Speech Emotion Recognition in Real Environments using Characteristics of Emotional Expression and Perception

Atsushi Ando

Contents

Abstract

Speech is one of the most basic and important forms of human communication. It consists of three components: linguistic, non-linguistic (those cannot be controlled consciously such as gender, age, and emotions), and para-linguistic information (those can be controlled consciously like intentions and attitudes). In order to realize speech communication between humans and machines, a great deal of research has been conducted on linguistic information recognition, i.e. automatic speech recognition. More recently, research on the recognition of non-/para-linguistic information has attracted much attention to achieve more natural communication that understands speech as well as humans do. This thesis focuses on the recognition of speaker's emotion, one of the important factors of non-linguistic information.

Emotion plays an important role in speech communication. All the speaking behaviors such as linguistic contents and attitudes are influenced by emotions. Therefore, speech emotion recognition is essential for understanding speech communications. There are a lot of practical applications such as supporting agents or "voice of the customer" analysis, human-like spoken dialogue systems that empathize with the speaker's emotions, and human psychological state detection like driver's irritability.

Although many emotion recognition studies have been conducted, there are two difficulties in real environments. First, the expression of emotion is extremely complex and diverse. Speaker's emotion is expressed by any or a combination of prosodic, linguistic, and dialogic features. For example, negative feelings can appear in a low tone of voice, negative words, long pauses, and little backchannels. It is very difficult to capture all of these characteristics to recognize emotions. Second, emotions are subjective information that is strongly influenced by the perceiver (listener). The criterion of emotion perception may differ from listener to listener. For example, some listeners perceive the speaker to be happy about an utterance, while others perceive the speaker to be in a normal state. However, the conventional emotion recognition studies ignore this listener dependency and just estimate the majority-voted emotion of multiple listeners, which results in mismatching between outputs of automatic emotion recognition system and user's feelings in real applications.

To achieve highly accurate emotion recognition in real environments, this thesis performs two-step researches. The first step is the detection of particular emotions in a real but limited sound environment. The constraints of the target emotions and environments mitigate the diversity of emotional cues, the first problem, which brings the recognition to a practical level of accuracy. Furthermore, these constraints decrease the differences in the emotion perceptions between listeners. The second step is the recognition of the wider range of emotions in a diverse real-world environment. This step aims to solve the second problem since the differences in perceived emotions among listeners will be larger.

The task of the first step is customer satisfaction estimation in contact center calls. It can be applied to automatic agent evaluations. Two levels of customer satisfactions, turn-level and call-level, are estimated in this task. The main problem is that it is difficult to capture complex emotion expressions that appear in prosodic, linguistic, and dialogic cues. To solve this problem, a novel customer satisfaction estimation framework named a hierarchical multi-task learning model is proposed. The key idea of the proposed method is to leverage two characteristics of a customer's emotional expression. First, the satisfaction degrees of customers depend on the context. For example, *dissatisfied* emotional states tends to continue several turns while *satisfied* are not. Second, call-level and turn-level satisfaction results are closely related to each other. Calls in which the call-level satisfaction is *satisfied* tend to show *satisfied* turns in the middle and end of the call. The proposed model learns these characteristics of emotion expressions to employ Recurrent Neural Networks (RNNs) and multi-task learning of two-level estimation tasks. Experimental results on two datasets, simulated and real calls, show that the proposed method significantly improves the estimation accuracy of both call-/turn-level customer satisfaction estimations compared to the conventional method.

The second step tackles four-class basic emotion classification in natural speech. One of the applications is emotion-aware dialogue control in spoken dialog systems. The problem is though emotion perception varies with the listener in natural speech, most of the conventional methods ignore this individuality and just model the majority decision of multiple listeners. This thesis presents a new emotion recognition framework that models the emotion perception of individual listeners. The proposed method named as a listener-dependent model can estimate not only the perceived emotion of each listener but also the majority decision. It is inspired by the domain adaptation in deep learning, which has achieved great success in speech processing. Emotion classification experiments on two datasets demonstrate that the proposed method significantly improves the accuracy of listener-dependent emotion recognition.

