NAIST-IS-DD1561205

Doctoral Dissertation

Emphasis Speech-to-Speech Translation Considering Acoustic and Linguistic Features

Do Quoc Truong

July 30, 2018

Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Do Quoc Truong

Thesis Committee:

Professor Satoshi Nakamura	(Supervisor)
Professor Yuji Matsumoto	(Co-supervisor)
Associate Professor Sakriani Sakti	(Co-supervisor)
Dr. Peter Bell	(The University of Edinburgh)

Emphasis Speech-to-Speech Translation Considering Acoustic and Linguistic Features*

Do Quoc Truong

Abstract

Speech-to-speech translation (S2ST) systems are capable of breaking language barriers in cross-lingual communication by translating speech across languages. Recent studies have introduced many improvements that allow existing S2ST systems to handle not only linguistic meaning but also paralinguistic information such as emphasis by proposing additional emphasis estimation and translation components. However, there are still problems remaining. First, existing emphasis modeling techniques assume emphasis speech is expressed at word-level with binary values indicating the change of acoustic feature. However, depending on the context and situation, emphasis can be expressed at arbitrary levels. This assumtion also limit the capability of the model in the way that it can only generate binary emphasized speech. Second, the existing emphasis S2ST approaches used for emphasis translation is not optimal for sequence translation tasks and cannot easily handle the long-term dependencies of words and emphasis levels. Third, the whole translation pipeline still separates emphasis and standard S2ST systems, making it not possible to perform joint optimization and inference. And finally, only binary levels of acoustic feature (emphasis speech) is taken into account while emphasis can be expressed in many ways including written form at arbitrary levels. This problem limits the capable of emphasis S2ST system that it can only translate acoustic features but not linguistic features of emphasis.

This thesis attempts to solve the problems above by (a) proposing an approach that can handle continuous emphasis levels in both emphasis modeling

*Doctoral Dissertation, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD1561205, July 30, 2018.

and translation components, and (b), combining machine and emphasis translation into a single model, which greatly simplifies the translation pipeline and make it easier to perform joint optimization. And finally, (c), we propose a data-driven approach on studying correlation of emphasis expressed in both text and speech as a first step toward acoustic-linguistic emphasis translation.

With regards to the experiments, the results on continuous emphasis modeling demonstrated its effectiveness in a emphasis detection task while producing more natural synthetic speech. Experiments on an emphasis translation task utilizing sequence-to-sequence approach with continous emphasis levels show a significant improvement over previous models in both objective and subjective tests. Moreover, the evaluation on joint translation model also show that our models can jointly translate words and emphasis with one-word delays instead of full-sentence delays while preserving the translation performance of both tasks. Finally, our studies on emphasis representations in both audio and text forms have investigated the way humans express emphasis in different contexts and analyzed ambiguities between emphasis levels.

Keywords:

Emphasis estimation, emphasis translation, speech-to-speech translation, joint optimization of words and emphasis

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Human-to-Human Multilingual Speech Interaction

1.1.1 Human languages

Languages help people to communicate and share knowledge to each other, they are also the main factors that contribute to the richness and diversity and of cultures. According to Harald [1] there are about 6,500 spoken languages all over the world as illustrated in Fig. 1.1, and although this number is approximated, it showed us the impressiveness of the globe's linguistic diversity. There are two type of languages, written and spoken languages, and they are different in many ways. The written language tends to be more complex than spoken with longer and more clauses sentence, while spoken language is more casual and very often contains repetitions and incomplete sentence. In addition, spoken language can convey more than just the linguistic content, that is, the emotion or traits of speakers, by utilizing prosodic or paralinguistic information [2].

With the diversity of the language, it is virtually impossible for ones to learn all languages. However, as the world is all connected, multi-lingual communication is needed more than ever. With the help of speech-to-speech translation technologies, it is possible for humans to communicate with others in different languages without having to learn them in the first place.

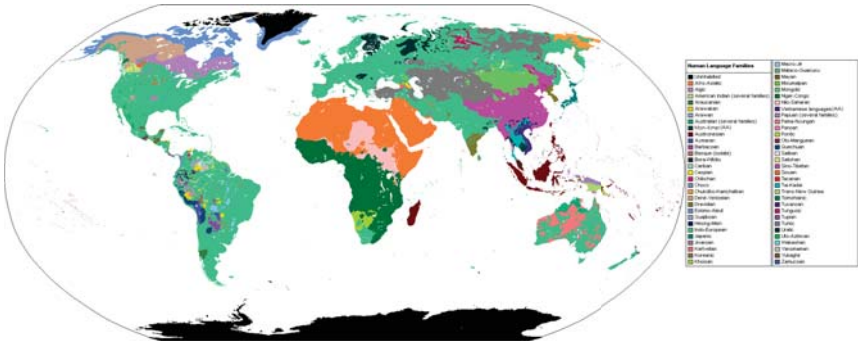


Figure 1.1. Language distribution
Image by PiMaster3-Wikipedia

1.1.2 Speech-to-Speech Translation

In order to break down the language barrier, speech-to-speech translation (an example is illustrated in Fig. 1.2) [3] techniques have been studied and developed for many years. It translates the speech across languages by the combination of three components: the automatic speech recognition (ASR), which converts speech into text; machine translation (MT), which translates text across languages; and finally, speech synthesis (TTS), which synthesizes speech from the translated text.

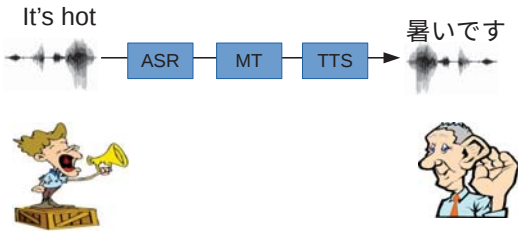


Figure 1.2. An example of English-Japanese speech-to-speech technologies help people to communicate without having to learn the other languages

The development of the speech translation technologies has been growing very rapidly. Since the first version was developed in 1980s with very limited vocabularies and simple rule-based translation, until now with very large vocabularies and continuous simultaneous speech translation [4, 5]. Some big companies have

also launched the speech translation technologies into their services. We can talk to some extent with other people in different language without learning that language.

However, as described in the previous section, speech conveys very rich information, not only "what" has been said, but also "how" is is expressed, that is, paralinguistic information. This is the current limitation of most of S2ST systems that they cannot translate paralinguistic information across languages. In the next section, we describe in detail about paralinguistic information and why it is important to be translated.

1.1.3 Paralinguistic information

Paralinguistic information is an interesting and powerful feature of speech, including age, gender, emotion, and emphasis [6]. In human–human communication, it is a valuable piece of information that help to determine speaker's states or traits, as people very often adapt their speaking style based on the assessment of their interlocutors' intention. Moreover, paralinguistic is also applied in human–machine communication, Burkhardt et al. [7] utilized paralinguistic information in detecting anger in call center and Mishne et al. [8] use it to assess quality of call center agents.

Among many type of paralinguistic, emphasis is an important element that is often used to distinguish between the focused and unfocused parts of an utterance [9]. It is particularly useful in misheard situations where speakers need to repeat the most important words or phrases of sentences, as illustrated in Fig. 1.3. That kind of situation is more likely to happen in cross-lingual speech communication using a S2S system. In speech-to-speech translation tasks, Tsiartas et al. [10] has also conducted a study on multi-lingual speech corpora and argued that emphasis information is an important factor contributes to the quality of S2ST performance. Therefore, if emphasis can be incorporated in S2ST systems, multilingual human–human conversation via S2ST will be more fulfilling experience.

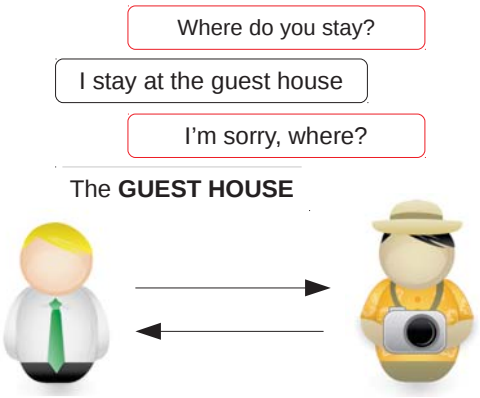


Figure 1.3. An example of a misheard situation where people put more focus on the important words.

1.2. Problems and Challenges

Integrating emphasis into S2ST systems is a challenging task. There are 2 main reasons, first, emphasis can be manipulated by many acoustic features including duration, power, and F0 [11, 12], and it can be expressed at arbitrary levels. Many works have been proposed to model and detect emphasis at binary levels using F0 feature [13, 14], however, this does not reflect the way emphasis expressed in real situation, where more than one word are emphasized and one can be more emphasized than the other.

Second, introducing emphasis into S2ST systems can potentially increase the complexity of the whole model. Kano et al. [15], Aguero et al. [16], and Anumanchipalli et al. [17] proposed model to estimate and translate emphasis. Although these approaches can handle emphasis in S2ST systems, they make the translation pipeline more complex by having 2 more components, emphasis estimation, emphasis translation (as illustrated in Fig. 1.4). In addition, Do et al. [18] requires a separate word alignment models before the emphasis translation to map the emphasis weights, and Anumanchipalli et al. [17] also needs phrase alignments to map F0 patterns. However, the word alignment can only be obtained after word translation, meaning that to translate emphasis, we need to wait for the machine translation system to predict all of the target language

sentences, creating a large delay to get the target output speech.

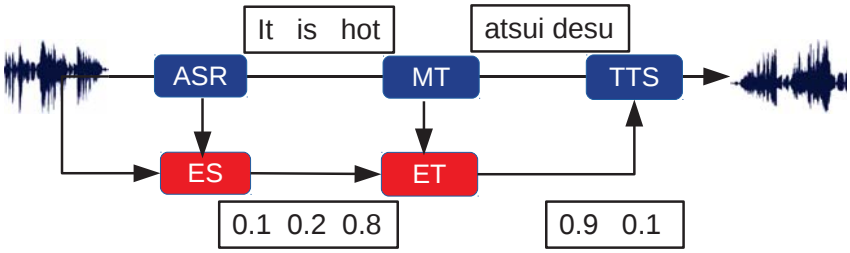


Figure 1.4. Existing emphasis S2ST systems. Two new components: emphasis estimation (ES) and emphasis translation (ET) are introduced. This greatly make the translation pipeline more complex and slow.

And finally, only acoustic feature (emphasis speech) is taken into account while emphasis can be expressed in many ways including written form at arbitrary levels. For example, one can add an adverb modifiers to increase the intensity of a sentence "it is extremely hot today". As we can see, alongside the main content of the sentence, which is weather, we can also perceive the intensity of it ("extremely"). This problem limits the capable of emphasis S2ST system that it can only translate acoustic features but not linguistic features of emphasis.

1.3. Contribution

As described in the previous section, paralinguistic information consists of many factors such as age, gender, emphasis and emotion. In this thesis, we focus on emphasis as it has been suggested having impact on human–human communication experience via S2ST systems. Other elements also should be taken into account, however, we leave them for the future work.

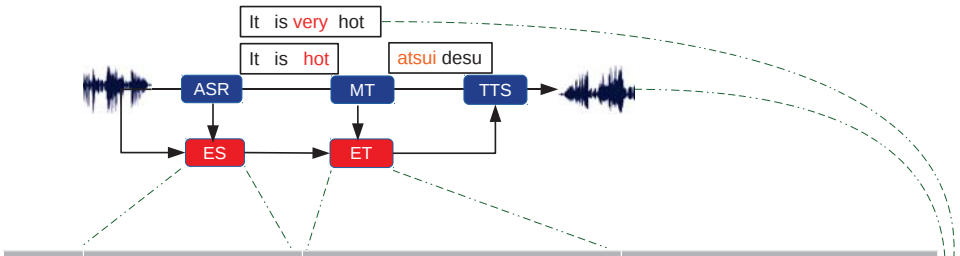
In order to address the problem described in Section 1.2, this thesis has proposed the following contribution:

Continuous emphasis modeling: This study proposed a model that can handle continuous emphasis levels in both emphasis modeling and translation. Resulting in a significant improvement of emphasis detection and translation result.

Word and emphasis joint translation models: Existing models translate emphasis and word separately, resulting in a complex model and difficulties in performing joint optimization. This study proposed an approach that can unify two models. As the result, we can preserve the performance while simplify and speedup the translation pipeline.

Translation of emphasis acoustic and linguistic features: Toward developing an emphasis translation system that can translate both acoustic and linguistic feature of emphasis, we conducted a data collection task and analyzed human perception of emphasis expressed in both text and speech form. From the analyzed result, we constructed a emphasis text transformation that can take a *neutral* text and transforms it into an *intensified* one.

The overall contribution, structure, as well as the difference from the master's work is illustrated in Fig. 1.5. The road map of this work is also showed in Figure. 1.6.



Master course	• Continuous emphasis modeling	• Discreted emphasis translation with CRF	
Ph.D. course	• Improved continuous emphasis modeling (see Chapter 3)	• Hard-attentional seq2seq • Joint translation • Emphasis translation with pauses (see Chapter 3)	• Data collection and analysis of acoustic and linguistic feature of emphasis • Acoustic-to-linguistic emphasis translation (see Chapter 5)

Figure 1.5. Contributions and structure of the thesis.

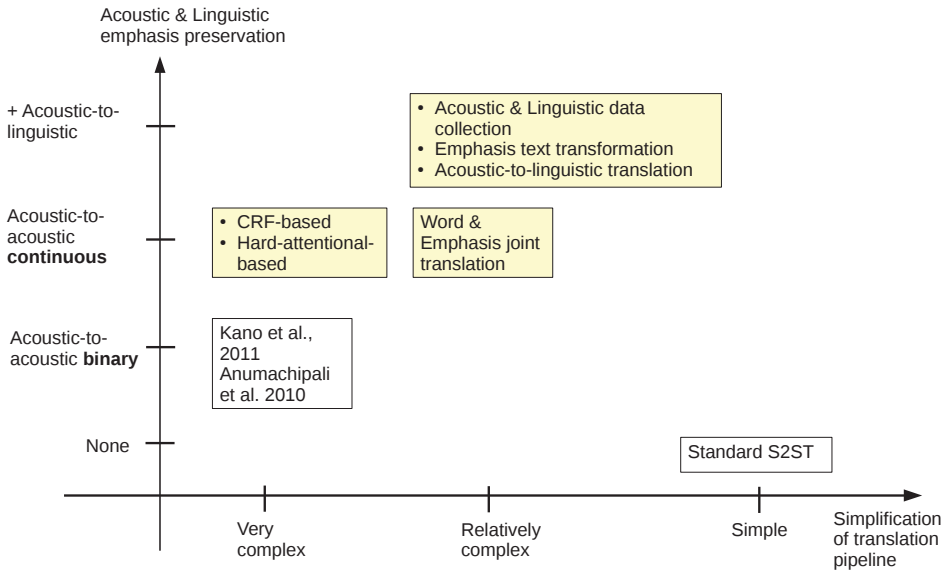


Figure 1.6. The road map of this study. The white, yellow, and red boxes denote existing works, proposed approaches, and the final target of studies on emphasis speech translation, respectively. The y-axes indicates the amount of emphasis information can be preserved and the x-axis denotes the complexity of the translation pipeline.

1.4. Outline

Chapter 2 gives an overview of the S2S system, and describes how it can translate the speech across languages. The three main components of the S2S system, speech recognition (ASR), machine translation (MT), and text-to-speech synthesis (TTS) are also described. This chapter points out that in the conventional speech translation system, the ASR system only recognize the text representing the meaning of the speech, and the MT and TTS only deal with the text as well. This is the main reason why the conventional S2S system can not translate the paralinguistic information.

With regards to the emphasis modeling and translation, Chapter 3 presents the use of linear regression hidden semi-Markov models (LR-HSMM) and emphasis adaptive training to model and estimate a real-numbered value that represents for word-level emphasis. The basic idea is that we model the emphasized speech and normal speech models separately, and the word-level emphasis is an interpolation parameter between those two models. If the words are emphasized, they will have a higher emphasis level than the other words. Chapter 4 our proposed approaches in emphasis translation including pause prediction, hard-attentional translation, and joint translation models.

Chapter 5 presents our proposed corpus that contain emphasis expressed in both speech and text form. The corpus is used to analysis human production and perception of emphasis. We show the results of experiments on how different emphasis levels can be expressed by speech and text, as well as the human perception on both acoustic and linguistic feature of emphasis. In addition, we also describe our proposed approach on transforming a *neural* text into an *intensified* text by utilizing PoS tags and ngram scoring.

Chapter 6 concludes the thesis with an overview of emphasis translation and the direction for the future research.

Chapter 2

Speech-to-Speech Translation

2.1. Overview

The conventional speech translation system (S2S) [3] consists of 3 main components, ASR, MT, and TTS. As illustrated in Figure 2.1, the work-flow of the S2S system can be described as follows. First, the ASR module transcribes an audio from a source language into a transcription. After that, the MT module translates the transcribed text into a target language. Finally, the TTS module synthesizes an audio given the target sentence. In the following sections, we give a short description for each component.

2.2. Automatic Speech Recognition (ASR)

ASR aims to convert the speech signal into the corresponding word sequence. Let's first take a look at how it works. The first step of the speech recognition is to extract speech features $\boldsymbol{o}$ given speech signals $\boldsymbol{x}$. The standard speech feature set is mel-frequency cepstral coefficients (MFCCs). The MFCC features contain the components of the audio signals that are good for identifying the linguistic content and discard irrelevant information such as paralinguistic information or noise. After that the ASR predicts the most plausible word sequence consists of K words $\boldsymbol{w} = [w_1, w_2, \cdots, w_K]$ that maximize the conditional probability,

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}}{\operatorname{argmax}} P(\boldsymbol{w}|\boldsymbol{o}). \tag{2.1}$$

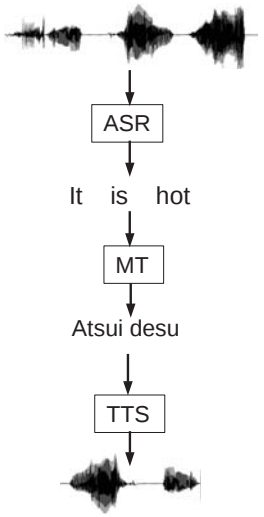


Figure 2.1. Conventional speech translation system

Applying the Bayes's rule we have,

$$P(\boldsymbol{w}|\boldsymbol{o}) \;=\; \frac{P(\boldsymbol{w})P(\boldsymbol{o}|\boldsymbol{w})}{P(\boldsymbol{o})} \propto P(\boldsymbol{w})P(\boldsymbol{o}|\boldsymbol{w}). \tag{2.2}$$

As we can see, the $P(\boldsymbol{w}|\boldsymbol{o})$ is factored into two parts. The first part is $P(\boldsymbol{w})$, also called language model probability and the second part $P(\boldsymbol{o}|\boldsymbol{w})$ is acoustic probability.

The standard way to calculate $P(\boldsymbol{w})$ is to use n–gram language modeling, where the $P(\boldsymbol{w})$ is break down into smaller parts,

$$P(w = [w_1, w_2, \cdots, w_K]) = \prod_{i=1}^{K} P(w_i|w_{i-n+1}^{i-1}), \tag{2.3}$$

where $P(w_i|w_{i-n+1}^{i-1})$ denotes the probability of the word w_i given $n-1$ preceding words (context words). Basically, the higher order of n–gram model will usually give better results. However, it is also becomes harder to calculate the n–gram probability for high order n–grams because of the data sparsity problem. The probability of the n–gram will become smaller when the order becomes higher, and is not helpful for the ASR system anymore. Usually the 3– or 4–gram language model is adopted for the standard ASR systems.

With regards to the acoustic model. The acoustic probability $P(\boldsymbol{o}|\boldsymbol{w})$ is calculated by

$$P(\boldsymbol{o}|\boldsymbol{w}) = \sum_{\boldsymbol{Q}} P(\boldsymbol{o}|\boldsymbol{Q})P(\boldsymbol{Q}|\boldsymbol{w}), \tag{2.4}$$

where $\boldsymbol{Q}$ is all possible phoneme sequences derived from $\boldsymbol{w}$. As we can see, the phoneme is used instead of the word. The reason is mainly because we can not collect enough data for every word for a particular language. $P(\boldsymbol{o}|\boldsymbol{Q})$ represents how likely it is that the speech feature $\boldsymbol{o}$ is observed given the phoneme sequence $\boldsymbol{Q}$ and is typically formulated by Gassian mixture hidden Markov model (GMM-HMM) [19, 20] as illustrated in Figure 2.2,

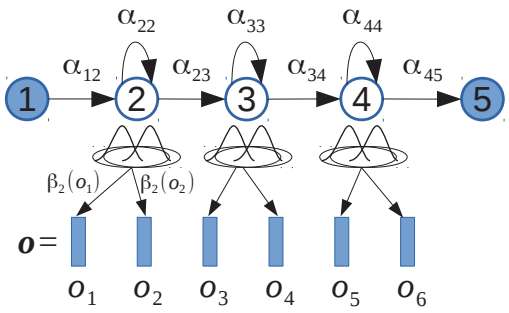


Figure 2.2. HMM-based phoneme model with 5 states (3 emitting states: 2, 3, and 4; and 2 non-emitting states: 1 and 5). The parameter α_{ij} is the state transition probability between the state i^{th} and j^{th}, and $\beta_i(o_k)$ is the probability that the observation o_k is generated from the i^{th} state.

