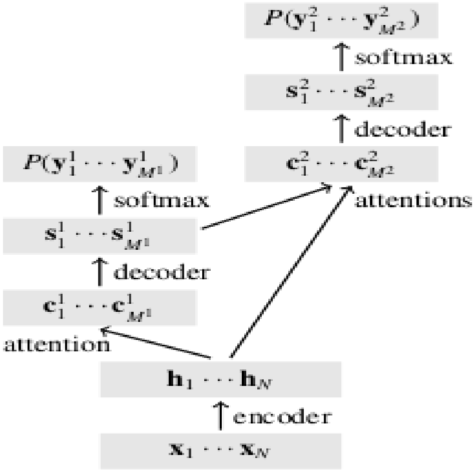


Figure 5: Architectures

The above figure shows two standard architectures. The left one represents single task architecture where the input is passed to encoder, and then the encoder states are passed to attention mechanism. The context vectors are finally passed to the decoder which predicts the output. The can be modelled to achieve a single task. The middle one represents the multitask architecture, where a encoder is shared between two tasks, and then there are two separate decoders to achieve the task. For example, one task be ASR and second one ST, so the source speech encoder is shares among both the tasks, one decoder is used to predict the source text transcriptions and the other decoder is used to predict target text translations. The third architecture is a cascaded model. This contains a encoder and two decoders. The first decoder receives input from the encoder which outputs target sequence corresponding to first task. The decoder states of the first decoder as passed as an input to the second decoder which predicts the target sequence corresponding to the second task.

8.2 The Triangle Architecture



$P(\mathbf{y}_1^2 \cdots \mathbf{y}_{M^2}^2)$
softmax
$\mathbf{s}_1^2 \cdots \mathbf{s}_{M^2}^2$
decoder
$\mathbf{c}_1^2 \cdots \mathbf{c}_{M^2}^2$
attentions
$P(\mathbf{y}_1^1 \cdots \mathbf{y}_{M^1}^1)$
softmax
$\mathbf{s}_1^1 \cdots \mathbf{s}_{M^1}^1$
decoder
$\mathbf{c}_1^1 \cdots \mathbf{c}_{M^1}^1$
attention
$\mathbf{h}_1 \cdots \mathbf{h}_N$
encoder
$\mathbf{x}_1 \cdots \mathbf{x}_N$

Figure 6: Architecture

The figure shows their proposed triangle model. There is an encoder and two decoders. The first decoder receives input only from the encoder. Whereas the second decoder receives input form both the encoder and the second decoder.This way it gets high level intermediate information from the first decoder. The inputs from both are concatenated and then passed to the second decoder. The results shows that the triangle model outperforms other models described above along with cascaded models especially for low resource speech translation. But the model shows negative results for high resource text translation.

9 Conclusion

In this paper, we presented the survey on the Automatic Speech Recognition approaches, Neural Machine Translation and its improvements and work on the integration of speech recognition and machine translation. We did it using N-Best, word lattice and confusion networks decoding. We also looked into the end to end models and different multitask networks that can be used to trained the spoken translation models. These are good at making use of auxiliary data. Spoken Language Translation research has flourished significantly in the past few years, necessitating a look-back at the overall picture that these individual works have led to. Based on the survey, we find that lot of research has been done on the ASR side, NMT side and their coupling and hence these three can be combined to form a complete spoken translation system and also the end to end models.

References

Antonios Anastasopoulos and David Chiang. 2018. Tied multitask learning for neural speech translation. *arXiv preprint arXiv:1802.06655*.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. 2019. Attention-passing models for robust and data-efficient end-to-end speech translation. *arXiv preprint arXiv:1904.07209*.

Ajay Anand Verma and Pushpak Bhattacharyya. Literature survey: Neural machine translation.