These two studies demonstrate that there are certain trends in the expression and perception of emotional information, and that emotion recognition performance in real environments can be improved to utilize these trends. This thesis contributes to the advancement of the use of emotion recognition in real environments and the realization of natural communication between humans and machines.

1 Introduction

1.1 Background

Human speech is the most basic and widely-used form of daily communication. Speech conveys not only linguistic information but also other factors such as speaker, emotion, and so on, all of which are essential for understanding speech communication. There have been many studies on the recognition of linguistic and some non-linguistic information, especially speaker identity. In recent years, the recognition of the rest of non-/para-linguistic information has also attracted attention. This thesis focuses on speech emotion recognition (SER), which is one of the most important aspects of non-linguistic information recognition.

There are a lot of SER applications. One is assisting agents in contact centers. Identifying the customer's positive/negative reactions yields a better understanding of the strengths and weaknesses of products and services. Online monitoring of CS will enable supervisors to take over the calls as soon as customers start to exhibit negative responses [1]. Another is a visualization of mental state. Driver state monitoring from speech prevents risky driving [2]. It is also applicable in mental health assessment [3]. The other is an advancement of spoken dialog systems. It allows yielding human-like responses such as sympathy, which gains rapport between humans and systems [4].

SER can be categorized into two tasks: dimensional and categorical emotion recognition. Dimensional emotion recognition is the task of estimating the values of several emotion attributes present in speech [5]. Three primitive emotion attributes, i.e. valence, arousal, and dominance, are commonly used [6]. Categorical emotion recognition is the task of identifying the speaker's emotion from among a discrete set of emotion categories [7]. The ground truth is defined as the majority of perceived emotion class as determined by multiple listeners. Comparing these two tasks, categorical emotion recognition is more suitable for most applications because it is easy to interpret.

1.2 Thesis Scope

This thesis aims to improve categorical SER performance in real environments. In all subsequent studies, the inputs are audio signals alone and the objectives are finite numbers of emotion categories. Note that the ground truth emotions in this thesis are those which perceived by third party listeners, not experienced emotions of speakers themselves, as following to the most SER studies [8–10].

A large number of SER methods have been proposed. One of the basic approaches is based on machine learning, shown in Figure 1.1. The main components are two: the feature extractor and the emotion recognition model. The audio feature is obtained from an audio signal with the feature extractor, then the posterior is evaluated by the emotion recognition model. Several types of features such as acoustic [8], lexical [11], and dialogic features [12] are employed as the features. Statistical models such as Deep Neural Networks (DNNs) and Support Vector Machines (SVMs) are used as the emotion recognition model. Most of the recent models are composed of multiple types of DNN layers like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and attention layers to learn time-varied emotional cues automatically. End-to-end emotion recognition models which have no feature extractors have also been proposed so as to leverage the rich information of raw audios [7, 13]. These

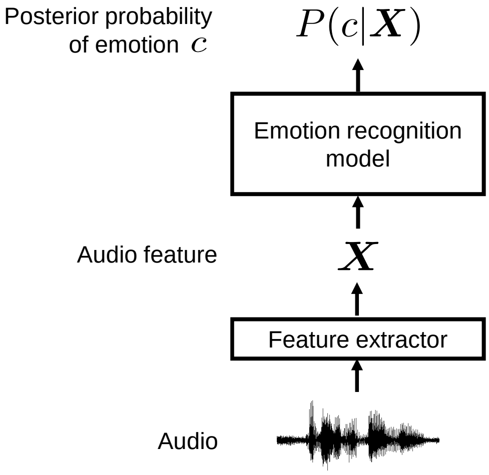


Figure 1.1: *Speech emotion recognition based on machine learning.*

models are trained with a large amount of training data, i.e., pairs of audio and the ground-truth emotion labels.

Despite these advances, SER is still a challenging task. There are two main difficulties; diversity of emotion expression and listener dependency of emotion perception. First, it has been reported that emotional cues appear in a great variety of ways. For example, negative feelings can appear in a low pitch of voice, negative words, long pauses, little backchannels, and so on. [14,15]. Furthermore, the expression of emotions depends on the speaker's situation; even particular product names are the indicators of customer's negative feelings in contact center calls [1]. It is difficult to learn these complex cues to recognize emotions accurately. Second, emotion perception depends on listeners. When listening to a certain speech, some listeners feel that the speaker is happy, while others feel he/she is in a neutral state. It should be considered for emotion recognition because it is necessary to give convincing recognition results for various audiences in practical applications.