Recently, the deep neural network (DNN) acoustic modeling has drawn much attention in speech recognition researchers as an alternative modeling technique to the GMM model. In the DNN-HMM approach, the DNN aims to calculate the posterior probability $\beta_i(o_k)$ instead of a mixture of Gaussians. The DNN-HMM outperforms the GMM-HMM when using the same amount of data, and in experiments in [21] it is required about 7 times larger amount of training data for the GMM-HMM to have the same performance as the DNN-HMM.

2.3. Statistical Machine Translation (MT)

The MT system lies in the middle of the S2S system, and has a job to translate the hypothesis from ASR module to a particular target language. There are many methods that can be applied to MT task such as tree-based [22] and phrase-based [23] translation models. This section gives a high-level description for the phrase-based translation model.

The phrase-based model uses the translation of phrases as atomic units. The idea is to split a sentence into small phrases and performs the translation as illustrated in Figure 2.3.

The MT system can be descibed in mathematics as follows. Given a source language sentence $\mathbf{f}$, the task is to find the best target language sentence $\mathbf{e}$ by applying Bayes's rule as follows,

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}}\, P(\mathbf{e}|\mathbf{f}) = \underset{\mathbf{e}}{\operatorname{argmax}}\, P(\mathbf{e})P(\mathbf{f}|\mathbf{e}), \tag{2.5}$$

where the $P(\mathbf{e})$ is the language model for the target sentence $\mathbf{e}$, and the $P(\mathbf{f}|\mathbf{e})$ is decomposed into

$$P(\mathbf{f}|\mathbf{e}) = \prod_{i=1}^{I} \phi(f_i|e_i)d_i, \tag{2.6}$$

where I is number of phrases in source sentence, d_i is the distance score which is calculated by a distance-based reordering model [23], and $\phi(f_i|e_i)$ is the phrase translation probability

$$\phi(f_i|e_i) = \frac{count(e_i, f_i)}{\sum_{f_i'} count(e_i, f_i')}. \tag{2.7}$$

The phrase is extracted from a word alignment which is the output of unsupervised learning methods [24, 25].

The advantages of phrase-based model is it can handle non-compositional phrases, and the more data we have, the longer phrases can be learned. The phrase-based translation model is also a successful approach, currently used by many big companies. This thesis utilized the phrase-based model for the speech translation system.

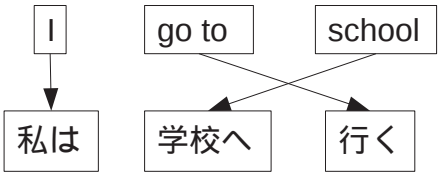


Figure 2.3. Example of phrase-based translation model

2.4. Text-to-speech Synthesis (TTS)

Text-to-speech is the last component in the S2S system, which synthesize the target audio given the translated hypothesis.

The most three common TTS techniques are unit selection [26], neural-net-based [27], and hidden semi-Markov model (HSMM) based [28]. In the unit selection method, the synthetic speech is created by concatenating the pieces of recorded speech extracted from a speech database. For example, the speech sound of “hh”, “ah”, “l”, and “ow” are concatenated to form the word “hello”. As a result, the synthetic speech is very natural and intelligible. However, this method requires a very large speech corpus to achieve a good performance, and it is also difficult to control or modify the synthetic speech (e.g., increase the duration of particular words, or change the speaking style) [29, 30].

The last two approaches, neural-net-based and HSMM-based, are parametric and generally, produce better prosodic patterns than the concatenation approach. The neural-net-based method, is, however, requires a large amount of speech data, which leads to difficulties to apply into expressive speech synthesis tasks, because the collection of expressive speech are both time and resource consuming. On the other hand, the HSMM-based approach allows us to modify the synthetic speech easily and the amount of required training data is also much smaller, say an hour. It is also possible to rapidly adapt an existing TTS model to a particular speaker using a limited amount of that speaker’s training data [31] which can not be done in unit selection methods.

The HSMM is a hidden Markov model (HMM) with explicit state duration probability distributions as shown in Figure 2.4, which improves the naturalness of the synthetic speech over the HMM model. The model training consists of three steps, label analysis, speech parameter extraction, and HMM training. The

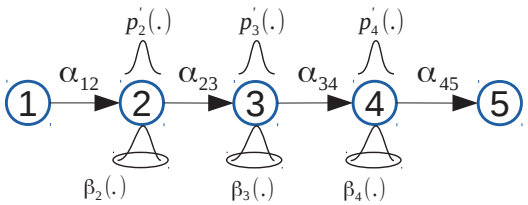


Figure 2.4. An example of hidden semi-Markov models with 5 states. α_{ij} is the state transition probability from state i to state j, $\beta_i(.)$ is the likelihood probability, and $p_i'(.)$ is the duration probability distribution for state i.

label analysis converts the word sequence into full context labels which represent many linguistic aspects (e.g., phone identity, accent, stress, location, and part-of-speech). This information makes it possible to model and synthesize audio more naturally. The second step is the speech parameter extraction which extracts acoustic parameters from speech signals. This process is different from ASR module in the way that it keeps all the information such as duration, speaker characteristic, emphasis.

Let's define the TTS in mathematics. The output speech parameter vector sequence $\boldsymbol{o}$ is determined by maximizing the likelihood function given the state sequence consists of T states $\boldsymbol{q} = [q_1, \cdots, q_T]$, and the HSMM model set $\mathcal{M}$

$$\hat{\boldsymbol{o}} = \underset{\boldsymbol{o}}{\operatorname{argmax}}\, P(\boldsymbol{W}\boldsymbol{o}|\boldsymbol{q}, \mathcal{M}), \tag{2.8}$$

where $\boldsymbol{W}$ is the weighting matrix for calculating the dynamic features [32].

This thesis adopts the HSMM-based TTS model. The reason is not only to inherit the advantages of the HSMM-based method, but also the flexibility to modify it to model the emphasized speech which is described in Section 3.

Chapter 3

Emphasis Speech Modeling

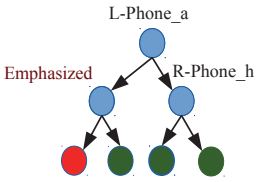
With regards to speech modeling techniques, DNN-based and hidden semi-Markov model (HSMM) approaches are most common used. The DNN-based approaches have recently drawn many attentions and yield better intelligible and natural sounds than the HSMM. However, it requires a large amount of training samples, usually more than 10 hours, to outperform the HSMM approach [33] and it is also not flexible in modeling expressive speech. The HSMM approach, however, require much less amount of data (1 to 3 hours) and provide flexibility to model the different varieties of speech [34]. Previous work on word-level emphasis modeling based on HSMMs has relied on state clustering with emphasis contextual factors. A simple approach uses emphasis contextual factors indicating whether or not a word and its neighbor words are emphasized, and creates an *emphasis decision tree* [35] or a *factorized decision tree* [36] with some nodes having an emphasized question, as illustrated in Fig. 3.2 (a). While these methods are both expressive and effective, they have a disadvantage in the way that they make a hard zero-one distinction between unemphasized and emphasized words, or in other words, they use binary emphasis representation. However, in reality, emphasis is more subtle, and can be better represented using a continuous variable where a larger number indicates a higher level of emphasis (as illustrated in Fig. 3.1).

In this section, we describe our approach to improve LR-HSMM-based emphasis modeling to solve this problem. First, to adopt continuous emphasis levels and to improve the parameter optimization process, we make an extension of cluster adaptive training (CAT) [37] to *emphasis adaptive training* as illustrated

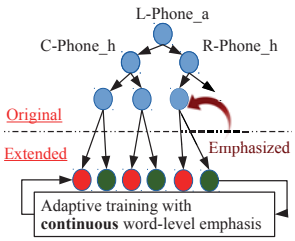
It	is	very	hot	today
0	0.1	0.6	0.8	0.2

Figure 3.1. An example of Word-level emphasis. In this example, the word "very" and "hot" are emphasized with higher emphasis value.

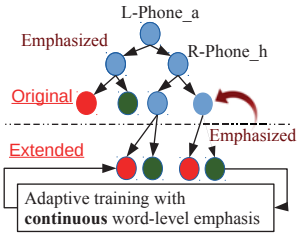
in Fig. 3.2 (b). Next, to take advantage of both parameter optimization and expressive decision tree modeling, we propose a *hybrid approach* that considers both the state clustering (binary emphasis) using the emphasis contextual factor and emphasis adaptive training (continuous emphasis), as illustrated in Fig. 3.2 (c).



(a) State clustering with emphasis context



(b) Emphasis adaptive training



(c) Hybrid system

Figure 3.2. Emphasis modeling techniques including (a) state clustering using contextual information, (b) emphasis adaptive training without emphasis context (b), and (c) a hybrid system that adopts both s trategies.

3.1. Binary-Level Emphasis Modeling

State clustering is an approach that helps to reduce the number of HSMM states and the need for a lar ge amount of training data by clustering HSMM states using

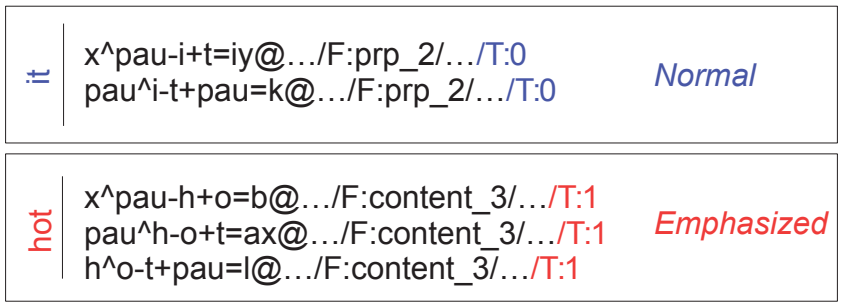


Figure 3.3. An example of full contextual labels for the word "it" and "hot" where the word "it" is normal and "hot" is emphasized. The context "T:*" is the additional emphasis factor, and the remaining items are traditional cont extual information.

some cluster criterion. This clustering is generally performed by using decision trees, which decide the cluster of HMM states based on a number of contextual factors. By simply adding an emphasis conte xtual factor to the cluster criterion, as illustrated in Fig. 3.3, we can model *normal* and *emphasized* HSMM states [35][1]. The decision tree constructed by having additional emphasis context (Fig. 3.2 (a)) can separate Gaussians components into *normal* and *emphasized* ones. Although this approach is simple and easy to implement, there are three problems: (1) it does not guarantee that the emphasis question appears in all paths starting from the root to leaf nodes, causing a problem that there are some nodes that make no distinction between emphasized and non-emphasized words; (2) it separates the training data into normal and emphasized parts, causing emphasized and normal nodes to only be trained with emphasized and normal data, respectively; and (3) emphasis is treated as a binary value indicating emphasized or not, and thus it is not possible to model emphasis at a "medium" level using continuous values.

[1]Of course, it is possible to use mor e contextual factors, i.e. indicating whether preceding and succeeding words are emphasized. However, in this work we omit these factors to maintain comparability of the evaluation with other approaches using the same context factors.

3.2. Continuous-Level Emphasis Modeling

The state clustering approach described in the previous section can model zero-one emphasis. However, in reality, emphasis is more subtle. For example, one sentence might have two emphasized words with one having a smaller level of emphasis than the other. Therefore, it may be better to represent emphasis as a continuous variable where a larger number indicates a higher emphasis level. In this section, we describe continuous word-level emphasis modeling [38] using linear-regression hidden semi-Markov models (LR-HSMMs) [39] with HSMM state clustering.

3.2.1 LR-HSMM definition

We assume a word sequence consists of J words $\mathbf{w} = [w_1, \cdots, w_J]$, and a length T acoustic feature vector $\boldsymbol{o} = [\boldsymbol{o}_1, \cdots, \boldsymbol{o}_T]$. As the observed feature vector $\boldsymbol{o}_t$ at time t, we use a combination of a spectral feature vector $o_t^{(1)}$ and F_0 feature vector $o_t^{(2)}$ as described in [40]. The likelihood function of the LR-HSMMs is given by

$$P(\boldsymbol{o}|\boldsymbol{\Lambda}, \mathcal{M}) = \sum_{\text{all } \boldsymbol{q}} P(\boldsymbol{q}|\boldsymbol{\Lambda}, \mathcal{M}) P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\Lambda}, \mathcal{M}), \tag{3.1}$$

where $\boldsymbol{q} = [\boldsymbol{q}_1, \cdots, \boldsymbol{q}_T]$ is the HSMM state sequence, $\boldsymbol{\Lambda} = [\lambda_1, \cdots, \lambda_j, \cdots, \lambda_J]$ is the word-level emphasis sequence, and $\mathcal{M}$ is an HSMM parameter set. The LR-HSMM has two separate Gaussian components, normal and emphasized Gaussians, which are derived by using a decision tree constructed using HSMM state clustering, which described in the above section.

Because emphasis is defined at the word level, all linear-regression states that belong to one word will share the same emphasis level, as illustrated in Fig. 3.4. The state output probability density function modeled by a Gaussian distribution[2] is given by

$$P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\Lambda}, \mathcal{M}) = \prod_{t=1}^{T} P(\boldsymbol{o}_t|q_t, \omega_t, \mathcal{M}), \tag{3.2}$$

[2]Specifically, a multi-space probability distribution [41] is used for the F_0 component in this paper.

$$P\left(\boldsymbol{o}_t|q_t=i,\omega_t,\mathcal{M}\right)=\prod_{s=1}^{2}\mathcal{N}\left(\boldsymbol{o}_t^{(s)};\boldsymbol{\mu}_{n,i}^{(s)}+\omega_t\boldsymbol{b}_i^{(s)},\boldsymbol{\Sigma}_i^{(s)}\right), \tag{3.3}$$

where $\boldsymbol{\mu}_{n,i}^{(s)}$ is the normal Gaussian mean vector at state i and stream s, ω_t is frame-level emphasis equivalent to λ_j if state $i \in w_j$; and s is a stream index (*i.e.*, $s=1$ for the spectral features and $s=2$ for the F_0 features), and $\boldsymbol{b}_i^{(s)}$ a vector expressing the difference between the normal and emphasized Gaussian mean,

$$\boldsymbol{b}_i^{(s)}=\boldsymbol{\mu}_{e,i}^{(s)}-\boldsymbol{\mu}_{n,i}^{(s)}, \tag{3.4}$$

where $\boldsymbol{\mu}_{e,i}^{(s)}$ is the emphasized Gaussian mean vector. The duration probability $P\left(\boldsymbol{q}|\boldsymbol{\Lambda},\mathcal{M}\right)$ is also derived in a similar way to the state output probability,

$$P\left(\boldsymbol{q}|\boldsymbol{\lambda},\mathcal{M}\right) = \prod_{i=1}^{N}P\left(d_i|\omega_i,\mathcal{M}\right), \tag{3.5}$$

$$P\left(d_i|\omega_i,\mathcal{M}\right) = \mathcal{N}\left(d_i;\mu_i^{(d)}+\omega_i b_i^{(d)},{\sigma_i^{(d)}}^2\right), \tag{3.6}$$

where $\mu_i^{(d)}$ and $b_i^{(d)}$ are the normal Gaussian mean and the difference between emphasized and normal Gaussian means, respectively; d_i is an HSMM state duration, $\omega_i=\lambda_j$ if $d_i \in w_j$; and N is the number of states in the sentence HSMM sequence (i.e., the sum of d_i over N HSMM states is equivalent to T).

3.2.2 Word-level emphasis sequence estimation

Given an observation sequence $\boldsymbol{o}$, and its transcription, the process to estimate the word-level emphasis sequence is as follows [38]: first, an LR-HSMM is constructed by selecting the Gaussian distributions corresponding to the context of the given transcription. Then, emphasis is estimated by determining maximum likelihood estimates of the emphasis weight sequence, which is the same as the cluster weight estimation process in the cluster adaptive training (CAT) algorithm [37]. The word-level emphasis weight sequence is estimated by maximizing the HSMM likelihood as follows:

$$\hat{\boldsymbol{\lambda}}=\arg\max_{\boldsymbol{\lambda}}P\left(\boldsymbol{o}|\boldsymbol{\lambda},\mathcal{M}\right). \tag{3.7}$$

This maximization process is performed with the EM algorithm [42].

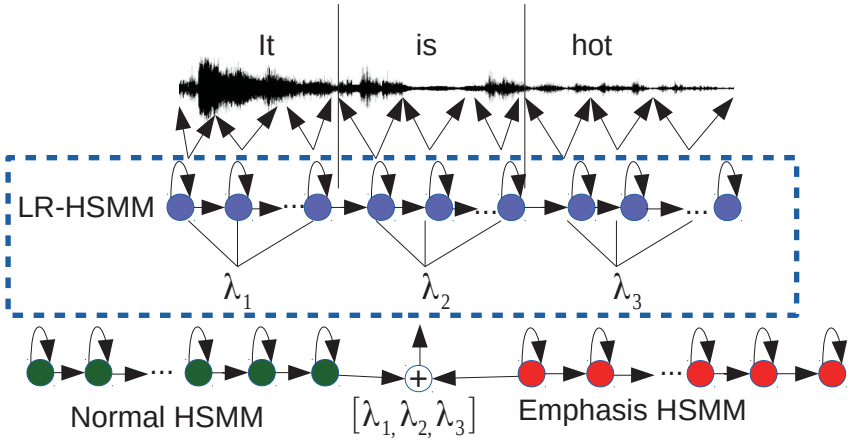


Figure 3.4. An example of linear-regression HSMMs. Each word has its own emphasis level λ_j, and all HMM states that belong to the same word will share the same emphasis level.

3.3. Emphasis Adaptive Training

First, we make an extension of CAT [37] to allow it to perform emphasis adaptive training. The idea of the proposed method is to iteratively estimate and update the word-level emphasis sequences and model parameters, respectively.

Given the estimated word-level emphasis sequence $\mathbf{\Lambda} = [\lambda_1, \lambda_2, \cdots, \lambda_J]$, we want to find the model parameters that maximize the likelihood function

$$\hat{\mathcal{M}} = \underset{\mathcal{M}}{\operatorname{argmax}} P\left(\boldsymbol{o}|\boldsymbol{\lambda}, \mathcal{M}\right). \tag{3.8}$$

The maximization process is performed with the EM algorithm as follows:

1. Use the existing model parameters to estimate word-level emphasis sequences $\hat{\mathbf{\Lambda}}$ as described in the previous section. In other words, this step automatically generates pseudo-labels for the training data.

2. Update the mean of the normal Gaussian component at stream s, $\boldsymbol{\mu}_{n,m}^{(s)}$, and duration state d, $\mu_{n,m}^{(d)}$, given estimated word-level emphasis sequences

$\hat{\boldsymbol{\Lambda}}$. Below is the formula used to update $\boldsymbol{\mu}_{n,m}^{(s)}$.

$$\boldsymbol{\mu}_{n,m}^{(s)} = \boldsymbol{G}^{-1}\boldsymbol{K} \tag{3.9}$$

$$\begin{aligned}\boldsymbol{G} = \sum_{m' \in m} \big[(1-\omega^{(m')}) \sum_t \gamma_t^{(m')} \boldsymbol{o}_t \\ - \omega^{(m')}(1-\omega^{(m')}) \sum_t \gamma_t^{(m')} \boldsymbol{\mu}_e^{(m)} \big],\end{aligned} \tag{3.10}$$

$$\boldsymbol{K} = \sum_{m' \in m} (1-\omega^{(m')})^2 \sum_t \gamma_t^{(m')}, \tag{3.11}$$

where m' is the untied model of linear Gaussian component m, $\omega^{(m')}$ is Gaussian-level emphasis that is equivalent to λ_j if the untied model m' belongs to the word w_j. The mean of emphasis Gaussian $\boldsymbol{\mu}_{e,m}^{(s)}$ and duration model $\mu_{n,m}^{(d)}$ can be updated in a similar way.

3. Go back to step 1 until the model is converged.

Note that in this paper, the covariance matrices of Gaussian components are kept unchanged for simplification.

Based on emphasis adaptive training, we propose an approach to model emphasis without the need of state clustering with emphasis context as illustrated in Fig. 3.2 (b). In this approach, the decision tree is constructed without any emphasis context, as illustrated in Fig. 3.2 (b) – Original. After that, the original leaf nodes are turned into intermediate nodes by adding an emphasis question splitting each of them into normal and emphasized nodes (Fig. 3.2 (b) – Extended). At this point, the emphasized and normal leaf nodes are equivalent. Then, to ensure that the parameters of emphasized and normal Gaussians are different, we add to the emphasized Gaussians a mean difference vector $\bar{\boldsymbol{b}}^{(s)}$, which is calculated based on the tree created from the state clustering approach. Finally, we adopt emphasis adaptive training described above to further optimize the parameters.

Unlike the previous approach where emphasized and normal Gaussians are trained only on emphasized or normal speech respectively, emphasis-adaptive-training-based approaches are able to utilize all the training data to train the model parameters. When training on emphasized samples, the emphasized Gaussian components get more weight (emphasis level) than normal Gaussians, and vice versa.