In this thesis, a step-wise strategy is taken to realize SER in real environments. The steps are illustrated in Figure 1.2. The first step is to recognize particular target emotions in real but limited environments. These limitations mitigate the problem of the diversity of emotion expression, which provides high recognition accuracy. Customer satisfaction estimation in contact center calls is selected as the task of the first step. Ground truths are determined by multiple well-educated real supervisors, which leads to ignoring the second problem of the perceptual differences between listeners. The second step is to estimate a wider range of target emotions in various situations. The task is a basic emotion classification for isolated speech in human-to-human communication. The ground truths are defined by a wide variety of listeners, and the main issue of this step is to handle listener dependency on emotion perception. Our ultimate goal, recognition of every emotion in any situation, is very difficult with current technology that it is out of scope in this thesis.

1.3 Thesis Overview

Customer satisfaction estimation in contact center calls is reported as the first task. There are two sub-tasks: call-level and turn-level, each of a series of customer's utterances, satisfaction estimation. The main problem is to learn customer satisfaction degrees that appear in various aspects of features. The key idea of the proposed approach is to utilize the characteristics of customer's emotional expression patterns. Analyses of the contact center calls indicate that there are typical emotional expression patterns in positive and negative calls, e.g. customers tend to represent their satisfactions in the middle and the end of positive calls. The proposed method, named as a Hierarchical Multi-Task (HMT) model, learns these characteristics by simultaneous training of the two-level estimation tasks. Three types of heuristic turn-level features, prosodic, lexical, and dialog features, are employed to capture emotional cues in indi-