However, the simple approach described here also has a weakness in that it forces emphasis questions to always be asked right before the leaf nodes. This has the potential to result in sub-optimal decision tree structure. We resolve this weakness in the following section.

3.4. Hybrid System for Continuous Emphasis Modeling

Next, we propose a hybrid approach that takes advantage of both of the above approaches. First, a decision tree (the original tree) with emphasis questions asked at some intermediate nodes is constructed as in Section 3.1. Then, we extend leaf nodes that belong to paths that do not have an emphasis question asked in any of the intermediate nodes as shown in Algorithm 1.

Algorithm 1 State splitting algorithm.

```
 1: procedure StateSplitting(s)
 2:     if s has emphasis question then
 3:         return
 4:     else
 5:         if s is leaf node. then
 6:             SET s as intermediate node.
 7:             ADD emphasis question to s.
 8:             SPLIT s into 2 leaf nodes.
 9:             return
10:         else
11:             StateSplitting(left node of s).
12:             StateSplitting(right node of s).
```

The state splitting process 6-8 will duplicate the mean and covariance matrix of Gaussian components of the state being split. After splitting the tree, every leaf node is guaranteed to represent either a normal or emphasized Gaussians. Then, to ensure that emphasized and normal Gaussians are different, the same procedure as the previous section is applied.

Finally, we perform emphasis adaptive training with continuous emphasis representations to further optimize model parameters for the nodes split by the line 8 of Algorithm 1.

3.5. Emphasis Estimation

Given an observation sequence $\boldsymbol{o}$, and its transcription, the process to estimate the word-level emphasis sequence is as follows [38]: first, an LR-HSMM is constructed by selecting the Gaussian distributions corresponding to the context of the given transcription. Then, emphasis is estimated by determining maximum likelihood estimates of the emphasis weight sequence, which is the same as the cluster weight estimation process in the cluster adaptive training (CAT) algorithm [37]. The word-level emphasis weight sequence is estimated by maximizing the HSMM likelihood as follows:

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} P\left(\boldsymbol{o}|\boldsymbol{\lambda}, \mathcal{M}\right). \tag{3.12}$$

This maximization process is performed with the EM algorithm [42].

3.6. Experiments

3.6.1 Experimental setup

In this section, we evaluate the performance of emphasized speech modeling using state clustering, emphasis adaptive training, and the hybrid approach. The experiments were conducted using a bilingual English-Japanese emphasized speech corpus [11], which has emphasized content words that were carefully selected to maintain the naturalness of emphasized utterances. The corpus detail is shown in Table. 3.1.

In the experiments, we selected 2 speakers for each language with 916 utterances for training and 50 utterances for testing. Thus, we have 100 testing samples in total for each language. The LR-HSMM model was trained for each speaker separately, resulting in 4 models in total. The speech features were extracted using 31 mel-cepstral coefficients including 25 dimension spectral parameters, 1

Table 3.1. The corpus detail.

Bilingual speakers	3
Monolingual speakers	6 Japanese, 1 English
Number of utterances / speaker	966

dimension log-scaled F_0, and 5 dimension aperiodic features. Each speech parameter vector includes static features and their delta and delta-deltas. The frame shift was set to 5 ms. Each HSMM model is modeled by 7 HMM states including initial and final states. We adopt STRAIGHT [43] for speech analysis.

With regards to emphasis adaptive training, we performed the adaptive training for the first 6 iterations, then re-estimate word-level emphasis sequences. These are then used to perform emphasis adaptive training until the model converges.

3.6.2 Objective evaluation on word-level emphasis prediction

In the first experiment, we evaluate the performance of the different models in emphasis prediction, where we are given an input speech signal and would like to predict whether each word is emphasized. For each system, we estimate the word-level emphasis sequences for the testing data, then classify them to normal and emphasized labels using a threshold of 0.5. Then, we calculate the F-measure for all systems. The result is shown in Fig. 3.5.

As we can see, the model using emphasis adaptive training and the hybrid approach outperform the state clustering approach in both languages by 2-5% F-measure[3]. One possible reason for this is that in state clustering approaches, emphasis questions do not appear at all paths starting from the root to leaf nodes, leading to some emphasized words having weak emphasis levels. To test this hypothesis, we perform an analysis showing the percentage of the number of decision tree traversing without asking for emphasized questions in the state

[3]We did not carry out subjective evaluation explicitly, however, our previous work [38] has shown that the human emphasis prediction has about a 4% reduction of F-measure compared to objective evaluation due to the lack of pauses in the synthetic speech.

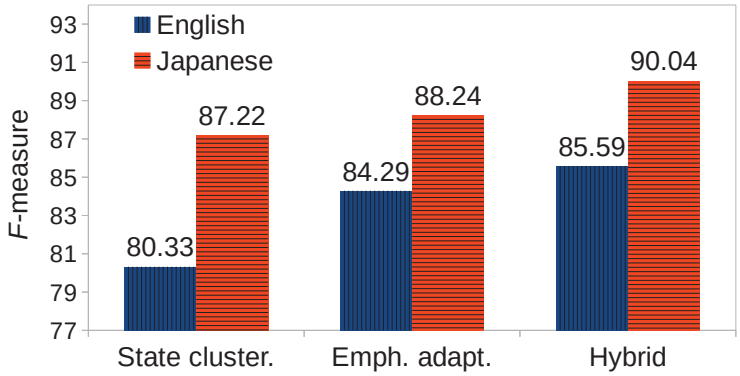


Figure 3.5. Emphasis prediction accuracy.

clustering approach for both languages. The result shown in Fig. 3.6 indicates that many times we traverse through the tree to derive emphasized and normal Gaussian components, emphasis questions are not asked at in all three acoustic feature streams for 11.37% times in English, and no emphasis question is asked more than half the time in at least one of the feature streams. On the other hand, the proposed approach guarantees that we can always derive different emphasized and normal Gaussian components. This is also one explanation for why the improvement of the hybrid system compared to other methods is larger in English than Japanese.

3.6.3 Subjective evaluation of naturalness

In the next experiment, we use the models to synthesize speech of the Japanese data and perform a preference test evaluation to evaluate the naturalness of the synthetic speech. 50 utterances in the testing data were synthesized with each system using the ground-truth emphasis labels (e.g., "it is *really* hot today" with emphasis label "0 0 1 0 0"). 7 Japanese native listeners performed a pairwise evaluation over all pairs of systems.

As shown in Fig. 3.7, the hybrid approach generated more natural audio compared to all others. We hypothesize the reasons are as follows:

- **State clustering**: The *emphasized* and *normal* Gaussians are trained using only *emphasized* or *normal* speech, respectively. Although emphasis ques-

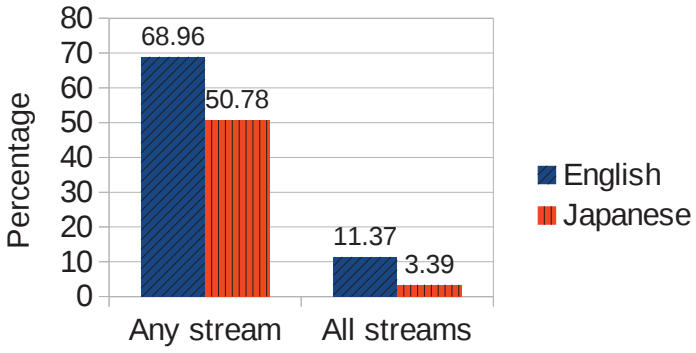


Figure 3.6. The percentage of decision tree traversing without asking for emphasized questions in the state clustering approach. "All streams" and "Any stream" indicate situations in which emphasis questions are not asked in all feature streams (lf0, duration, and spectral) or any of them, respectively.

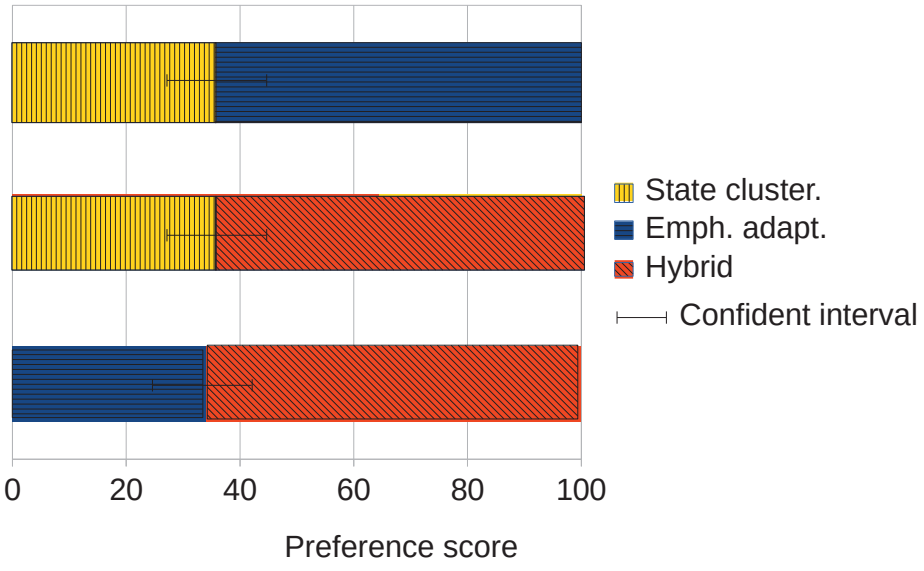


Figure 3.7. Preference score of synthetic speech for each system.

tions are placed in the decision tree according to the likelihood function, due to the limitation of training data, some Gaussian components do not get a sufficient amount of training samples, leading to low quality synthetic speech.

- **Adaptive training**: In this approach, we can utilize all training data to train Gaussian components. When training on emphasized speech samples, the *emphasized* Gaussian components get more weight (emphasis level) than *normal* Gaussians, and vice versa, leading to higher quality than the state clustering approach. However, the emphasis questions are forced to be asked right before the leaf node (not according to likelihood function), this potentially makes the audio become unnatural.

- **Hybrid approach**: This approach inherits advantages from both above systems. The decision tree has emphasis questions are placed according to likelihood function, some paths that do not have emphasis questions are refined using state splitting, and the model parameters are further optimized using adaptive training.

3.6.4 Effect of ASR Errors on Emphasis Estimation

In this experiment, we investigated the effect of ASR errors on emphasis estimation. The reason why we conducted this experiment is to investigate the effect of 3 different ASR errors: deletion, insertion, and substitution on the emphasis estimation. By knowing those effect, we can adjust the ASR system in a way that has smallest effect on the emphasis estimation. For example, we can adjust the word insertion penalty of the decoding process to trade off between deletion and insertion error.

In order to conduct the experiment, we simulated 3 type of ASR errors for each utterance as follows,

- Deletion: For each utterance, we randomly delete one word.

- Substitution: For each utterance, we randomly substitute one word with another word that has similar pronunciation. This is usually the case of substitution error in real ASR system.

- Insertion: For each utterance, we randomly insert one word in an existing dictionary.

Then, we performed the emphasis estimation on modified transcription, and calculate the F-measure. The result is shown in Figure 3.8. As we can see, for both languages, the insertion error has largest effect on emphasis estimation, while the deletion and substitution errors has smallest effect to English and Japanese, respectively. The result indicates that when we perform the emphasis translation, we should adjust the ASR to produce less insertion error by increasing the word insertion penalty during decode process. Although it will increase the deletion error, the effect to emphasis estimation can be reduced.

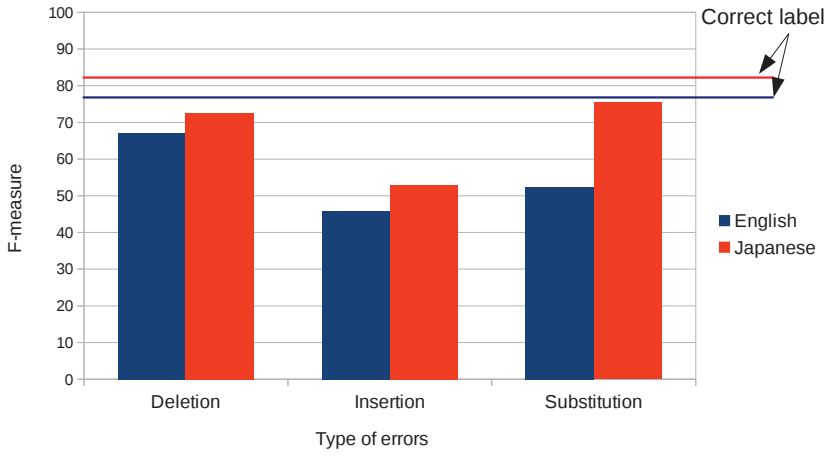


Figure 3.8. Effect of ASR errors on emphasis estimation

3.7. Discussion

In this chapter, we have proposed methods for emphasis adaptive training that deals with continuous emphasis levels and a hybrid system that combine state clustering and adaptive training approaches. Experiments showed that the proposed model outperforms other methods by 2-5% F-measure of emphasis estimation accuracy, and produces more natural audio.

Apart from improving emphasis estimation performance, continuous emphasis levels also has a major potential advantage against binary emphasis in emphasis translation. Figure 3.9 and 3.10 respectively show an example of emphasis translation with binary and continuous values. As we can see, with binary emphasis values, there are only 2 possible outcomes of translation, the result can be only either correct or incorrect. However, with continuous values, there are unlimited outcomes and if the system makes a small mistake, it might still be able to preserve emphasis in the target language. Our work on emphasis translation in [18], where we evaluated the performance between systems using ground-truth label (binary values) and estimated emphasis values, has showed that although the ground-truth labels are used, the emphasis translation performance is still lower than the system used estimated emphasis values.

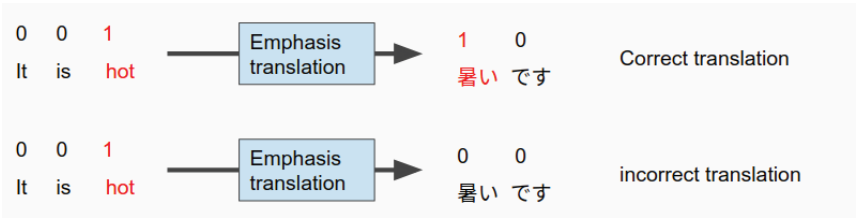


Figure 3.9. Emphasis translation with binary values

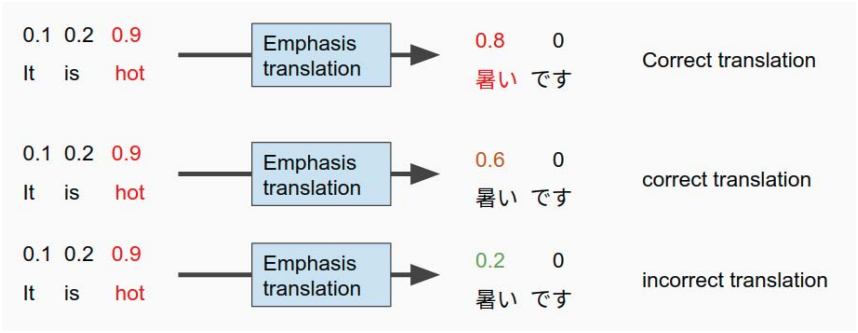


Figure 3.10. Emphasis translation with continuous values

On the other hand, there are still some limitations remaining. First, the system is not robust against unknown data, its performance dropped significantly when an unknown speaker speech is observed. Specifically, on a unknown speaker

test set, the accuracy on emphasis prediction of the system is only approximately 20%. Second, the approach also has difficulties in handling emphasis words with the emphasis level just slightly higher than normal words or in case speakers use pauses rather than acoustic features. In such situation, if we lower the emphasis detection threshold to have better recall of emphasis detection, the precision will also drop significantly.

In the future, we plan to incorporate emphasis adaptive training with more sophisticated clustering such as factorized decision trees, and MLLR adaptation.

Chapter 4

Translation of Emphasis Acoustic Features

Previous chapter has shown the way we model all acoustic features of emphasis as a single real-numbered emphasis level. In this chapter, we describe our proposed approaches that take the emphasis level and translate it into a target language so that people can perceive how source language speech is emphasized acoustically. Various approaches are proposed including conditional random fields (CRF), hard-attention sequence-to-sequence, and joint translation models (as illustrated in Fig. 4.1).

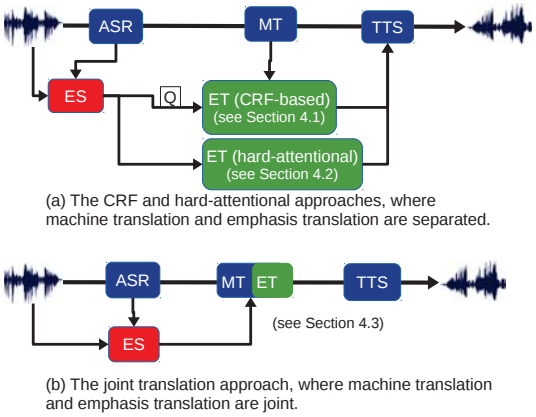


Figure 4.1. Proposed approaches on emphasis translation (green boxes).

4.1. CRF-based Emphasis Translation

4.1.1 Conditional Random Field

Conditional random fields (CRF) [44] are discriminative models which model the conditional probability $P(\mathbf{y}|\mathbf{x})$ directly instead of the join probability $P(\mathbf{y}, \mathbf{x})$ (also known as the generative model). The advantage of the conditional model is that we do not have to model the $P(\mathbf{x})$ which can include complex dependencies. This leads to much simpler and compact model than the generative approach with the ability to incorporating a large number of input feature $\mathbf{x}$.

The linear-chain CRF can be depict as a undirected graph in Figure 4.2. θ and μ are equivalent to the transition probability and emission probabilities in the hidden Markov model. Given a training data that consists of T samples

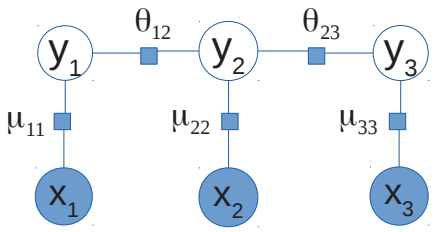


Figure 4.2. The linear-chain conditional random field with the model parameters $\{\theta, \mu\}$.

$D = [(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_t, y_t), \cdots, (\mathbf{x}_T, y_T)]$, the conditional probability is calculated as,

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \sum_{t=1}^{T} \exp \left\{ \sum_{i,j \in S} \theta_{ij} \mathbf{1}_{\{y_t=i\}} \mathbf{1}_{\{y_{t-1}=j\}} + \sum_{i \in S} \sum_{o \in O} \mu_{oi} \mathbf{1}_{\{y_t=i\}} \mathbf{1}_{\{x_t=0\}} \right\} \tag{4.1}$$

where $\{S, O\}$ are the state and observation space, $\{i, j, o\}$ denote the states and observation, respectively.

We can write the Equation 4.1 more compactly by using the concept of feature function $f_{ij}(y, y', \mathbf{x}) = \mathbf{1}_{\{y=i\}} \mathbf{1}_{\{y'=j\}}$ and $f_{io}(y, y', x) = \mathbf{1}_{\{y=i\}} \mathbf{1}_{\{x=o\}}$. The annotation θ_k and f_k are used for the general model parameter θ_{ij} and μ_{io} and the feature function that ranges over both f_{ij} and f_{io}. Then the Equation 4.1

becomes

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}, \tag{4.2}$$

where $Z(\mathbf{x})$ is a normalization function

$$Z(\mathbf{x}) = \sum_{\mathbf{y}'} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y'_t, y'_{t-1}, \mathbf{x}_t) \right\} \tag{4.3}$$

The model parameter θ are optimized by maximizing the conditional probability

$$\mathcal{L}(\theta) = P(\mathbf{y}|\mathbf{x}). \tag{4.4}$$

The simplest way to optimize $\mathcal{L}(\theta)$ is gradient descent, but the convergence speed is slow [45]. In practice, the common optimization approach is limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) [46].

4.1.2 Emphasis Translation with Conditional Random Fields

Our next step is emphasis translation, or to take word-level emphasis estimates in the source languages $\hat{\boldsymbol{\lambda}}^{(f)}$, and convert them to emphasis estimates in the target language $\hat{\boldsymbol{\lambda}}^{(e)}$. We perform estimation of emphasis in the target language using the conditional random fields (CRFs) approach. To train CRFs to predict target side emphasis, we create training data consisting of source and target words $\mathbf{w}^{(f)}$ and $\mathbf{w}^{(e)}$, and the corresponding estimated emphasis values. As $\hat{\boldsymbol{\lambda}}^{(e)}$ is a sequence of continuous values, and CRFs requires discrete state sequences, we first quantize $\hat{\boldsymbol{\lambda}}^{(f)}$ and $\hat{\boldsymbol{\lambda}}^{(e)}$ into buckets, giving us a discrete sequence $\hat{\boldsymbol{\lambda}}^{(f)'}$ and $\hat{\boldsymbol{\lambda}}^{(e)'}$. We then create CRFs training data that consists of N samples $D = [(\mathbf{x}_1, \lambda_1^{(e)'}), \cdots, (\mathbf{x}_n, \lambda_n^{(e)'}), \cdots, (\mathbf{x}_N, \lambda_N^{(e)'})]$, where $\mathbf{x}_n$ is a feature vector for each word in $w_n^{(e)}$ consisting of:

- source word-level emphasis $\lambda_j^{(f)}$, and its context,
- source word $w_j^{(f)}$, and word context,
- source word part of speech (PoS) $pos(w_j^{(f)})$, and PoS context,
- target word $w_n^{(e)}$, and word context,
- target word PoS $pos(w_n^{(e)})$, and PoS context,

where context means the information of one succeeding and one preceding words.

To decide which source features correspond to a target word $w_n^{(e)}$, we use one-to-one word alignments between $w_j^{(f)}$ and $w_n^{(e)}$. The CRFs model parameters are optimized using Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) approach [46] as implemented in CRFSuite [47].

4.2. Pause prediction

Pause prediction is not a new research field, with a large body of research trying to tackle this problem [48, 49, 50]. The main distinction between these previous methods and our work is that while previous methods attempted to predict pauses from text (linguistic) information only, in our work we are given information about whether the word in question is emphasized, which gives us a stronger signal about whether pauses should be inserted or not. In this section, we describe two approaches that are able to utilize both linguistic and emphasis information to predict pauses based on CRFs.

The pause prediction problem can be described as follows: Given a word sequence and its word-level emphasis sequence, we want to predict in which of the below 4 positions a pause is inserted.

Before : a pause is inserted before the word.

After : a pause is inserted after the word.

Both sides : pauses are inserted before and after the word.

None : there is no pause inserted.

Generally speaking, this is a classification problem with 4 classes.

4.2.1 Pause extraction

The first step is to extract pauses from the training data by 3 steps, first, we train a speech recognition model on the same data, this step will give us a speaker dependent acoustic model for each speaker. Then, we perform forced alignment on the training data to derive audio-text alignments. Finally, from the alignment, we extract all pause segments that have duration at least 50ms as pauses.

4.2.2 CRF-based pause prediction

We adopt a CRF model to predict pause position, the input features include words, part-of-speech tags, emphasis degree, and context information of the preceding and succeeding units. Table 4.1 shows an example of input features. In the example, the word *hot* is the emphasized word, and we can see that a pause is inserted after the word *is* and before the word *hot*. In a standard sentence, this placement of a pause may seem unnatural. However, because the word *hot* is emphasized intentionally, the pause can be inserted to give a sign that the word *hot* is important.

Table 4.1. An example of input features for the sentence "it is <p> hot" with word-level emphasis sequence "0 0.1 0.8". Note that pauses are represented by commas, and we also use the context information of the preceding and succeeding units.

Position	**Word**	**Part-of-speech**	**Emphasis**
None	it	PRP	0
After	is	VBZ	0.1
Before	hot	JJ	0.8

4.3. Hard Attentional Neural Net Based Emphasis Translation

Even though a CRF-based ET can preserve emphasis, its major problem is that it must quantize continuous emphasis levels into discrete labels. Although this mechanism increases the ratio of the number of labels and their training samples, the translation model is prone to make very bad predictions. For instance, instead of predicting 0.9, it might predict 0.1. The reason is it cannot capture the difference between 0.9 and 0.1 because those values are treated as separate discrete labels.

Another problem with the CRF-based approach is that although it model local dependencies well (by adding more feature functions), it has difficulty handling long-term dependencies. One can use many feature functions to handle this

problem, but as feature functions increase, more data are required. And since emphasis translation requires parallel emphasized speech, which is very hard to collect, this approach is not practical.

In this section, we describe our proposed hard-attentional approach based on sequence-to-sequence (seq2seq) LSTM models, which are type of recurrent neural networks (RNNs), that can solve remaining problems exist in CRF-based approaches.

4.3.1 Sequence-to-sequence LSTM Models

In this section, we introduce sequence-to-sequence (seq2seq) LSTM techniques, which works best for sequential data such as sequence of words, and is, currently, considered as standard approach for machine translation. We utilize seq2seq approaches because they have achieved impressive results for many tasks, such as speech recognition [51, 52] and machine translation (MT) [53]. Particularly, attentional-based seq-to-seq [54, 55] achieved state-of-the-art performances for MT and ASR tasks and can model long-term dependencies, overcoming the problems of local dependencies in CRFs. In addition, models can be defined that can simultaneously handle both continuous and discrete variables, as well as cost functions that take into account label distances, for example, mean squared errors.

LSTM [56] is a special kind of recurrent neural network model that can capture long-term dependencies by special units called *memory blocks* and also manages the information going through it using forget, input, and output gates. Given input vector $\mathbf{x}_t$ at time t and hidden vector $\mathbf{h}_{t-1}$ and cell state $\mathbf{C}_{t-1}$ at time $t-1$, the information flow can be described:

- Calculate forget gate $\mathbf{f}_t$:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f \times [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f). \tag{4.5}$$

- Calculate input gate $\mathbf{i}_t$ and estimate cell state $\widetilde{\mathbf{C}}_t$:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \times [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i), \tag{4.6}$$

$$\widetilde{\mathbf{C}}_t = tanh(\mathbf{W}_C \times [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C), \tag{4.7}$$

- Update cell state $\mathbf{C}_t$:

$$\mathbf{C}_t = \mathbf{f}_t \times \mathbf{C}_{t-1} + \mathbf{i}_t \times \widetilde{\mathbf{C}}_t, \tag{4.8}$$

- Calculate output vector $\mathbf{h}_t$:

$$\begin{aligned} \mathbf{v}_t &= \sigma(\mathbf{W}_v \times [\mathbf{h}_{t-1,\mathbf{x}_t}] + \mathbf{b}_v), & (4.9)\\ \mathbf{h}_t &= \mathbf{v}_t \times tanh(\mathbf{C}_t). & (4.10) \end{aligned}$$

Here $\mathbf{W}$ and $\mathbf{b}$ are the matrix and bias vectors of the neural network layers. The core component of an LSTM is cell state $\mathbf{C}_t$ (Eq. (4.8)), which is controlled by forget gate $\mathbf{f}_t$ that is multiplied by the previous cell state values to decide which history information it should forget, and input gate $\mathbf{i}_t$, which is multiplied to the estimated cell state to decide which information is sent to the cell state.

An attention seq-to-seq LSTM model consists of an LSTM encoder, which encodes the input information, an LSTM decoder, which takes the encoded output to make a prediction, and an attention layer, which calculates an attention vector. The seq-to-seq translation model can be written as follows:

- Encode the input features to obtain hidden states $\mathbf{h}^{(s)}$:

$$\mathbf{h}_i^{(s)} = enc(\mathbf{h}_{i-1}^{(s)}, \mathbf{x}). \tag{4.11}$$

- Compute the attention vector $\mathbf{a}_j^{(t)}$ and context vector $\mathbf{c}_j$:

$$\mathbf{a}_j^{(t)} = att(\mathbf{h}_j^{(t)}, \mathbf{h}_i^{(s)}), \tag{4.12}$$

  where j is the prediction time step, and $\mathbf{h}_j^{(t)}$ is the decoder hidden state. Given $\mathbf{a}_j^{(t)}$ as weights, context vector $\mathbf{c}_j$ is computed as the weighted average over all source hidden states $\mathbf{h}^{(s)}$.

- Predicts target labels y_j,

$$\begin{aligned} P(y_j|y_{<j}, \mathbf{x}) &= softmax(\mathbf{W}_t \widetilde{\mathbf{h}}_j^{(t)}), & (4.13)\\ \widetilde{\mathbf{h}}_j^{(t)} &= tanh(\mathbf{W}_c[\mathbf{c}_j; \mathbf{h}_j^{(t)}]). & (4.14) \end{aligned}$$

Our proposed hard-attentional model for emphasis translation is a modified version of the seq-to-seq model described in the following section, based on an assumption that we have a target language word sequence that was predicted from an external MT model and word alignments from an external word alignment model.

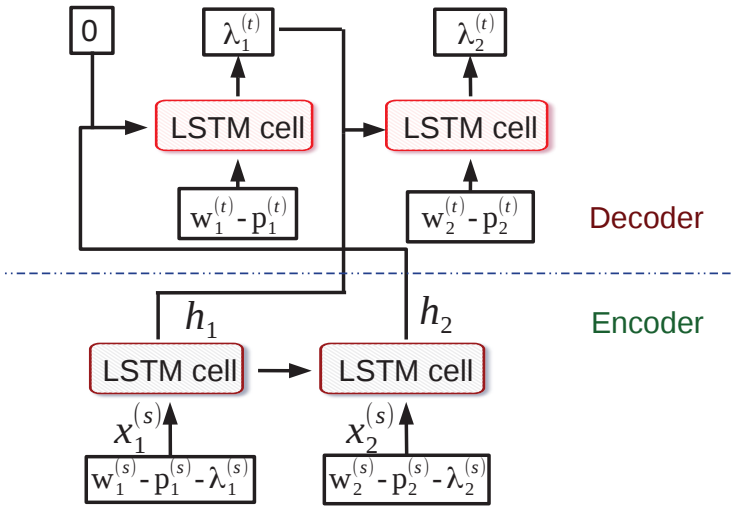


Figure 4.3. Unfolded hard-attentional encoder-decoder LSTM model for translating emphasis sequence $\boldsymbol{\lambda}^{(e)}$ into target output sequence $\boldsymbol{o}^{(f)}$. It considers many linguistic features including word sequence $\mathbf{w}_i^{(e,f)}$ and part of speech sequence $\mathrm{p}_i^{(e,f)}$ from both source and target languages.

The whole encoder-decoder process can be written as a function of input features:

$$\lambda^{(t)} = f(\mathbf{x}^{(s)}), \tag{4.15}$$

where $\lambda^{(t)}$ is the target output emphasis sequence, $\mathbf{x}^{(s)}$ is the sequence of the source-language input features including words $\mathbf{w}^{(s)}$, PoS $\boldsymbol{p}^{(s)}$, and emphasis weights $\boldsymbol{\lambda}^{(s)}$. The previous work [57] reported that both words and PoS tags play a crucial role to have a good translation model.

4.3.2 The encoder

As illustrated in Fig. 4.3, the encoder is a standard LSTM model that takes input vector $\mathbf{x}_i^{(e)}$ that consists of words ($w_i^{(e)}$), part-of-speech tags ($\mathrm{p}_i^{(e)}$), and emphasis levels ($\lambda_i^{(e)}$), and encodes them into a single vector that is suitable for predicting emphasis levels.

The input PoS tags are converted into one-hot vectors with the size is equal to PoS vocabulary size. Word embeddings [58] are applied to map words onto vectors that capture the similarity between words. All these input features are concatenated into a single vector and fed to the encoder.

The encoder is pre-trained by appending a linear neural-net layer on top of it with an output size of 1 to predict the emphasis level that is fed into the input layer, similarly as an auto-encoder model [59] (Fig. 4.4 (a)). We want output hidden layer $\mathbf{h}$ to represent the features that are the most useful to predict emphasis levels (called "emphasis representations").

4.3.3 The decoder

The decoder is also a standard LSTM model, and the input layer contains both linguistic information (words, PoS) and vector representations calculated by the encoder, based on a novel hard-attentional model.

The name hard-attentional reflects how the decoder calculates the emphasis representation vectors used as input. The example in Fig. 4.3 demonstrates this mechanism. Assume that the word pairs $w_1^{(t)}$-$w_2^{(s)}$ and $w_3^{(t)}$-$w_1^{(s)}$ are aligned based on word alignments. To generate output $\lambda_2^{(t)}$, linguistic features $w_2^{(t)}$ and $\mathrm{p}_2^{(t)}$, and the previous output $\lambda_1^{(t)}$, the decoder takes encoded $\mathbf{h}_1$ from the encoder output, because word pair $w_1^{(s)}$-$w_2^{(t)}$ is aligned. For unaligned words, we use zero vectors as the emphasis representation vectors.

We propose 2 decoders as follows:

- **LSTM_emph**: The model directly predicts target emphasis sequence $\boldsymbol{\lambda}^{(t)}$.
- **LSTM_diff**: The output of the model is considered as the difference from the input emphasis level. The target emphasis level of the j-th word is calculated by, $\lambda_j^{(t)} = f(\mathbf{x}^{(s)}) + \lambda_i^{(s)}$, where the model gets "attention" from word $w_i^{(s)}$.

The intuition behind the **LSTM_diff** is an intention to put stronger weight on the corresponding source-language emphasis when predicting target emphasis.

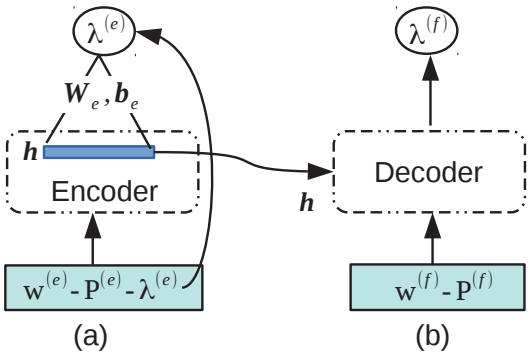


Figure 4.4. Training procedure for the hard-attentional model.

4.4. Joint Words and Emphasis Translation

4.4.1 Limitation of Complex Translation Pipeline

To translate emphasis, the translation pipeline requires ES and ET components in addition to the S2ST system. It is now very complex with 5 components, and 6 internal dependencies (represents by black arrows) (Fig. 4.5). All downstream components have to wait for upstream outputs, resulting in large translation delays. Moreover, each component uses very different techniques, causing difficulties to perform joint training and decoding.

Even though the hard-attention seq-to-seq model solves the problems of the CRF-based, the problem with a complex translation pipeline remains. In this section, we propose a joint translation framework based on an attentional NMT that simultaneously combines MT and ET to translate words and emphasis at the same time. Our approach is also based on seq-to-seq approaches as the hard-attention approach. The difference is that we do not require external MT or word alignment models anymore. All components are combined into a single joint translation model, allowing us to perform joint optimization and inference.

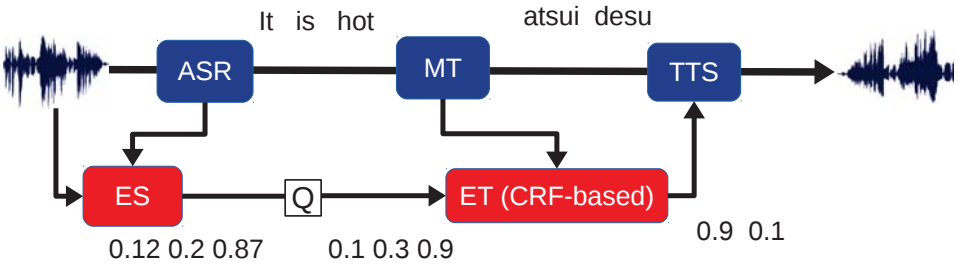


Figure 4.5. Existing emphasis speech translation model that consists of many separate components and dependencies. It also requires emphasis quantization (Q) before the translation.

The major difficulty when integrating emphasis with word translation is that the amount of text data usually overwhelms the amount of emphasis data. This is because the emphasis data are derived from parallel emphasized speech that is much harder to collect than parallel text data, which can be massively collected by crawling websites [60].

4.4.2 Overview

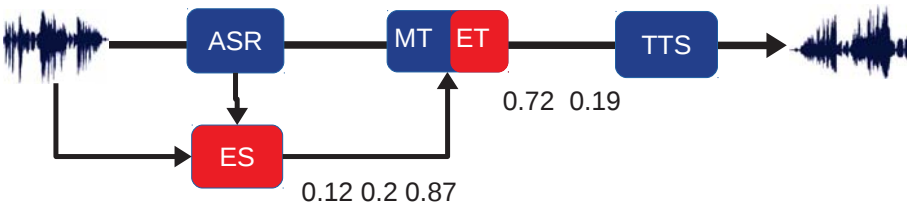


Figure 4.6. Proposed joint model simplifies translation pipeline and can jointly translate words and emphasis with one-word delay.

We define the joint translation model as follows. Given a source language word and an emphasis sequence are denoted as $\boldsymbol{W}^{(s)}$ and $\boldsymbol{e}^{(s)}$, respectively. The model predicts one target word $w^{(t)}$ at a time followed by a prediction of its emphasis weight $e^{(t)}$. Next, we detail how the encoder and decoder handle both words and emphasis weights.

4.4.3 Encoder with emphasis weights

One way to embed emphasis weights into the encoder is to concatenate them with word representation to form an input vector $[w_i^{(s)}, e_i^{(s)}]$ of the encoder (*Emp-Enc*) and compute the hidden unit:

$$\boldsymbol{h}_i^{(s)} = enc([w_i^{(s)}, e_i^{(s)}]). \tag{4.16}$$

By doing this, we ensure that the emphasis weights are also encoded with words. However, since the effect of emphasis on MT remains unknown, we need to explore alternative ways to incorporate emphasis into the encoder to analyze such an effect. Therefore, we propose adding emphasis after encoding words (*SkipEnc*) as follows:

$$\boldsymbol{h}_i^{(s)} = [enc(w_i^{(s)}), e_i^{(s)}] \tag{4.17}$$

The idea of *SkipEnc* is that if emphasis weights negatively affect machine translation, adding them after the encoder might weaken the effect.

4.4.4 Decoder with emphasis weights

As illustrated in Fig. 4.7, the decoder has two components. A word prediction layer follows the standard NMT, and emphasis prediction layer $\boldsymbol{W}_e$ that takes input is the combined vector of the predicted word and the decoder hidden activation as follows:

$$e_i^{(t)} = \boldsymbol{W}_e([\widetilde{\boldsymbol{h}}_i^{(t)}, w_i^{(t)}]). \tag{4.18}$$

However, as described above, the lack of emphasis data compared with the text data might saturate the effect of the source emphasis when going through many hidden layers. To overcome this problem, we utilize residual connection in the way that the source emphasis weight is also used when predicting target emphasis weights (Fig. 4.7),

$$e_i^{(t)} = \boldsymbol{W}_e([\widetilde{\boldsymbol{h}}_i^{(t)}, w_i^{(t)}]) + e_{id(\boldsymbol{a}_i)}^{(s)}, \tag{4.19}$$

where function $id(\boldsymbol{a}_i)$ returns the index of the largest value of weighted vector $\boldsymbol{a}_i$ that indicates the source aligned word.

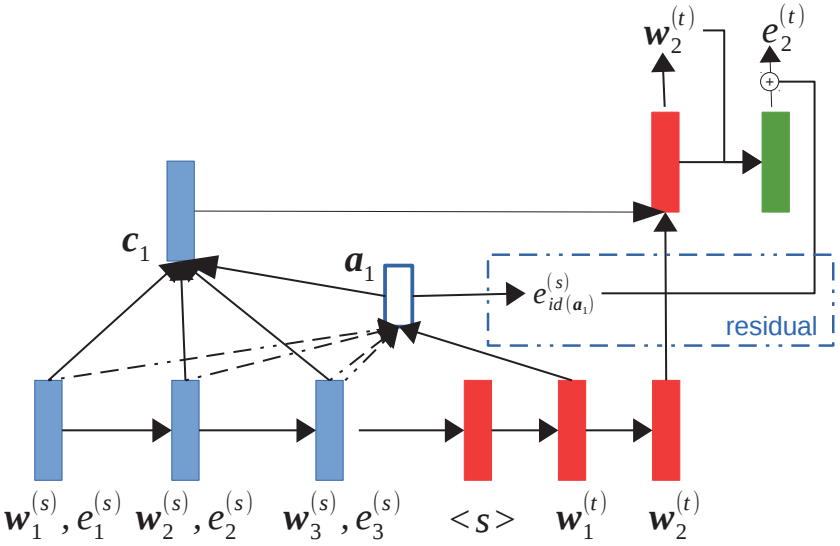


Figure 4.7. Joint word-emphasis translation framework with word dependencies and residual connection.

4.4.5 Training procedure

To train our model, we utilize two objective functions, cross entropy (CE) for word prediction and mean square error (MSE) for emphasis prediction, because the CE function greatly outperforms MSE with discrete labels, which is the case of word prediction. Since emphasis weights are continuous, the CE function cannot be utilized as the objective function for emphasis prediction.

The training algorithm is standard back propagation through time (BPTT) in which the errors from the machine and emphasis translations are sequentially back–propagated. Note that the errors are not joint because their scales are different.

4.5. Experiments

In our experiment, we first evaluated the effectiveness of using continuous emphasis level in a translation model and the ability to handle the long-term dependencies of the proposed hard-attentional seq-to-seq model against the previous CRF-based approach.

Regarding the evaluation of the joint translation model, since our proposed model is our first attempt, to the best of our knowledge, for integrating emphasis and word features in a single model, analysis must be conducted on the effect of emphasis on the standard MT translation model (Section 4.5.6). This critical step not only shows the effect of emphasis as a feature, but also provides some vital cues for optimally joining ET and MT.

Moreover, to further reduce the complexity of the input features and network structure, we also evaluated the effect of PoS tags on the ET and MT models (Section 4.5.7). Do et al. [57] argued that PoS tags are crucial features that can boost the ET performance by 4% F-measure. However, it creates another dependency for the translation model. On the other hand, the seq-to-seq translation model is capable of not only learn how to translate but also be able to learn the semantic meaning of words, and since the semantic meaning are closely related to syntactic meaning (PoS tags) [61], we expect that it can avoid the need of the PoS tag feature.