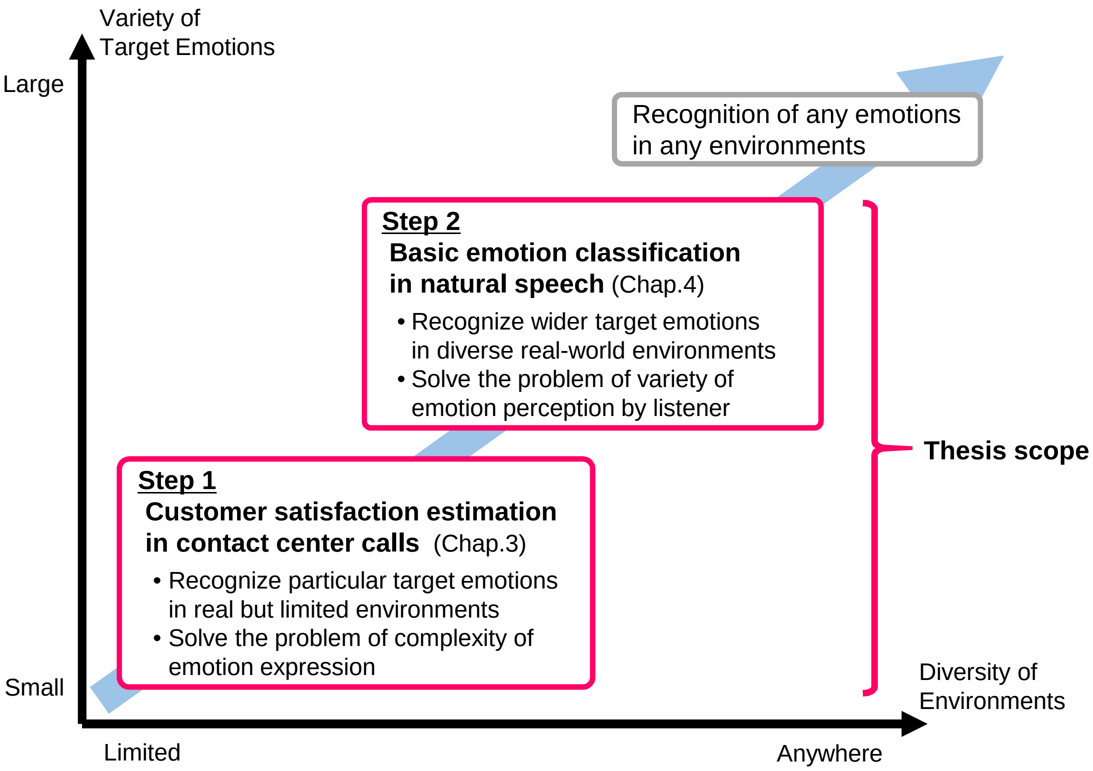

Variety of
Target Emotions
Large
Small
Limited
Anywhere
Diversity of
Environments
Recognition of any emotions
in any environments
Step 2
Basic emotion classification
in natural speech (Chap.4)
• Recognize wider target emotions
in diverse real-world environments
• Solve the problem of variety of
emotion perception by listener
Step 1
Customer satisfaction estimation
in contact center calls (Chap.3)
• Recognize particular target emotions
in real but limited environments
• Solve the problem of complexity of
emotion expression
Thesis scope

Figure 1.2: *The scope of this thesis.*

vidual turns. Furthermore, an adaptation framework for the HMT model is presented in order to mitigate the domain dependency problem. Experiments on simulated and real calls reveal that the proposed method outperforms the conventional frameworks.

The second task is basic emotion classification in natural speech. The challenge is to solve listener dependencies of emotion perceptions. To solve this problem, a novel SER framework that constructs listener-dependent emotion perception models is proposed. The model can estimate the listener-specific perceived emotion from an audio signal and a target listener indicator. It can also evaluate the dominant emotion by averaging the outputs of multiple listener-dependent models. Three types of DNN adaptation frameworks that have been successful in speech processing are employed for the listener-dependent model. Experiments on two emotional speech corpora show the individuality of listener perception and the effectiveness of the proposed approach.

This thesis is organized as follows. The related work has been reviewed in Chapter 2. The first task, customer satisfaction estimation, is described in Chapter 3. The study of basic emotion classification is presented in Chapter 4. Finally, the summary and future work are in Chapter 5.

2 Related work

2.1 Introduction

This chapter presents related work on speech emotion recognition. First, to overview the general information of speech emotion recognition, the theory of emotion, the position of emotion in phonetics, and the task descriptions of speech emotion recognition are discussed. Next, conventional techniques for the two speech emotion recognition tasks that are tackled in this thesis are described; customer satisfaction estimation in contact center calls and basic emotion classification in natural speech.

2.2 Speech Emotion Recognition

2.2.1 Description of Emotion

There are two main theories to explain the construction of emotion. The first is basic emotion theory established by Ekman [16]. He theorized that several emotion categories are recognized universally in different cultures. Six basic emotions are reported in his work: anger, disgust, fear, happiness, sadness, and surprise. Plutchik has been developed the basic emotion theory to determine the wheel of emotions; there are eight primary emotions grouped on a positive and negative basis [17]. They also reported that complex emotions could be formed by combining basic emotions. The second is multi-

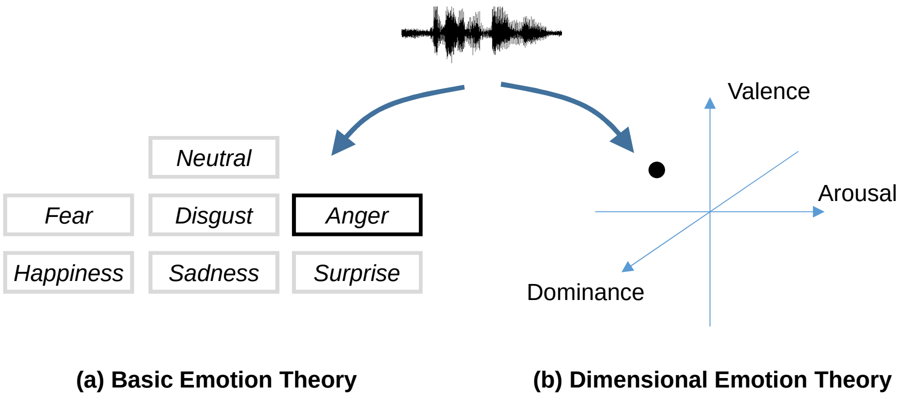


Figure 2.1: *An example of the two emotion theories: basic emotion [16, 17] and dimensional emotion [18, 19].*

dimensional factor theory [18]. It attempts that emotions underlie a few dimensions. Valence, Arousal, and Dominance are often used as the factors of dimensions [19]. One of the advantages of this theory is that it can capture the similarities between emotional experiences. The comparison of the two theories is shown in Figure 2.1.