Finally, based on the analysis result, we conduct experiments with the joint translation model in comparison with both hard-attention and CRF-based approaches (Section 4.5.8).

4.5.1 Experimental setup

Corpus

The corpus consists of emphasis and machine translation data. The former contain 966 parallel English and Japanese utterances [62]. In each language, at least one of the content words in the sentence is emphasized, and the number of emphasized words is identical between languages. The number of speakers is 8, including 3 native English ($\text{En}^{\{1,2,3\}}$) and 5 native Japanese ($\text{Ja}^{\{1,2,3,4,5\}}$) speakers.

To create training and testing data for our emphasis translation evaluation, we divided 966 utterances of each speaker into 2 sets of 866 and 100 samples such that the same sentences are used for all speakers. We then paired the 866 utterances of each English speaker with those of all 5 Japanese speakers, resulting in 4330 ($866 * 5$) training, and 100 testing samples for each English speaker. The testing data consist of 157 emphasized words, in which 30 exist in the training

data and 127 do not.

Regarding to the machine translation data, we utilized 2 sets, the BTEC and BTEC+TED corpora, which contain ∼450k and ∼670k parallel sentences, respectively. We created 2 training MT datasets to evaluate the effectiveness of the emphasis information on the MT task with more varieties of testing conditions. The overall data used for experiments is shown in Table. 4.2

Table 4.2. Experimental data detail.

ET training data	4330 samples
MT training data	450k samples (BTEC) and 670k (BTEC+TED)
ET evaluation data	300 samples
MT evaluation data	5000 samples

Emphasis translation procedure & measurement

In this paper, to evaluate the performance of the emphasis translation in isolation, we assumed that the MT system produces 100% correct translation outputs. Word alignments

To measure the emphasis translation accuracy, we first performed emphasis translation to derive the target emphasis sequences and then measured its accuracy in the target language both objectively or subjectively (Fig. 4.8). In the objective evaluation, the target emphasis values are classified as "emphasized" or "not emphasized" using a threshold of 0.5[1] and compared them with true values. In the subjective evaluation, we first synthesized the audio from the translated emphasis sequences, and gave the output audio to 7 Japanese native listeners to predict the emphasized words[2]. In both evaluations, we calculated the F-measure, which ranged from 0 to 100 representing how accurately the system preserved emphasis in the target language.

[1]This has been reported in the previous work [18] as having the best performance to classify emphasized and normal words.

[2]There is no constraint on how emphasized words are expressed, it is up to the listeners to make a binary decision on whether a word is emphasized.

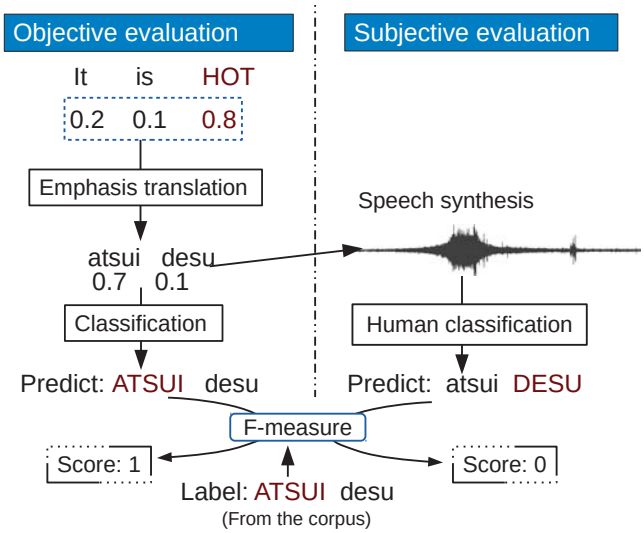


Figure 4.8. Example of the emphasis translation procedure and measurement methods.

CRFs

In the CRF model, the word-level emphasis was quantized to the closest {0, 0.3, 0.6, 0.9}[3], the input features are words, PoS tags, and PoS contexts in the target language side. The model directly predicts the target side emphasis sequence. This setting achieved the best performance compared to other features combinations.

Hard-attentional model

- **The encoder:** The encoder input consists of words, PoS tags, and emphasis levels. The input layer has 138 dimensions including 100 word embeddings, 37 one-hot PoS tags, and the emphasis levels. The hidden layer has 100 dimensions.

- **The decoder:** The input gate consists of 100 word embedding dimensions and 17 one-hot PoS dimensions. The attentional vector taken from the encoder was added to the input gate's output. The input words and PoS

[3]The buckets yield the best performance among other buckets

are also respectively converted into word-embedding and one-hot vectors.

The word embeddings for both the encoder and decoder were pre-trained using the BTEC travel conversation corpus [63] using word2vec toolkit [58].

Joint translation model

Our encoder and decoder models have 1 layer (unless stated otherwise), 512 cells, and 512-dimensional word embeddings. We trained for a maximum of 20 epochs using the RMSprop algorithm [51]. Emphasis prediction layer $\boldsymbol{W}_e$ was frozen when trained with fake emphasis data to avoid learning from unrealistic emphasis weights.

When trained with text data, the learning rate was set to $1e-4$ and $5e-5$ when trained with emphasis data. We employed an early stop learning rate schedule and reduced the learning by a factor of 2 whenever loss increased on the development set and stopped the training when the learning rate fell below 1e-5. Our mini-batches were 128 and 10 for the word translation and the emphasis translation task, respectively. The batches were shuffled before every training epoch.

4.5.2 Pause prediction evaluation

In this experiment, we evaluate the performance of pause prediction models based on CRFs. 4 classes were used, they are "none", "before", "after", and "both sides".

We evaluate the performance of the CRF-based pause prediction model using different combination of input features, which includes words, part-of-speech tags, word-level emphasis degree, and information of preceding and succeeding units. The measurement metric is F-measure, which is the harmonic mean of precision and recall. The result is shown in Table 4.3.

First, by comparing the 1st line with the 2nd and 3rd line. We can see that emphasis information is important for pause prediction, improving 3% F-measure. Second, the last line that shows the input feature without context information has lower accuracy compared to the 1st line, which has context information, indi-

Table 4.3. Pause prediction performance using different combination of input features. "ctx" denotes context information of a preceding and succeeding units.

Emph.	Emph. ctx.	Word	Word ctx.	Tag	Tag ctx.	F-measure
✓	✓	✓	✓	✓	✓	**88.76**
		✓	✓	✓	✓	85.38
				✓	✓	84.81
✓		✓		✓		85.71

cating that the context information is also very important because it gives more information for pause prediction.

4.5.3 Emphasis translation with pause evaluation

In the final experiment, we evaluate the S2S translation system integrating with the CRF-based pause prediction model. Four systems were:

No-emphasis : A speech translation system without emphasis translation as described in [38].

Baseline : An emphasis translation system (CRF-based) without pause prediction as described in [38].

+Pause : The baseline system with the CRF-based pause prediction model.

Natural : Natural speech by native Japanese speaker.

First, we synthesize audios from each system. Then, we asked 6 native Japanese listeners to listen to the synthesized audio and identify the emphasized word. Finally, we score each system with F-measure. In addition, we perform an objective evaluation where the emphasized word is detected by an emphasis threshold of 0.5[4] yielding 91.6% F-measure. Note that it is not possible that the subjective result is better than the objective result, because there is a chance that text-to-speech systems make mistakes in synthesizing emphasized audios. The result is shown in Fig. 4.9.

[4]This value is an optimized value that has been tested in [38].

As reported in [38], the baseline system outperforms *No-emphasis* system in conveying emphasis across languages. However, it is still 4% lower accuracy than the objective evaluation. By integrating the pause prediction model, we gain 2% F-measure, which is closer to the objective result. The result indicates that pauses are an important type of information that helps listeners perceive the focus of speech better, and also prove our conjecture that pause might be used to indicate that upcoming words are important.

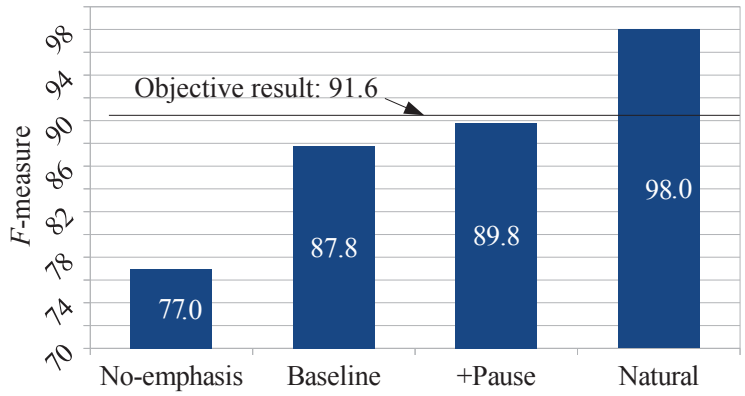


Figure 4.9. Subjective evaluation of emphasis translation with pause insertion.

4.5.4 Hard-attentional models: objective evaluation

We performed a preliminary experiment using the same corpus as in a previous work [18] with 916 training samples and 50 testing samples. The results showed that our proposed method achieved a 92.6% F-measure, which exceeds the previous work by 1%. Although the dataset was too small to conclude that the proposed method is better than CRFs by such a small margin, it demonstrates that the proposed method performs comparably with the previous work on the same corpus. To make the result more reliable, we conducted larger scale experiments with the dataset described in the Section 4.5.1.

Fig. 4.10 shows the objective F-measure for emphasis prediction on this larger amount of data. In all 3 test sets and on average, the proposed methods outperformed the CRFs. According to the bootstrap resampling significance test [64],

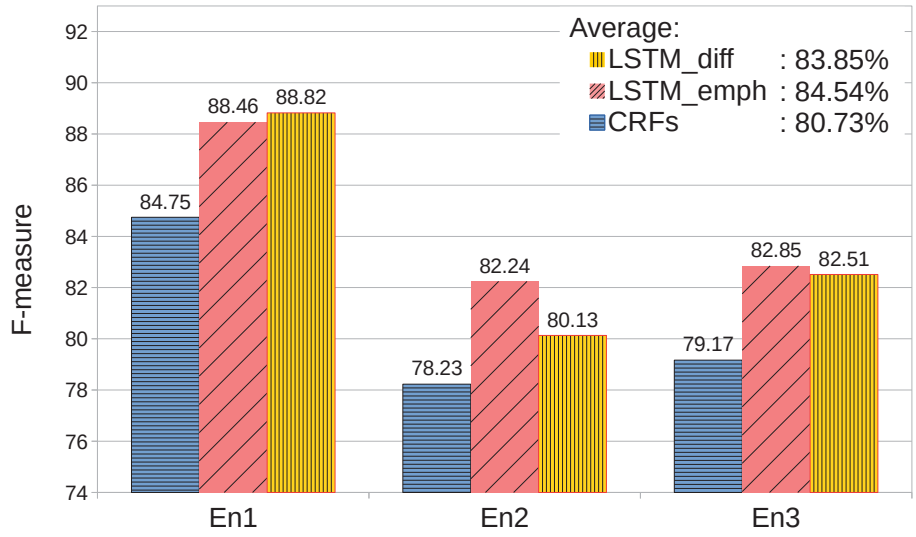


Figure 4.10. Objective emphasis prediction of hard-attentional enc-dec with LSTM_diff and LSTM_emph architectures.

both results are significant at the $p < 0.01$ level. On the other hand, the difference between *LSTM_diff* and *LSTM_emph* was not significant, demonstrating that the LSTM model can learn emphasis level differences between aligned words without explicitly defining them in the equations.

Furthermore, we scrutinized the advantage of the proposed model with respect to using continuous variables. If they are useful, we expect that the emphasis values in the middle of the range will be modeled better by the proposed method. To test this hypothesis, we split the input emphasis levels into 3 sets based on the emphasis level of the word: < 0.3, 0.3–0.6, > 0.6. Then we calculated F-measure for the *CRFs* and *LSTM_emph* on individual sets[5]. The result in Table 4.4 indicates that both systems have equivalent performance when a word is considered normal or emphasized (emphasis levels below 0.3 or over 0.6), but when the emphasis levels fall between 0.3-0.6, *LSTM_emph* outperformed the *CRFs*. This demonstrates the limitation of the *CRFs*, which require emphasis level quantization to handle continuous variables, but LSTMs do not.

[5]Because the accuracies of *LSTM_diff* and *LSTM_emph* are similar, below we only show the results of *CRFs* and *LSTM_emph*.

Table 4.4. F-measure for CRF and LSTM_emph emphasis translation on different input emphasis levels.

<0.3		0.3–0.6		>0.6	
CRF	LSTM	CRF	LSTM	CRF	LSTM
88.05	87.69	70.85	81.41	92.53	92.75

4.5.5 Hard-attentional models: subjective evaluation

Finally, we performed a subjective evaluation to verify whether human listeners can perceive the same improvement between *CRFs* and *LSTM_emph* as in the objective evaluation. We used the "En1" test set for this evaluation.

We obtained a result of 83.0% for *LSTM_emph* and 81.0% for *CRFs* indicating that humans perceived a slightly smaller improvement compared to the objective result. Moreover, the *CRF* system's performance dropped with a smaller margin (3.70%) than the proposed method (5.82%). The reason is because in the *LSTM_emph* approach, 268 emphasized words were recognized correctly in the objective evaluation, but 14 of them having emphasis levels fall between 0.5-0.8 are mis-recognized by human listeners while this does not happen in the *CRF* approach since these emphasis levels are just slightly higher than the threshold, leading to slightly emphasized synthetic speech that is hard to perceive by human listeners. In the *CRF* approach, the emphasis levels are quantized into buckets of {0, 0.3, 0.6, 0.9, ...}, so when a word is considered as emphasized (larger than the threshold 0.5), the distance to the threshold is usually large.

4.5.6 Effect of using emphasis as additional features on standard NMT systems

Even though previous works translated emphasis weights separately from NMT, no analysis has addressed whether emphasis weights in NMT have a positive or negative effect. Such analysis, however, is important before integrating emphasis translation into NMTs. To address ths oversight, we explored the effect of emphasis as an input feature on machine translation performance.

We kept the same decoder structure like standard NMT systems so that no em-

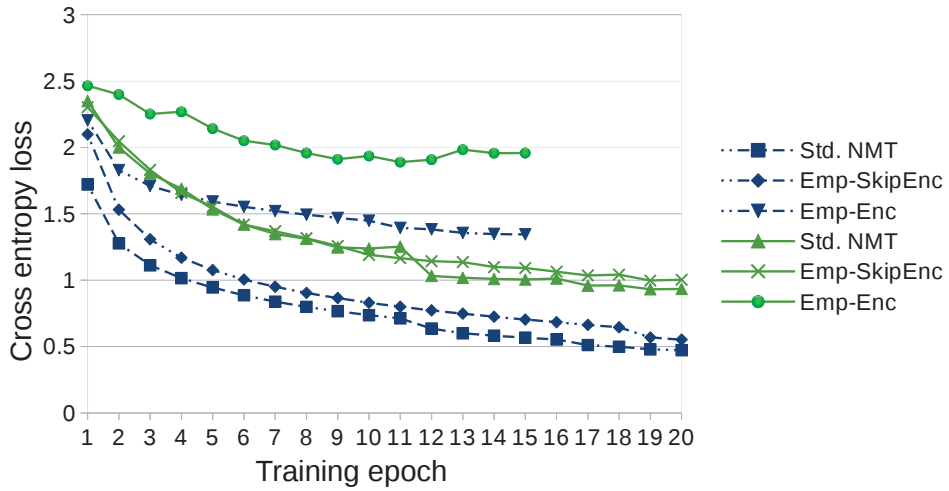


Figure 4.11. Effect of emphasis on standard NMT systems. The solid and dash lines denote MT performance on development and the training sets, respectively.

phasis prediction is performed and evaluated two encoders with emphasis weights added in different positions as described in Section 4.4.3. The baseline is the standard NMT system without emphasis weights (*Std. NMT*). Fig. 4.11 shows the result of the cross entropy loss of word prediction performance on the training and development sets. The loss is higher in both approaches (*SkipEnc* and *Emp-Enc*) than *Std. NMT*, indicating that emphasis did not improve the NMT performance. We hypothesize that such a negative effect is due to the fact that emphasis weights are paralinguistic while NMT is translating linguistic information. Only using emphasis weights as an additional feature without translating is insufficient for the model to learn anything useful from emphasis.

Although the NMT performance was degraded when using emphasis weight features, *SkipEnc* has a minimal effect compared with *Emp-Enc*. This is because in *SkipEnc*, the encoder avoids excessive influence from the negative effect of the faked emphasis weights; therefore, we can preserve the performance of the standard NMT. The rest of our experiments used the *SkipEnc* model.

4.5.7 Joint translation models: Effect of PoS tags on ET and MT models

Figure 4.12 shows the performance of the ET model using the *EmpEnc* join translation approach with and without the PoS tag feature. With PoS tags, the model converges faster and provides better performance in the first 10 iterations. But both systems eventually converge to a similar point when we train them for 15 iterations. We also observed the same tendency in the MT task (Fig. 4.13). The result indicates that PoS tags still help the translation model, but if we train it on a sufficient amount of iterations, its help is minimalized. We hypothesize that this is because the model can learn semantic meaning of a word that is similar to what the PoS tags represent.

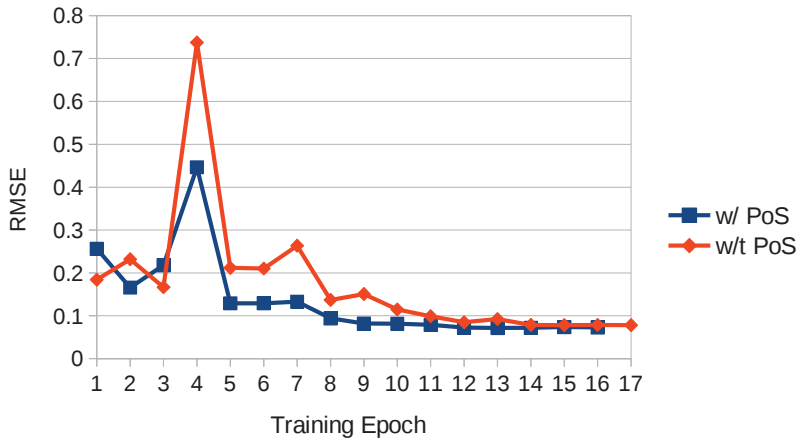


Figure 4.12. ET performance in joint translation models on a development set with/without PoS tags.

4.5.8 Joint translation models: Emphasis translation performance

From the result of the above sections, we conducted the following experiments using *SkipEnc* architecture without PoS tag features and completely trained the model for both emphasis and word prediction. Fig. 4.14 shows the F-measure,

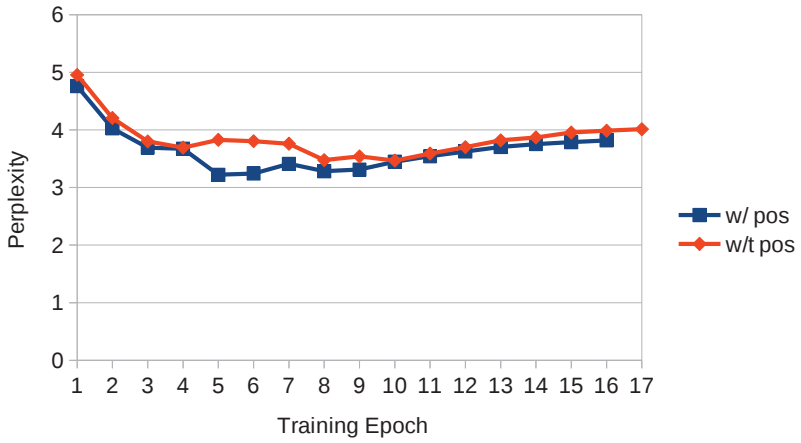


Figure 4.13. MT performance in joint translation models with/without PoS tags.

precision, and recall for emphasis prediction using the *SkipEnc* encoder with *baseline* and *residual* decoders.

Looking at the F-measure, the *residual* decoder outperformed the *baseline* decoder by a 2.7% F-mesure. The *baseline* decoder's precision, however, is higher than of the *residual* one, indicating that the *residual* connection makes more mistakes that predict high emphasis weights for normal words. Similarly, the high score for the *residual* decoder's recall indicates that it preserves more emphasized words than the *baseline* system.

The contrastive precision and recall performance of the two systems indicates that better performance can be gained by combining them. In the next section, we describe our combination technique and compare its result with previous works.

4.5.9 Joint translation models: model combination for emphasis translation

The model combination works as follows. First, we performed emphasis translation on the development set and calculated the precision and recall scores. Then, for content words, we selected the emphasis weights predicted from the system with higher recall, and for the non-content words, we selected emphasis weights with lower recall.