2.2.2 Speech Emotion

In phonetics, speech is categorized as a factor of non-linguistic information. It has been reported that speech conveys three types of information: linguistic, non-linguistic, and para-linguistic [20, 21]. Linguistic information is the symbolic information that is represented by a set of discrete symbols, and it can be transcribed in written language. Non-linguistic, or sometimes referred to as extralinguistic [22], information is such factors as the age, gender and physical and emotional state of the speaker, and so on, and can not generally be controlled by the speaker. Para-linguistic is the factors that are controlled by the speaker to modify and supplement the linguistic information, e.g. intentions, attitudes, and speaking styles, etc.

2.2.3 Tasks of Speech Emotion Recognition

Following the emotion construction theories, there are two major tasks in SER. One is categorical emotion recognition [23, 24]. The task is classifying each utterance into a finite number of target emotions. A subset of basic emotions is often used as the target. Detection of particular emotions like frustration detection [25] is included in this task. The main advantage is that it is easy to interpret the recognition results, which is suitable for many applications. The other is dimensional emotion recognition. It follows dimensional factor theory to regress the value of each dimensional factor. Some studies tackled utterance-level estimation [26], while others were frame-level evaluation in an utterance to capture emotional changes by time [27]. The advantage of this task is to measure the similarity of estimated emotions. This thesis adopts categorical emotion recognition as the tasks with a view to practical applications.

Another important aspect of emotion lies in its representation and observation in speech communication. One framework is the Brunswik functional lens model [28] illustrated in Figure 2.2. It supposes at least two participants, the speaker and the listener, and provides three types of emotion representations. One is the experienced emotion that is the latent emotional state of the speaker. Another is the expressed emotion that represents in the speaker's behaviors including voice, face, body gestures, etc. The other is the perceived emotion that is decoded from the behaviors by the listener. These representations may sometimes be mismatched [29], and it is very difficult for the experienced and expressed emotions to determine the ground truths because no one can validate the correctness of them. Thus this thesis defines the perceived emotions as the objectives of the recognition, as with most conventional studies [8–10].

The ground truth of the perceived emotion is defined as the majority-voted emo-

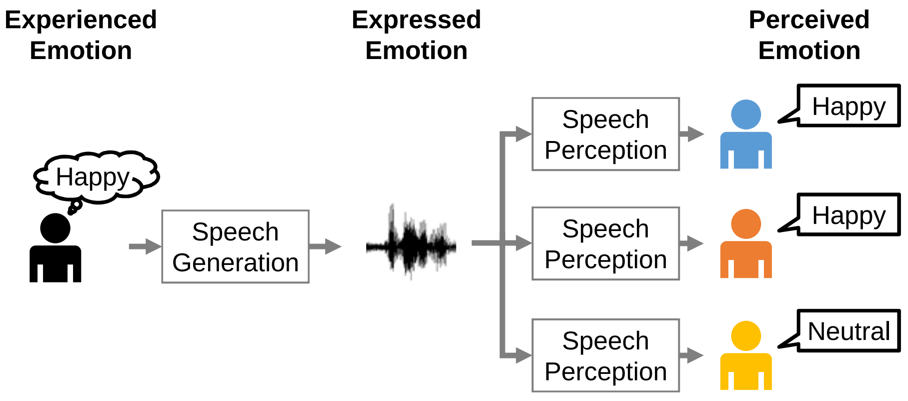


Figure 2.2: *An illustration of emotion representation in human communication by the Brunswik functional lens model [28].*

tional category of multiple listeners for categorical emotion recognition. SER methods are evaluated on the basis of the accuracy of the majority-voted emotion estimation. Most conventional methods use the majority-voted emotions directly for training, while several studies employ listener-wise perceived emotions to model diversity of emotion perceptions. The latter approaches are described in Section 2.3.2.