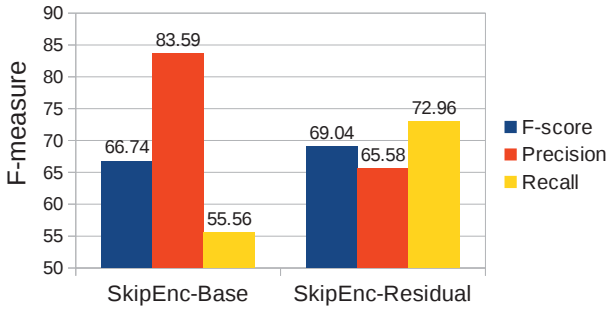


Figure 4.14. Emphasis translation performance in joint translation model

We also performed emphasis translation using previous approaches based on conditional random fields (CRFs) [18] and LSTM hard-attention models [65]. The input features for these approaches are words and emphasis weights that resemble the proposed approach. The result is shown in Fig. 4.15. Compared with CRFs, our proposed approaches perform bettered with a $\sim$5% F-measure and have a closed performance with the LSTM hard-attention approach with a $\sim$2% lower F-measure.

The result matches our expectation because both the CRFs and LSTM hard-attention approaches use ground-truth one-to-one word alignments and have independent words and emphasis translation models. On the other hand, our proposed approaches do not require word alignment models and can translate words and emphasis twice as fast as hard-attention models.

4.5.10 Joint translation models: machine translation performance

We evaluated the machine translation performance with various depths of hidden layers. The baseline system is the standard NMT without emphasis weights used in both the encoder and decoder. As shown in Table 4.5, with hidden layer depths of 1 and 2, the performance different of the proposed approach and the baseline is negligible, indicating that optimizing the model with emphasis weights can compensate for the negative effect of emphasis found in Section 4.5.6.

With a hidden layer depth of 3, all of the models seem to be over-fitted with the training samples, resulting in the loss of performance. However, interest-

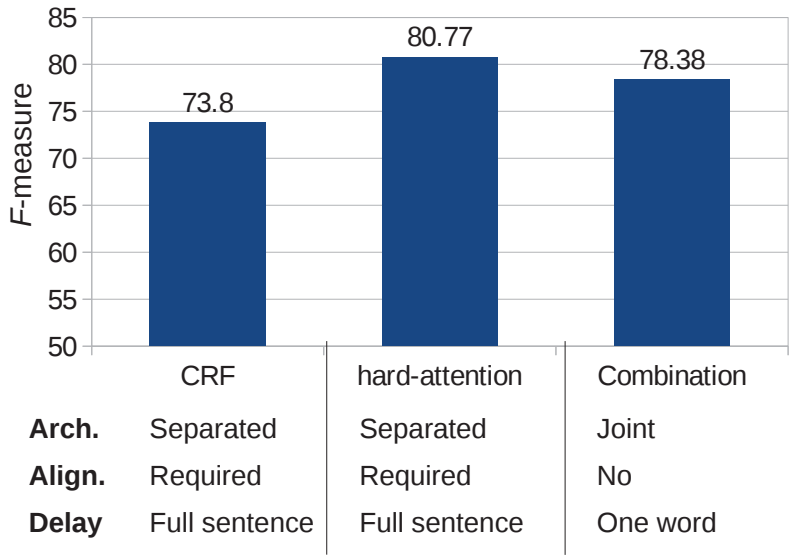


Figure 4.15. Comparison of emphasis translation performance of proposed and previous approaches. Graph also shows differences in terms of translation architecture (Arch.), word alignment requirement (Align.), and the translation delay (Delay).

ingly, the proposed approaches have smaller drops in performance. Specifically, the *SkipEnc-Residual* approach only dropped ∼1% of BLEU, while the baseline system without emphasis weights dropped ∼3% of BLEU.

We hypothesize that emphasis weights work as regulation parameters that help preventing over-fitting. In particular, they might help to enhance the attention vectors during the decoding process. The intuition is that a pair of content words in the source and target languages that have the same meaning are often also having the high values of emphasis (this is by the design of the corpus). To support this hypothesis, we conducted an analysis on the content words between references,the result is shown in Figure. 4.16. As we can see, the NMT system trained with emphasis (red boxes) have closer number of content words to the references compared with the one trained without emphasis.

Table 4.5. Machine translation performance in joint translation model. Various depths of hidden layers denoted as *d(1,2,3)* were evaluated.

System	**BLEU**
Baseline (d1)	27.67
SkipEnc-Base (d1)	27.25
SkipEnc-Residual (d1)	27.19
Baseline (d2)	27.44
SkipEnc-Base (d2)	27.70
SkipEnc-Residual (d2)	27.72
Baseline (d3)	23.68
SkipEnc-Base (d3)	**25.41**
SkipEnc-Residual (d3)	**26.36**

4.6. Discussion

In this chapter, we proposed methods to accurately translate emphasis, and reduce translation complexity. Unlike previous work where emphasis is considered to be discrete labels and has difficulty handling long-term dependencies, our proposed hard-attention seq-to-seq model can solve both problems in a single model

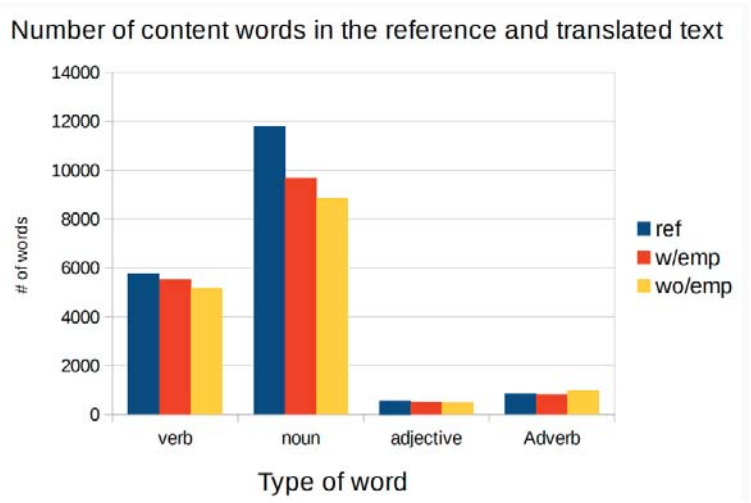


Figure 4.16. Content words counts comparison between references and hypotheses. The NMT system trained with emphasis (red boxes) have closer number of content words to the references compared with the one trained without emphasis. This give a hint that emphasis help to enhance the attention vectors between content words.

by utilizing the LSTM-based encoder-decoder that can capture long-term dependencies and handle continuous emphasis in its objective function. The evaluation on emphasis translation task demonstrates that our model can translate emphasis significantly better than previous work.

With regards to effect emphasis and PoS tags on a machine translation task, we discovered that emphasis does not help standard MT systems if it is simply used as an additional feature. Experiments with PoS tags also showed that it helps the model converge faster, but it does not help improve the accuracy if the model is well-trained. Another important outcome of our result is that our proposed model can learn good features from words and emphasis without PoS tag dependencies.

Our work on the joint translation of words and emphasis demonstrated that our proposed joint translation model can accurately translate emphasis and words with one-word delay, but the previous work requires a full-sentence delay. The model significantly reduced the complexity by removing word alignments and PoS tag features. We also found that emphasis can help MT performance prevent

over-fitting.

Future work will integrate ES and ASR to completely remove any dependencies introduced by adding emphasis translation to standard S2ST systems. Joint training the whole system is another very interesting topic. Thanks to the seq-to-seq model, we can apply it to all components to seamlessly integrate them into a joint translation model. In addition, recent works on speech recognition have shown that TDNN is comparable (or even better in certain cases) with LSTM. Integrating these models into emphasis estimation and translation will further reduce the model complexity and potentially speed up the translation pipeline.

Chapter 5

Translation of Emphasis Acoustic and Linguistic Features

Previous sections have described emphasis speech translation systems taking into account acoustic features. However, their limitation is that they are oblivious to the emphasis-to-text translation scenario (linguistic features), where emphasis can be preserved in the text form in the target language instead of speech.

An example of 2 ways emphasis translation is shown in Fig. 5.1 where the source language speaker expressed a very strong feeling in the word "atsui", which means hot. One possible way to translate such information is translating the acoustic information of the word "atsui" to the word "hot", or even better, adding the word "totemo" which means "very" to express the strong feeling. This is the situation where the system takes into account linguistic features.

To model such E-S2ST systems, it is important to understand how emphasis is expressed in both text and speech. However, most of studies related to emphasis analyses and modeling focus on only either speech or text. Su [66] and Bennett et al. [67] conducted studies on the use of different intensifiers, while the works in [68] and [69] used speech corpora to assess emotion and personality. Most of emphasis speech corpora also contains emphasis expressed in binary values only [62, 36, 14]. This is the main reason for the lack of emphasis analyses with both text and speech.

In an attempt to tackle the problem, this section presents an effort to design a text corpus that contains various emphasis levels manually annotated by humans.

Speech data is also recorded by participants to reflect the same emphasis levels as the perceived in the text. The corpus is later used to conduct human evaluations where we evaluate their perception of different levels of emphasis in both speech and text. We are particularly interested in the ambiguities of the boundaries between these levels. In addition, we also evaluate whether participants can perceive the same emphasis level across text and speech.

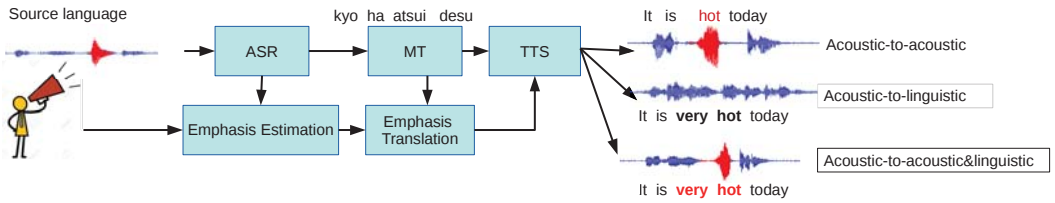


Figure 5.1. An example illustrating various ways to translate emphasis information from one language to another.

5.1. Emphasis in Text

Emphasis in text (a.k.a intensities) is manifested by using intensifiers [70, 71]. For instance, the sentence "it is hot today," does not contain any other information rather than the temperature. However, if we use adverbs to modify the meaning of the word "hot", for example, "it is very hot today", then readers can perceive not only temperature information but also the feeling of the writer. These adverbs are also known as intensifiers used to scale the meaning of a sentence upwards or downwards in term of degrees of emphasis [72]. Su [66] conducted experiments on the practical use and the semantic meaning of the 4 commons intensifiers including quite, pretty, rather, and fairly. While Ito et al. [73] examined intensifiers in a corpus from a socially and generationally stratified community.

According to Kirk [70], intensifiers are classified into 3 categories including "doubtful" (using diminisher adverbs such as slightly, a bit), "strong" (using moderato words such as quite), and "very strong" (using booster words such as extremely). Examples are shown in Table 5.1. The intensifiers are often used to modified the meaning of nouns, verbs, and adjectives with one condition that

those words must be gradable or be measurable in terms of quantity [72, 74]. For example, words such as “hot”, “cold”, “hurt” are gradable while “I”, “you”, or “house” are non-gradable and cannot be emphasized.

Table 5.1. Examples of different level of emphasis in text using intensifiers.

Level	**Example sentence**
Doubtful	It is a little bit hot
Neutral	It is hot
Strong	It is quite hot
Very strong	It is extremely hot

Table 5.2. Examples of emphasis expressed in text form

Level	**Example**	**Perception**
Doubtful	This is a tiny bit unreasonable. Yes, it has shown a mildly remarkable increase in population during the last ten years	Doubtful Somewhat-strong
Somewhat-strong	The doctor’s quick arrival brought about her fairly speedy recovery. I was significantly distressed to be suspected of wrongdoing when I was innocent.	Somewhat-strong Very strong
Strong	We are very busy and considerably short-handed. Super short man of about thirty.	Strong Very strong
Very strong	I think the bill is perfectly accurate. He is an unforgivably tight-lipped man.	Very strong Strong

5.2. Emphasis in Speech

Unlike emphasis in the text that has been classified into 3 classes, studies of emphasis in speech are often consider it as binary values and is produced at word-

level [36, 35]. Many analyses and corpus collection have also been conducted to find how humans produce emphasis by changing pitch, duration, and power. Do et al. [62] has designed a corpus by asking speakers to utter an utterance with a pre-defined word that is marked to be emphasized and all other words are marked as neutral. The corpus is suitable to study emphasis in misheard situations where only one word is emphasized. However, in normal conversation, emphasis might be expressed in various degrees depending on how speakers want to express their idea. It is also possible that not only the important word but also the adjacent words should be emphasized to preserve the naturalness. Therefore, constraining speakers to put emphasis on a single word does not reflect real conversation scenarios.

5.3. Correlation of Intensity and Emphasis: A Data-driven Approach

As described above, emphasis has been studied separately between text and speech representations. However, it is crucial to understand humans perception of emphasis in both forms in order to accurately translate it across languages. To do that, we need a corpus with emphasis speech expressed in the same way in both text and speech.

5.3.1 Text design

The text data is constructed from a large amount of sentences. We first perform part-of-speech tagging to find sentences with at least one adjective. In this paper, we focus on adjectives because it is easier to find gradable adjectives than nouns or verbs. After filters out sentences without adjectives, we manually select a set of 1050 sentences for annotators to create their emphasized version.

As described in the previous section that 3 levels of emphasis in text including “doubtful”, “strong” and “very strong”. However, because the emphasis levels in speech have not defined yet, it is possible that in speech emphasis can be expressed at finer levels. Therefore, we added another level called “somewhat-strong” that is a bit stronger than neutral but less than “strong” levels.

To created emphasized text sentences, annotators are asked intensifying the original sentence to a certain level. We also ask them to be creative and use as many as possible intensifiers to produce more varieties of emphasis.

5.3.2 Audio recording

Similar to the text data collection, speakers need to record audio of the same utterance but with different emphasis expressed on a marked word. However, unlike the text collection, the speakers do not know which emphasis level they need to express, this is to mitigate any bias that speaker will be forced to expressed exactly the same emphasis degrees as in text. They are instead given a pair of text sentences including an original sentence and its emphasized version, and are instructed to figure out the emphasis level from the emphasized text and utter the original sentence with the same emphasis level they perceived (as illustrated in Fig. 5.2).

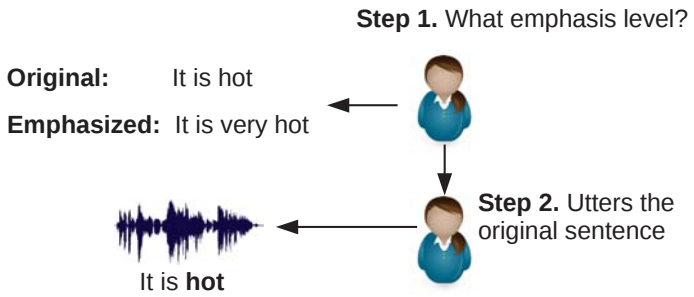


Figure 5.2. An example of audio recording without pre-defined emphasis levels.

The reason why we asked speakers to utter the original instead of the emphasized sentence is because we want human perception of emphasis on speech will not be influenced by text. This is important because if emphasis is expressed in both text and audio forms in the same sentence, humans perception will be affected by both, leading to the difficulties in analyzing the correlation of emphasis in text and speech.

5.3.3 Experiments

The corpus

We have collected 1050 original (neutral emphasis level) and 4200 emphasized (1050 x 4 levels) English sentences annotated by a native English speaker. A total of 7750 audio files expressing different emphasis degrees of the same sentence have been recorded by 2 native English speakers. This corpus is used for the following emphasis perception evaluations in text and speech.

Additionally, we have also collected Japanese translated version of the 5250 English sentences for the purpose of analyzing emphasis linguistic features across languages.

Human Perception Evaluation Approach

To analyze the change of acoustic features when humans produce different emphasis levels, we first perform automatic audio-text alignment to get word boundaries information. The, we calculate acoustic features including F_0, duration, power, and pauses used before and after important words.

Then, to evaluate human perception on emphasis expressed in text and audio, we conducted three crowded-source evaluation tasks as follows,

- The audio task: only audios with different emphasis levels are given to participants. The participants, then, select an emphasis level that they perceived in the audio. The goal of this experiment is to find out to what extent human can correctly perceive emphasis levels and also to investigate the emphasis perception ambiguities.

- The text task: the task description and goal are similar to the *audio task* except we only give texts with different emphasis levels to participants.

- The audio & text task: in this task, we give participants both audio and text and they will decide whether or not they can perceive the same emphasis level. We hypothesize that when the same emphasis level between speech and text is given, participants can perceive the similarity. However, there might be also ambiguities among different emphasis levels as well.

We utilized the crowdflower platform to run our crowded-source evaluation, participants are selected from English speaking countries and with a minimum quality level of 2[1]. All results presented in the following subsections are calculated using samples that have a confidence score equal or higher than 0.6[2]

With regards to the instruction of each task, we gave to the participants the same guideline for both the *audio task* and *text task* with a definition of 5 emphasis levels and their audio and text examples, respectively. In the *audio & text* task, we did not give participants examples of the same or different emphasis in audio and text. Instead, we gave them separate examples as in the *audio task* and *text task*. By doing this, participants will have some knowledge of emphasis in audio and text separately (the purpose is to mitigate any bias) and it is up to them to figure out what is similar and different emphasis levels in speech and text.

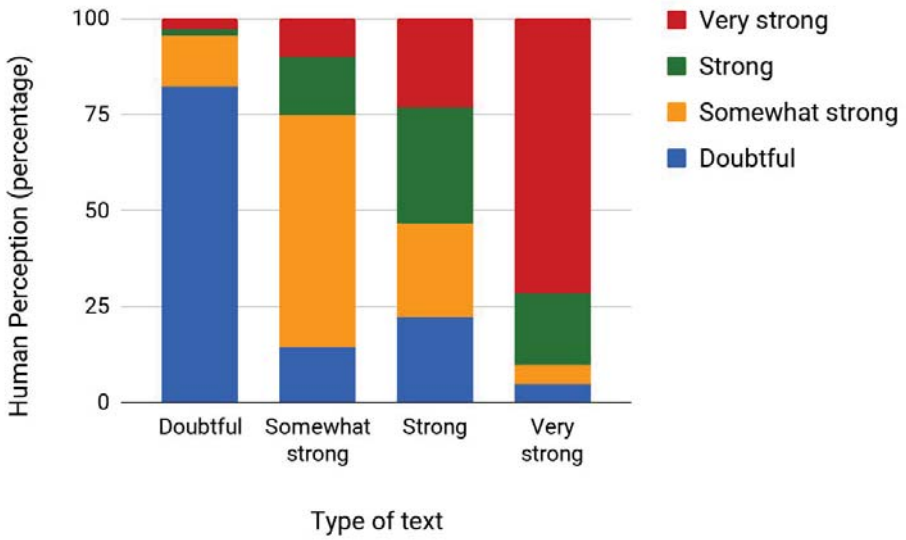


Figure 5.3. Human perception on emphasis with only text clues. The horizontal axis shows the ground-truth labels while the bars show human perception for each emphasis level.

[1]The quality level is given to participants based on their experience and performance in the past. It ranges from 1 to 3 with 1 is the beginner and 3 is the advanced level.

[2]Each sentence is judged by 3 participants, a confidence score above 0.6 means at least 2 participants made the same decision.

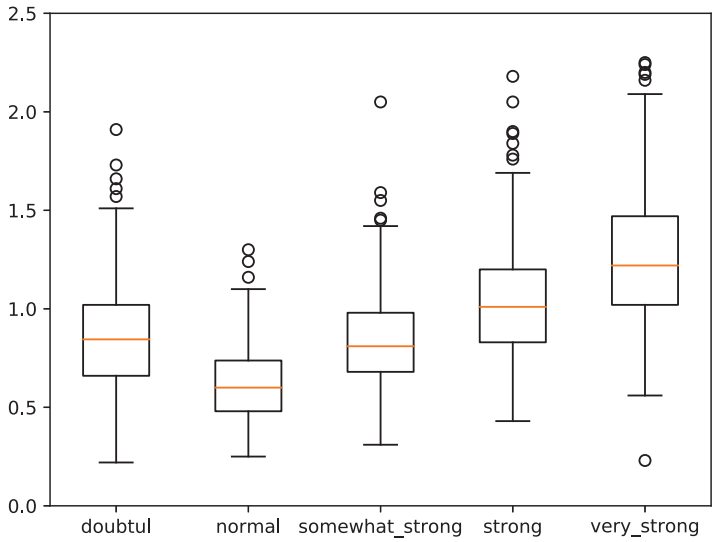


Figure 5.4. Duration distribution of the important words

Figures 5.4, 5.5, and 5.6 show the distribution of words F0, duration, and power. As we can see, the duration and power distribution shows a very clear distinction between emphasis levels. It is interesting to see the duration distribution of the "doubtful" and "somewhat strong" levels look alike. The reason is that when expressing hesitation as in the "doubtful" level, speakers often stretch the duration of the word a bit longer than normal. This behavior is similar with the "somewhat strong" level where we want to express something just a little bit stronger than normal.

The F0 distributions are, however, similar between levels. This observation surprised us as we expected to see differences here. By manually listening to audio samples, we found that the F0 patterns are indeed different across levels but at phrase or sentence levels, not very clear at the word level.