2.3 Conventional Studies for Speech Emotion Recognition

2.3.1 Customer Satisfaction Estimation

Customer satisfaction (CS) estimation is one of the major fields in SER. There are two tasks: call-level and turn-level CS estimation. The call-level and turn-level CS mean customer satisfaction degrees in the entire call and each turn in the call, respectively. An example of the CS ground-truth in two tasks is shown in Figure 2.3.

Table 2.1: *Studies of call-level/turn-level customer satisfaction estimation.*

Task	Author	# class	Feature	Classifier	Label	# of train	year
Call-level	Gamon [30]	4	Lexical	SVM	self	40884 docs	2004
	Gupta+ [31]	3	Acoustic/Lexical	GMM	(no info.)	20 calls	2007
	Godbole+ [32]	6	Lexical	NB/SVM/rule	annot.	3000+ docs	2008
	Park+ [1]	2/5	Acoustic/Lexical/Contexual	DT/LR/NB/SVM	self	115 calls	2009
	Hassan+ [33]	2	recognized emotions	DT/NB/k*/NN	self	39 calls	2010
	Llimona+ [34]	3	Gender/Duration	(Analyses)	self	17309 calls	2015
	Chakraborty+ [35]	4	Acoustic/Dialog event	rule	annot.	73 calls	2015
	Chowdhury+ [12]	3	Acoustic/Lexical/Interactive	SVM	annot.	739 calls	2016
	Sun+ [36]	2	Acoustic/Lexical	SVM	self	8652 calls	2016
	Segura+ [37]	2	Raw waveform	CNN	self	19360 calls	2016
	Cong+ [38]	2	Acoustic/Word seq.	NN/CNN	self	31120 calls	2016
	Bockhorst+ [39]	Regress	Lexical/Telephone-Log	Rank model	self	8726 calls	2017
	Luque+ [40]	2	Acoustic/Word seq.	CNN	self	17612 calls	2017
Turn-level	Devillers+ [41]	4	Acoustic/Lexical	MLE	annot.	4548 turns	2003
	Vidrascu+ [42]	5	Acoustic/linguistic event	SVM	annot.	2294 turns	2007
	Morrison+ [43]	2	Acoustic	SVM/RF/NN/K*	annot.	388 turns	2007
	Devillers+ [44]	3	Acoustic	SVM	annot.	7452 turns	2010
	Polzehl+ [45]	2	Acoustic/Lexical	SVM/NN	annot.	16802 turns	2011
	Nomoto+ [46]	2	Acoustic/Lexical/Interactive	SVM	annot.	800 turns	2011
	Erden+ [47]	2	Acoustic/Lexical	SVM/GMM/NN	annot.	8512 turns	2011
	Vaudable+ [48]	3	Acoustic/Lexical	SVM	annot.	3684 turns	2012
	Galanis+ [49]	2	Acoustic/Speaker info.	SVM	annot.	1396 turns	2013
	Chakraborty+ [50]	2	Lexical/recognized emotions	SVM/NN/k-NN	annot.	354 turns	2016
	Seng+ [51]	6	Acoustic/Visual	NN	annot.	1679 turns	2018

In most cases, a two-channel call signal alone is available in both tasks. Not only prosodic but also lexical and interactive features are utilized because the vocabulary used in contact center calls is biased and turn-taking characteristics will be related to the customer's emotion. The conventional studies are outlined in Table 2.1.

Call-level Estimation

The approaches of conventional call-level estimation can be categorized into two groups; extracting call-level features and integrating short-term estimated emotions.

The first group estimates call-level CS by call-level features and a classifier. Many heuristic call-level features have been investigated. Acoustic features such as statistics of pitch, intensity, duration, and Mel-Frequency Cepstral Coefficients (MFCCs) in customer utterances are widely used [1, 12, 31, 35, 36]. Lexical features such as word