Figures 5.7 and 5.8 shows the preceding and succeeding pauses duration distributions. Generally speaking, to decrease or increase the emphasis level, there is a short pause inserted on both side of a word, and its duration becomes longer

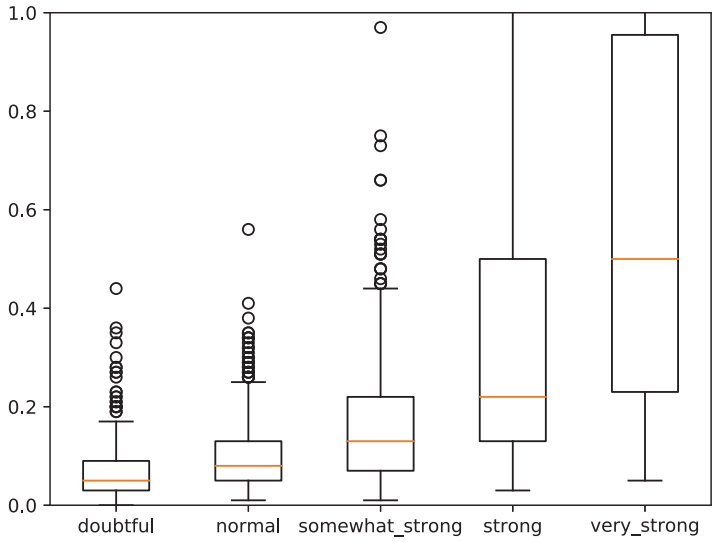


Figure 5.5. Power distribution of the important words.

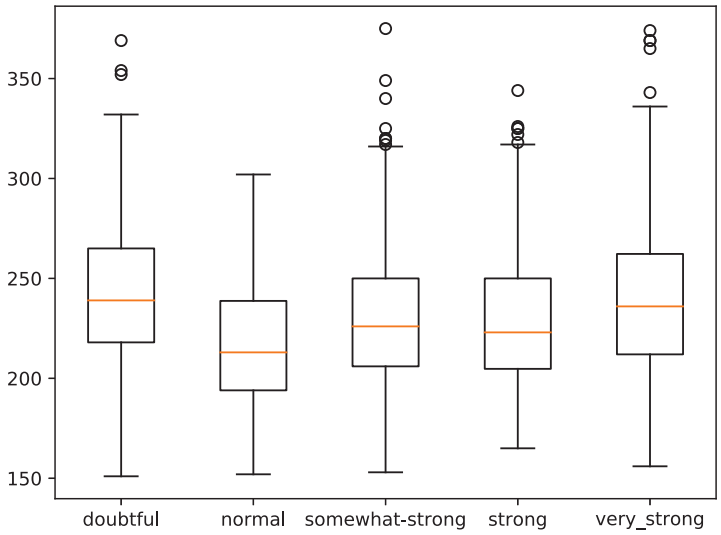


Figure 5.6. Average F0 distribution of the important words.

as the emphasis level increased. The result also showed an interesting characteristic of the "doubtful" level as its preceding pauses duration is exceptionally much longer than other levels and its succeeding level. This long preceding pause play an important role to express "doubtful" feeling as a signal that tell other interlocutors in a conversation that next word (information) is uncertain.

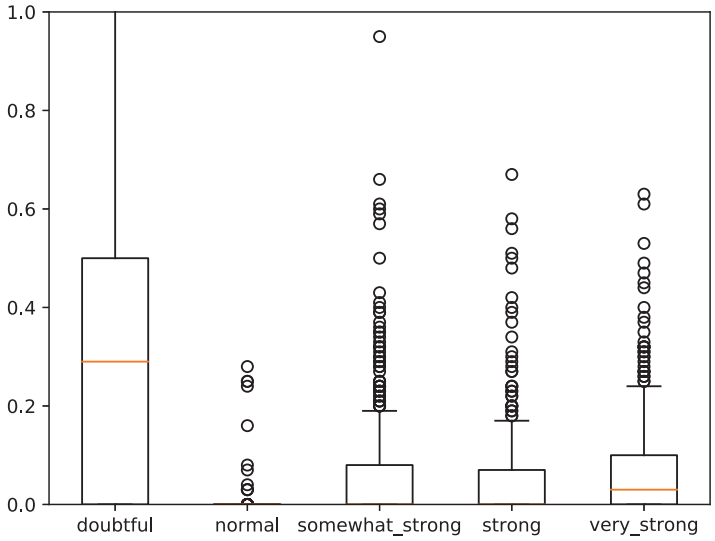


Figure 5.7. Preceding Pauses Duration Distribution

Emphasis perception with text clues

The result in Fig. 5.3 showed that participants can perceive emphasis accurately when doubtful, somewhat-strong and very strong emphasis texts are presented (column 1, 2 and 4). In particular, 82.18% doubtful, 60.12% somewhat-strong and 71.57% very strong texts are predicted correctly. The strong emphasis level (column 3), however, poses more difficulties and ambiguities to participants as they made more mistakes. We observed more than 50% strong texts are predicted as somewhat-strong and very strong.

Table 5.2 shows examples of each emphasis level as well as the correct and incorrect perception. As we can see, the second sentence of the somewhat-strong

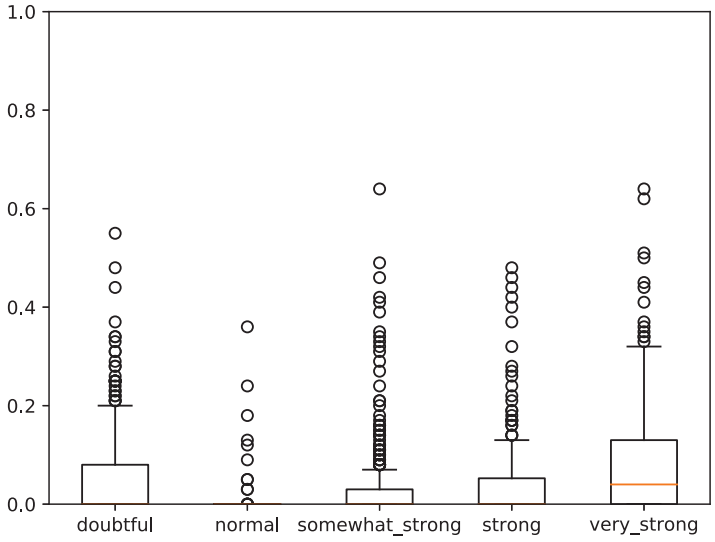


Figure 5.8. Succeeding Pauses Duration Distribution

and strong levels, where participants perception is very strong, are more likely a very strong level. This observation indicates that the low prediction performance on the strong level is not only due to human perception but also because of the way annotators emphasize a neutral sentence. In other words, it is not always possible to express a certain emphasis level by adding adverbs to the sentence.

Emphasis perception with audio clues

In this experiment, only audios with emphasis are presented to participants. The data consists of 5000 audio files including 5 emphasis levels spoken by 2 native English speakers. The result is shown in Fig. 5.9. As we can see, the ambiguity when using audio to express emphasis is generally less than using text. In particular, 54.46% of strong and more than 70% of other emphasis levels audios are perceived correctly by participants.

However, although the strong emphasis level is perceived better using audio clues, there are still 46.54% are misrecognized, mostly to the adjacent levels including somewhat strong and very strong.

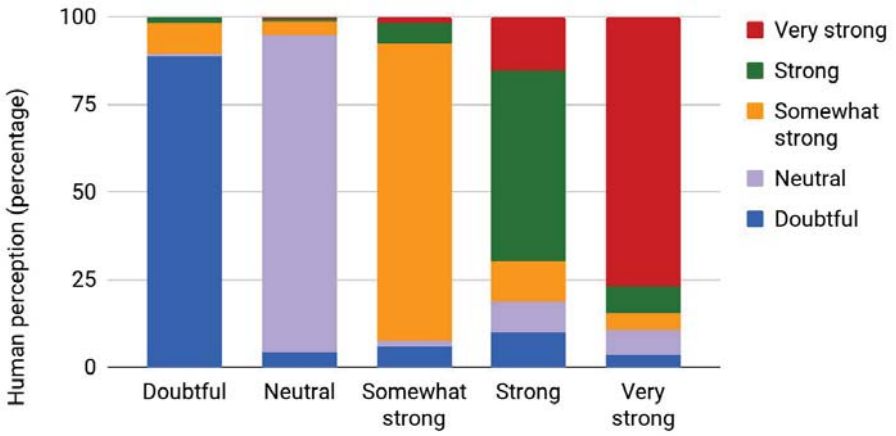


Figure 5.9. Human perception on emphasis with only audio clues

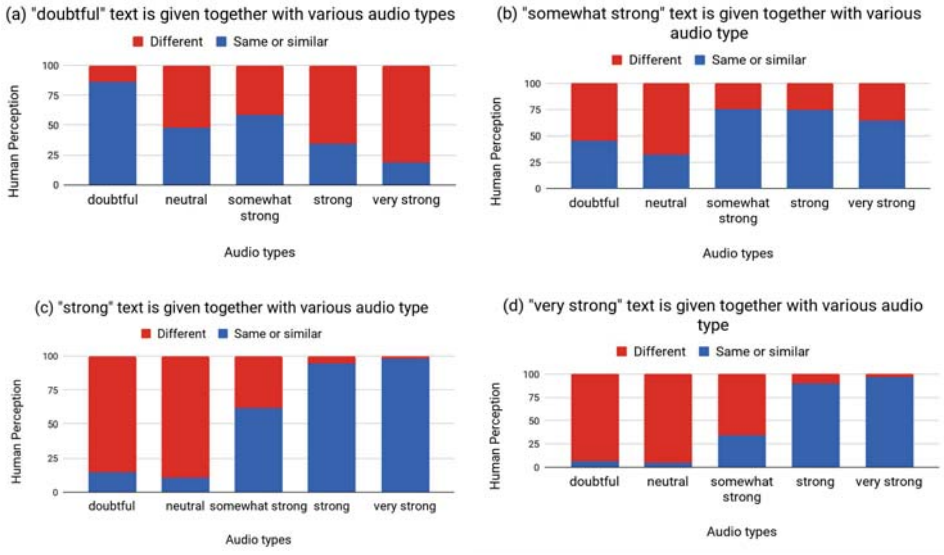


Figure 5.10. Human perception on emphasis with both audio and text clues

Emphasis perception with both text and audio clues

In this experiment, we focus deeper on the correlation of emphasis perception in text and audio. We gave participants pairs of audio and text and they decide whether or not they perceive the same emphasis level.

To capture more information and varieties, we build the dataset to include not only the same but also different emphasis levels between text and audio. For example, the doubtful level text is paired with doubtful, neutral, somewhat strong, strong and very strong audios, and so on. In total, we produced 5000 text-audio pairs spoken by 2 native speakers. The results are shown in Fig. 5.10 (a, b, c, d) with each sub-figure shows the result of one specific text type paired with all other audio types. From Fig. 5.10 (a) we can see that participants can perceive the same emphasis level most of the time when both doubtful text and audios are given. We also observed the same tendency for other emphasis levels as well in sub-figures b, c, and d. These observations support our hypothesis that the same emphasis levels should be perceived in the same way.

Moreover, we are also interested in the emphasis perception ambiguities. In Fig. 5.10 (a), we can see that participants not only perceive similar emphasis level for doubtful texts and doubtful audios, but also think doubtful texts and somewhat strong audios are quite similar. In particular, 58.70% of this pair are perceived as similar. Looking back at the results in Fig. 5.9 (first column) and Fig. 5.3 (third column) we hypothesize that this ambiguities is more likely due to the difficulty in expressing doubtful text as participants classified 13.22% of doubtful text as somewhat strong while only 8.74% of somewhat-strong audio are missed classified as doubtful.

The sub-figures b and c also show ambiguities between somewhat strong and strong emphasis levels. This is consistent with the result of the previous evaluations. Moreover, participants also think the strong text and very strong audios are similar (sub-figure d). We hypothesize that this is due to the number of emphasis levels in speech are too high that cause difficulties for speakers to produce significant difference emphasis between strong and very strong levels. To mitigate ambiguities, the number of emphasis levels in speech should also be 3 as in the text.

5.4. Acoustic-to-linguistic emphasis translation

In this section, we describe our approach to translate emphasis acoustic features into linguistic features within a language. As illustrated in Fig. 5.11, the proposed system consists of 2 components, an acoustic emphasis classification, which predict an emphasis level of given input speech; and an emphasis linguistic transformation, which takes a "neutral" sentence and the predicted emphasis level to generate an "emphasized" text sentence.

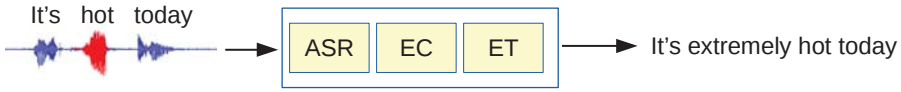


Figure 5.11. Emphasis acoustic-to-linguistic translation system.

5.4.1 Acoustic emphasis classification

The design of the acoustic emphasis classification depends on the result of the analysis on how human produce and perceive emphasis different emphasis levels described in Section 5.3.

We first focus on emphasis levels which have the least emphasis perception and production ambiguities, which are "doubtful", "somewhat strong", and "very strong". As we can see in the power distribution of emphasis words shown in Fig. 5.5, the "very strong" level is clearly distinguished from the "somewhat strong" and "doubtful" levels. The preceding pause distribution shown in Fig. 5.7, on the other hand, differentiates the "doubtful" from others. These analysis results gave us a crucial piece of information that just by using power and preceding pause duration, we might be able to train a system that can accurately predict emphasis levels given an input speech.

5.4.2 Emphasis linguistic transformation

The next step is to take the predicted emphasis level generated from the system described above and a "neutral" text output from an ASR system to generate

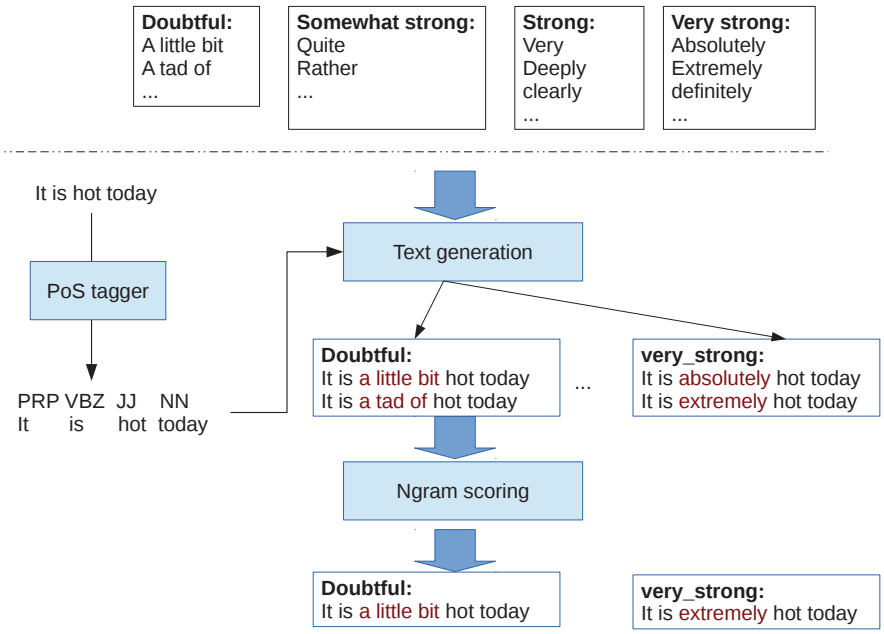


Figure 5.12. An example illustrating emphasis text transformation.

an "emphasized" (or "intensified") sentence. An example illustrating how the method works is showed in Fig. 5.12.

The idea of *intensifying* a *neutral* text is to add modifiers to the sentence that express the intensity of a certain emphasis level. The whole process is described as follows,

- **Modifier extraction**: the first step is extracting a list of modifiers (adverbs) with their corresponding emphasis classes. We do so by performing part-of-speech tagging using the Stanford PoS tagger toolkit [75] and extracting words with the adverb PoS tag.

- **Text generation**: given an input sentence and a target emphasis level, we generate all possible intensified sentences by adding modifiers before an adjective word. The modifiers are taken from the list of the input target emphasis level that is extracted in the previous step.

- **Ngram scoring**: the final step is to select the best sentence from the list of all possible intensified sentences. We do so by using a pre-trained n-gram model to calculate the perplexity of each sentence. Since the perplexity score represent for the "realistic" of the sentence, we simply pick the one with lowest perplexity.

5.4.3 Experiments

We conducted our experiments with the data collected that has been described in section 5.3. To reduce the ambiguities discovered in the previous section, the experiments are conducted with 3 emphasis levels including "doubtful", "neutral", and "very-strong".

With regards to emphasis classification task, we chose SVM to perform the task. The reason is we have seen from the analyses in the section 5.3.3 that power, word duration, and preceding pauses duration are pretty distinguish across emphasis levels, this is a hint for us that the margin maximization algorithm of SVM will perform good on this data set. While neural net based approaches can also be used, it might not perform well with a limited amount of data that we have collected.

Emphasis classification performance

To evaluate the accuracy of the emphasis classification task, we calculated F-score for each emphasis levels, which represent how accurate the system can predict the level given input acoustic features. The result is showed in Figure 5.13. Overall, the system perform pretty good with an average F-score of 87%. The very-strong level is the most easy level to predict while the doubtful level seems to be more difficult to handle.

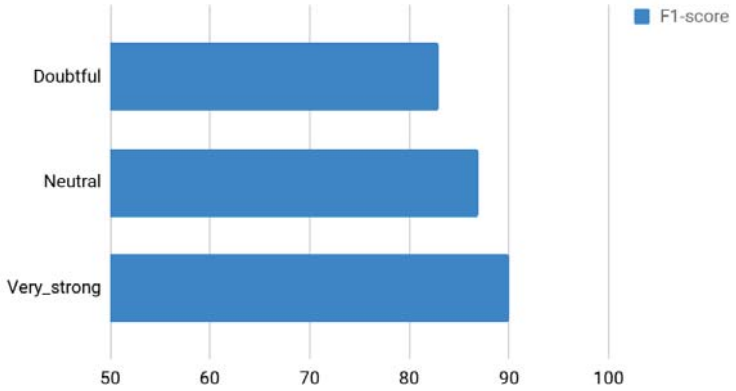


Figure 5.13. Emphasis classification performance.

Emphasis text transformation evaluation

To evaluate how well the system can generate an intensified sentence given a neutral sentence and the predicted emphasis level, we ran a crowded-source evaluation task where each participants read a pair of sentences, which includes a generated and reference sentences, and decide if they perceive the same intensity levels between them.

The number of evaluated pair of sentences are 100. We chose crowflower to run the evaluation task with participants selected from English speaking countries. The result is shown in Figure 5.14. As we can see, participants can perceive the same emphasis levels most of the time (93%). The 7% of error mainly comes from the error of the emphasis classification task.

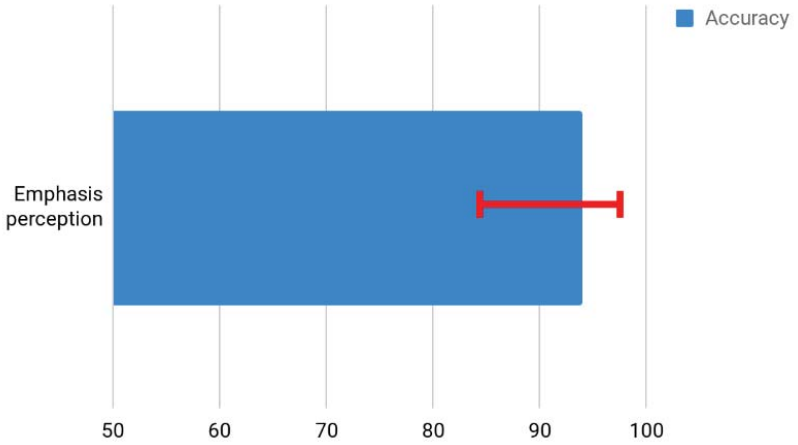


Figure 5.14. Emphasis text transformation performance. The result shown here is the human perception of emphasis between a generated sentence and the corresponding reference sentence.

5.5. Discussion

In this chapter, we presented the first English corpus that contains emphasis in both text and speech representation. Emphasis is expressed not with just binary values but with various levels. We also conducted human emphasis perception on the corpus to find out to what extent humans can clearly distinguish emphasis levels and whether they can perceive the same emphasis level in both text and speech. The results indicate that although humans can distinguish emphasis levels most of the time, there is high ambiguities of the "somewhat-strong" and "strong" emphasis levels in text, this supports the definition of 3 levels emphasis in [70] instead of 4. The analysis on speech emphasis shows less ambiguities than in text, indicating that it is easier for speakers to express emphasis in speech. Analyses on emphasis with both text and speech clues show that humans can perceive the same emphasis level between both representation, although there are still ambiguities between the "somewhat-strong" and "strong" levels.

In future, we will extend the corpus to include other languages and utilize it and the result found in this study to construct emphasis speech translation systems that can translate emphasis in both speech and text representation. By

doing so, we can handle a wider scenarios including low-resource languages where it is not feasible to collect emphasized speech data or in lecture translation where students might prefer to read instead of listen to audios.

Chapter 6

Conclusion and Future Work

6.1. Conclusion

The work presented here has addressed a number of existing emphasis modeling and translation problems and has conducted the first attempt to study the correlation of acoustic and linguistic features of emphasis considering human perception.

- The proposed continuous emphasis level modeling technique has showed the effectiveness in emphasis prediction and synthesis tasks. Specifically, our approaches outperform existing works by 2-5% F-measure of emphasis prediction and produce more natural synthetic speech. This improvement is not only important for emphasis modeling, but also help to improve the overall emphasis translation, because errors from this component will be cascaded into all downstream components in the emphasis translation system.

- With regards to emphasis translation, our proposed approaches based on seq2seq techniques can handle continuous emphasis levels and be able to jointly translate word and emphasis in a unified model. Moreover, the approach also greatly simplify the translation pipeline while still preserve the performance as in the previous complex translation model.

- The study on correlation of acoustic and linguistic features of emphasis

has discovered the patterns on how different emphasis levels are expressed with linguistic and acoustic features, as well as the ambiguities on human perceived them. These clues are important to construct a translation system that can flexibly translate emphasis expressed in different form.

6.2. Future work

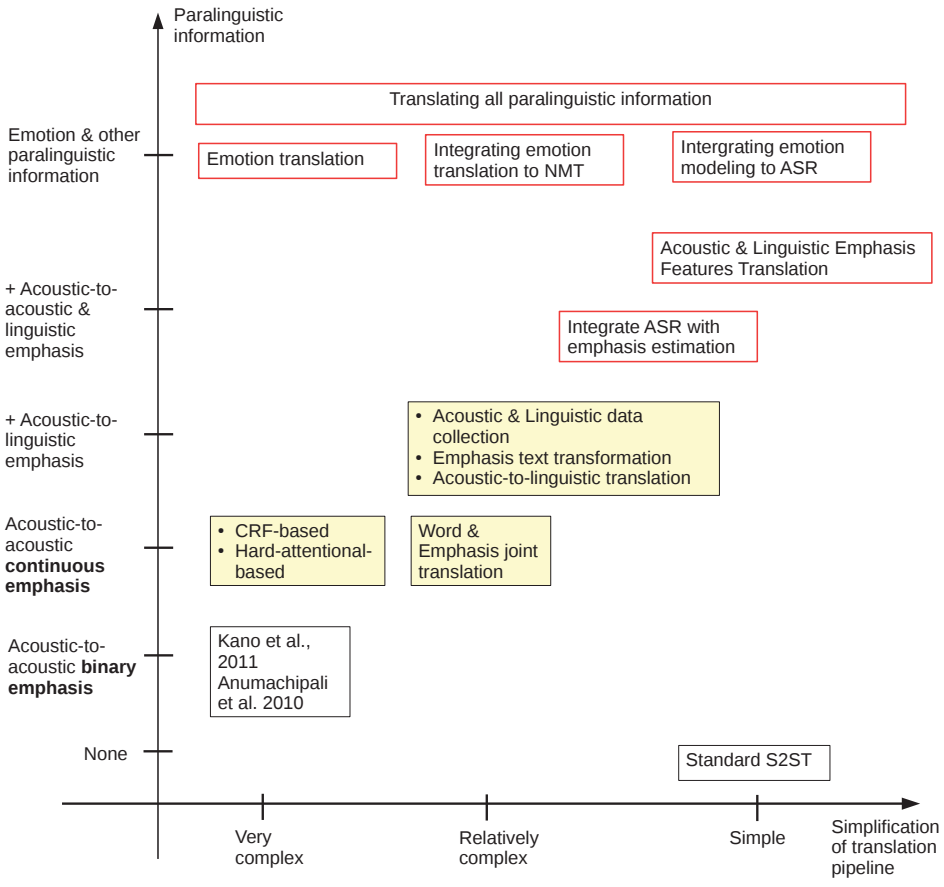


Figure 6.1. The road map for emphasis translation of this study including what has not been done (red boxes).

6.2.1 Combining emphasis estimation and speech recognition

Although our proposed joint translation model has combined machine and emphasis translation models into a single component, the emphasis estimation and speech recognition are still separated. Recent studies on machine translation have proposed approaches for real-time translation [76], which means, we can perform translation of speech in real-time with partial ASR results. If we can also combine emphasis estimation process and ASR together, we can not only further simplify the translation pipeline, but also be able to translate emphasis speech in real-time.

6.2.2 Emphasis translation considering both acoustic and linguistic feature

This thesis has handled acoustic emphasis, or emphasis that is expressed by changing acoustic features including power, duration, and F_0. However, there is another way to express the emphasis, that is changing the linguistic information. For instance, adding adverbs words to emphasize the meaning of the utterances. For instance, we can say "it is really hot today" instead of "it is hot today" to emphasize the more intensified feeling about the weather's temperature. By using the linguistic emphasis, the translation model can express the emphasis in more flexible ways.

6.2.3 Multi-speaker Emphasis Estimation

Although the approaches proposed in this work can translate emphasis across languages, they are still speaker-dependent. That means, given an unknown speaker, the error rate on emphasis estimation will be high, this error will also be cascaded into all downstream components, degrading the whole system performance. Developing a speaker-independent emphasis estimation system will allow the translation system handles more realistic scenarios.

6.2.4 Handling emphasis in natural conversation

The works presented in this thesis have handled emphasis used misheard scenarios where people often exaggerate acoustic features of words or phrases in order to express emphasis. However, in normal conversations, emphasis is also often used to convey the most important information of an utterance. In such scenarios, people do not intentionally exaggerate the word, but instead tend to use softer acoustic features so that it is just enough to let other listeners perceive the information while not affecting the naturalness of the sentence. It is also possible that more than one word are emphasized within a single utterance. It is important to take into account these use cases of emphasis to have an emphasis translation system that can fulfilling user experiences in all scenarios.

6.2.5 Incorporating more paralinguistic information

The final goal of this study is to translate all aspects of paralinguistic information, and it can not be completed without emotion, accent, and gender. Emotion is a very interesting but also difficult to handle type of paralinguistic information. The reason is the emotion can be expressed in very different way among people and also across languages, cultures. It is also sometime even hard for human to find out the emotion of the other in a conversation. Accent and gender are, generally, easier to recognize than emotion. And if we can incorporate these features into the translation system, user experiences can be taken to a new level as target language listeners might feel that they are talking to the same source language speaker.

6.2.6 Non-parallel data

Although the current system can preserve emphasis across languages, it still need parallel data to train the system. The parallel data is required for all components including ASR, MT, and TTS. This requirement leads to the problem of lack of training data. On the other hand, many researches have been conducted on utilizing non-parallel data to train the system and if we can incorporate such studies, we can utilize a large amount of training data without spending resources in collecting them.

6.2.7 Emphasis on All Content Words

This study has focused on emphasis on adjectives as it is the most frequent word that is emphasized among all other content words. However, it does not cover all usage scenarios where nouns and verbs can be emphasized, too. It is necessary to incorporate all content words in the translation system so that we can handle a wider use cases.

References

[1] Harald Hammarström. Linguistic diversity and language evolution. *Journal of Language Evolution*, 1(1):19–29, 2016.

[2] Michael A Halliday. Spoken and written language. 1989.

[3] S. Nakamura. Overcoming the language barrier with speech translation technology. *Science & Technology Trends - Quarterly Review No.31*, April 2009.

[4] A. W. Black, R. D. Brown, R. Frederking, K. Lenzo, R. Singh J. Moody, A. Rudnicky, and E. Steinbrecher. Rapid development of speech-to-speech translation systems. In *Proceedings of ICSLP*, 2002.

[5] H. Shimizu, G. Neubig, S. Sakti, T. Toda, and S. Nakamura. Constructing a speech translation system using simultaneous interpretation data. In *Proceedings of IWSLT*, Heidelberg, Germany, December 2013.

[6] Björn Schuller, Stefan Steidl, Anton Batliner, Felix Burkhardt, Laurence Devillers, Christian Müller, and Shrikanth Narayanan. Paralinguistics in speech and language–state-of-the-art and the challenge. *Computer Speech & Language*, 27(1):4 – 39, 2013.

[7] Felix Burkhardt, Jitendra Ajmera, Roman Englert, Joachim Stegmann, and Winslow Burleson. Detecting anger in automated voice portal dialogs. In *Proceedings of Interspeech*, 2006.

[8] Gilad Mishne, David Carmel, Ron Hoory, Alexey Roytman, and Aya Soffer. Automatic analysis of call-center conversations. In *Proceedings of CIKM*, pages 453–459. ACM, 2005.

[9] K. Yu, F. Mairesse, and S. Young. Word-level emphasis modelling in HMM-based speech synthesis. In *Processing of ICASSP*, pages 4238–4241, March 2010.

[10] A. Tsiartas, P. G. Georgiou, and S. S. Narayanan. A study on the effect of prosodic emphasis transfer on overall speech translation quality. In *Proceedings of ICASSP*, 2013.

[11] Q. T. Do, S. Sakti, G. Neubig, T. Toda, and S. Nakamura. Collection and analysis of a japanese-english emphasized speech corpus. In *Proceedings of COCOSDA*, Phuket, Thailand, September 2014.

[12] H. Fujisaki. Information, prosody, and modeling - with emphasis on tonal features of speech. In *Proceedings of Speech Prosody*, pages 1–10, 2004.

[13] B. Arons. Pitch-based emphasis detection for segmenting speech recordings. In *Proceedings of ICSLP*, pages 1931–1934, 1994.

[14] L.S. Kennedy and D.P.W. Ellis. Pitch-based emphasis detection for characterization of meeting recordings. In *Proceedings of ASRU*, pages 243–248, November 2003.

[15] T. Kano, S. Takamichi, S. Sakti, G. Neubig, T. Toda, and S. Nakamura. Generalizing continuous-space translation of paralinguistic information. In *Proceedings of INTERSPEECH*, pages 2614–2618, 2013.

[16] P. D. Aguero, J. Adell, and A Bonafonte. Prosody generation for speech-to-speech translation. In *Proceedings of ICASSP*, volume 1, 2006.

[17] G. K. Anumanchipalli, L. C. Oliveira, and A. W. Black. Intent transfer in speech-to-speech machine translation. In *Proceedings of SLT*, pages 153–158, Dec 2012.

[18] Q. T. Do, S. Takamichi, S. Sakti, G. Neubig, T. Toda, and S. Nakamura. Preserving word-level emphasis in speech-to-speech translation using linear regression HSMMs. In *INTERSPEECH*, 2015.

[19] Leonard E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73(3):360–363, 05 1967.

[20] J. Baker. The DRAGON system–an overview. *IEEE*, 23(1):24–29, Feb 1975.

[21] G. Hinton, L. Deng, D. Yu, A.R. Mohamed, N.Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. S. G. Dahl, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition. *IEEE*, 29(6):82–97, November 2012.

[22] G. Neubig. Travatar: A forest-to-string machine translation engine based on tree transducers. In *Proceedings of ACL*, pages 91–96, Sofia, Bulgaria, August 2013.

[23] P. Koehn. *Statistical Machine Translation*. Cambridge University Press, 1st edition, 2010.

[24] F.J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[25] G. Neubig, T. Watanabe, E. Sumita, S. Mori, and T. Kawahara. An unsupervised model for joint phrase alignment and extraction. In *Proceedings of ACL*, HLT '11, pages 632–641, 2011.

[26] A.J. Hunt and A.W. Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *Proceedings of ICASSP*, volume 1, pages 373–376 vol. 1, May 1996.

[27] Heiga Zen, Andrew Senior, and Mike Schuster. Statistical parametric speech synthesis using deep neural networks. In *ICASSP*, pages 7962–7966. IEEE, 2013.

[28] H. Zen, K. Tokuda, T. Masuko, T. Kobayasih, and T. Kitamura. A hidden semi-Markov model-based speech synthesis system. *IEICE*, E90-D(5):825 – 834, 2007.

[29] E. Eide, A. Aaron, R. Bakis, W. Hamza, M. Picheny, and J. Pitrelli. A corpus-based approach to <AHEM/> expressive speech synthesis. In *Proceedings of ISCA*, pages 79–84, 2004.

[30] A. W. Black. Unit selection and emotional speech. In *Proceedings of EUROSPEECH*, pages 1649–1652, 2003.

[31] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai. Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm. *IEEE*, 17(1):66–83, January 2009.

[32] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura. Speech parameter generation algorithms for HMM-based speech synthesis. In *Proceedings of ICASSP*, volume 3, pages 1315–1318 vol.3, 2000.

[33] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. In *Proceedings of SSW*, page 125, 2016.

[34] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi. Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing. *IEICE*, 88(11):2484–2491, 2005.

[35] K. Morizane, K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano. Emphasized speech synthesis based on hidden Markov models. In *Proceedings of COCOSDA*, pages 76–81, August 2009.

[36] K. Yu, F. Mairesse, and S. Young. Word-level emphasis modelling in HMM-based speech synthesis. In *ICASSP*, pages 4238–4241. IEEE, 2010.

[37] M.J.F. Gales. Cluster adaptive training of hidden Markov models. *IEEE*, 8(4):417–428, 2000.

[38] Q. T. Do, S. Takamichi, S. Sakti, G. Neubig, T. Toda, and S. Nakamura. Preserving word-level emphasis in speech-to-speech translation using linear regression HSMMs. In *Proceedings of InterSpeech*, September 2015.

[39] T. Nose, J. Yamagishi, T. Masuko, and T. Kobayashi. A style control technique for HMM-based expressive speech synthesis. *IEICE*, E90-D(9):1406–1413, September 2007.

[40] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura. Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. In *Proceedings of Eurospeech 1999*, 1999.

[41] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura. Speech synthesis based on hidden Markov models. *IEEE*, 101(5):1234–1252, 2013.

[42] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *The royal statistical society*, 39(1):1–38, 1977.

[43] H. Kawahara, I. Masuda-Katsuse, and A.de Cheveigné. Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F_0 extraction: Possible role of a repetitive structure in sounds. *Speech Communication*, 27(3-4):187–207, 1999.

[44] John D. Lafferty, A. McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, pages 282–289, 2001.

[45] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, pages 93–128, 2006.

[46] J. Nocedal. Updating Quasi-Newton Matrices with Limited Storage. *Mathematics of Computation*, 35(151):773–782, 1980.

[47] N. Okazaki. Crfsuite: a fast implementation of conditional random fields (CRFs), 2007.

[48] T. T. Nguyen, G. Neubig, H. Shindo, S. Sakti, T. Toda, and S. Nakamura. A latent variable model for joint pause prediction and dependency parsing. In *Proceedings of InterSpeech*, Dresden, Germany, September 2015.

[49] V.K.R. Sridhar, J. Chen, S. Bangalore, and A. Conkie. Role of pausing in text-to-speech synthesis for simultaneous interpretation. In *Proceedings of ISCA Workshop on Speech Synthesis*, 2013.

[50] J. Tauberer. Predicting intrasentential pauses: Is syntactic structure useful? In *Proceedings of the Speech Prosody*, pages 405–408, 2008.

[51] A. Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013.

[52] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Proceedings of ICASSP*, pages 6645–6649. IEEE, 2013.

[53] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Proceedings of NIPS*, pages 3104–3112, 2014.

[54] M. T. Luong, H. Pham, and C.D. Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[55] Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015.

[56] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[57] Quoc Truong Do, Tomoki Toda, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura. Preserving word-level emphasis in speech-to-speech translation. *IEEE*, 25(3):544–556, March 2017.

[58] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[59] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of ICML*, pages 1096–1103, 2008.

[60] J. Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of LREC*, 2012.

[61] Diego Marcheggiani and Ivan Titov. Encoding sentences with graph convolutional networks for semantic role labeling. *CoRR*, abs/1703.04826, 2017.

[62] D. Q. Truong, S. Sakti, G. Neubig, T. Toda, and S. Nakamura. Collection and analysis of a Japanese-English emphasized speech corpus. In *Proceedings of Oriental COCOSDA*, September 2014.

[63] G. Kikui, E. Sumita, T. Takezawa, and S. Yamamoto. Creating corpora for speech-to-speech translation. In *Proceedings of EUROSPEECH*, pages 381–384, 2003.

[64] P. Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, pages 388–395, 2004.

[65] Quoc Truong Do, Sakriani Sakti, Graham Neubig, and Satoshi Nakamura. Transferring emphasis in speech translation using hard-attentional neural network models. In *Proceedings of Interspeech*, September 2016.

[66] Yujie Su. Corpus-based comparative study of intensifiers: quite, pretty, rather and fairly. *Journal of World Languages*, 3(3):224–236, 2016.

[67] Erin Bennett and Noah D Goodman. Extremely costly intensifiers are stronger than quite costly ones. In *CogSci*, 2015.

[68] Carlos Busso and Shrikanth S. Narayanan. The expression and perception of emotions: comparing assessments of self versus others. In *INTERSPEECH*, 2008.

[69] Tim Polzehl, Sebastian Moller, and Florian Metze. Automatically assessing personality from speech. In *Semantic Computing (ICSC)*, pages 134–140. IEEE, 2010.

[70] John M Kirk. *Corpora Galore: Analyses and Techniques in Describing English: Papers from the Nineteenth International Conference on English Language Research on Computerised Corpora (ICAME 1998)*, volume 30. Rodopi, 2000.

[71] Ekkehard König and Volker Gast. *Reflexives and Intensifiers: The use of Self-forms in English*. 2002.

[72] Sylvia Chalker. *The Oxford Dictionary of English Grammar: 1000 Entries*. Oxford University Press, 2003.

[73] Rika Ito and Sali Tagliamonte. Well weird, right dodgy, very strange, really cool: Layering and recycling in english intensifiers. *Language in society*, 32(2):257–279, 2003.

[74] Angeliki Athanasiadou. On the subjectivity of intensifiers. *Language Sciences*, 29(4):554 – 565, 2007.

[75] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of ACL*, pages 55–60, 2014.

[76] Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. Learning to translate in real-time with neural machine translation. In *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Valencia, Spain, April 2017.

List of Publications

Journals

- **Quoc Truong Do**, Sakriani Sakti, Satoshi Nakamura. Sequence-to-Sequence Models for Emphasis Speech Translation. IEEE Transactions on Audio, Speech and Language Processing 2018.
- **Quoc Truong Do**, Tomoki Toda, Graham Neubig, Sakriani Sakti, Satoshi Nakamura. Preserving Word-level Emphasis in Speech-to-speech Translation. IEEE Transactions on Audio, Speech and Language Processing 2017.

International Conferences (Peer-reviewed)

- **Quoc Truong Do**, Sakriani Sakti, Satoshi Nakamura. Toward Expressive Speech Translation: A Unified Sequence-to-Sequence LSTMs Approach for Translating Words and Emphasis. Interspeech 2017
- **Quoc Truong Do**, Sakriani Sakti, Graham Neubig, Satoshi Nakamura. Transferring Emphasis in Speech Translation Using Hard-Attentional Neural Network Models. Interspeech 2016
- **Quoc Truong Do**, Tomoki Toda, Graham Neubig, Sakriani Sakti, Satoshi Nakamura. A Hybrid System for Continuous Word-level Emphasis Modeling Based on HMM State Clustering and Adaptive Training. Interspeech 2016.
- **Quoc Truong Do**, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. Improving Translation of Emphasis with Pause prediction in Speech-to-speech Translation Systems. IWSLT 2015.

- **Quoc Truong Do**, Michael Heck, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. The NAIST ASR System for the 2015 Multi-Genre Broadcast Challenge: On Combination of Deep Learning Systems Using a Rank-Score Function. ASRU 2015.
- **Quoc Truong Do**, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. Collection and Analysis of a Japanese-English Emphasized Speech Corpus. OCOCOSDA 2014.
- **Quoc Truong Do**, Satoshi Nakamura, Marc Delcroix, Takaaki Hori. WFST-Based Structural Classification Integrating DNN Acoustic Features and RNN Language Features for Speech Recognition. ICASSP 2015.
- **Quoc Truong Do**, Shinnosuke Takamichi, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. Preserving Word-level Emphasis in Speech-to-speech Translation using Linear Regression HSMMs. Interspeech 2015.
- Sashi Novitasari, **Quoc Truong Do**, Sakriani Sakti, Dessi Lestari, Satoshi Nakamura. Construction of English-French Multimodal Affective Conversational Corpus from Drama TV Series. LREC 2018.
- Sashi Novitasari, **Quoc Truong Do**, Sakriani Sakti, Dessi Lestari, and Satoshi Nakamura. Multi-modal Multi-task Deep Learning for Speaker and Emotion Recognition of TV-series Data. OCOCOSDA 2018.
- Sahoko Nakayama, Takatomo Kano, **Quoc Truong Do**, Sakriani Sakti, Satoshi Nakamura. Japanese-English Code-Switching Speech Data Construction. OCOCOSDA 2018.
- Michael Heck, **Quoc Truong Do**, Sakriani Sakti, Graham Neubig and Satoshi Nakamura. The NAIST English Speech Recognition System for IWSLT 2015. IWSLT 2015.

Domestic meeting

- **Quoc Truong Do**, Sakriani Sakti, Satoshi Nakamura. Joint Translation of Words and Emphasis in Speech-to-Speech Translation using Sequence-

to-sequence Models. ASJ 2017.

- **Quoc Truong Do**, Shinnosuke Takamichi, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. Word-level Emphasis Transfer in Speech-to-speech Translation. ASJ 2016.

- **Quoc Truong Do**, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura. Analysis of Word-level Emphasis Across English-Japanese. SIG-SLP 2015.