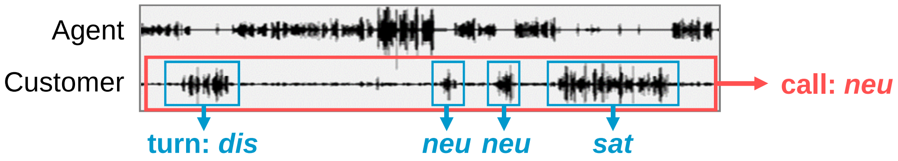


Figure 2.3: *An example of the call-level and the turn-level customer satisfaction. The descriptions of dis, neu, sat means dissatisfied, neutral, and satisfied, respectively.*

N-gram or Bag-of-Words are also employed to capture emotion-related idioms such as appreciation or criticism [1,12,30–32,36,39]. Note that most previous studies draw their linguistic features from automatic speech recognition results to extract the features without transcription. Some conventional methods employ interactive features like turn overlap or call dominance, which reflects customer's feelings [39]. Dialog event features such as answer repetition were proposed for interactive voice response systems [35]. Meta information of calls like a history of previous interactions and in-queue waiting time has been employed [1,39]. It has also been reported that the gender combination of customer and agent is associated with call-level CS [34]. Simple multi-class classifiers such as Support Vector Machine (SVM) were used in these works. In contrast with these heuristic feature-based methods, recent studies employ Deep Neural Networks (DNNs) to acquire call-level features automatically. Low-level features such as audio signals or automatic speech recognition results are used as input to the classifiers [37,38,40]. These conventional methods mainly focus on the global characteristics of calls, but some real calls are too complex for these frameworks to estimate call-level CS accurately. For example, some of the real contact center calls contain both positive and negative customer attitudes; customer dissatisfaction in the first half, and better feelings in the second half.

The other group integrates short-term emotion recognition results for call-level estimation [33]. A pre-trained emotion classifier is applied to each customer utterance to get posterior probabilities of the customer's short-term emotion. The results are integrated by heuristic rules (e.g. overall average or last-K emotions) to estimate call-level CS. One of the advantages of this framework is that it is possible to consider the localized characteristics of the customer in a call. However, there are several problems in this method; short-term emotion classifiers are usually trained by domain-mismatched data such as acted emotional speech, and the heuristic integration rules proved to be too rough to capture long-range sequential changes in customer emotions. Our proposal in this thesis employs this approach while using LSTM-RNNs to model long-range sequential information.

Turn-level Estimation

Conventional work on turn-level CS estimation is similar to that of call-level estimation. Most solutions employ turn-level features with a simple classifier. Acoustic and lexical features are also commonly used in the turn-level task [41–51]. Several types of turn-level statistics of acoustic features have been developed [43, 44]. Lexical features based on bag-of-words or CS-wise language models have also been widely used [47, 48]. Interactive features similar to the call-level features are effective [46]. It is reported that speaker information defined by call meta-information, such as gender, is also effective [49]. Some studies proposed the use of linguistic event features found in transcriptions like laughing [41]. In recent years, the use of visual features has been proposed for video-based calls [51]. One conventional method similar to the call-level method utilizes emotion recognition results [50]. These methods are applicable to online CS estimation, i.e. estimation of current CS degree during a call using information from the call beginning to the current time.

One of the problems with conventional methods is that long-range sequential information is ignored. While conventional methods assume that CS class in individual turns is independent, customer's feeling in a turn is strongly related to the surrounding agent/customer turns. It is desirable to consider long-range sequential information of turns in a call. Another problem is that call-level CS is seldom considered in the turn-level estimation. The use of call-level CS will improve turn-level estimation performance because the distributions of turn-level CS are related to call-level CS. Note that the proposed method described in Chapter 3 of this thesis achieves both points. The proposal, the HMT model, uses LSTM-RNNs to learn long-range sequential changes in turn-level CS; call-level CS is taken into consideration in the training step to acquire the relationship.

2.3.2 Basic Emotion Classification

This section describes the conventional studies in basic emotion classification. It is the task that classifies an isolated speech into a finite number of target emotions. The target emotions are usually a subset of basic emotions proposed by Ekman [16]. The ground truth is defined as the majority-voted emotion of multiple listeners.

There are two major approaches to basic emotion recognition. The first is those based on acoustic information alone, while the second is based on the multimodal features, i.e. combination of acoustic with other features.

Approach 1: Acoustic-based Method

It is reported that human perception of emotion is affected by prosodic characteristics of speech. For example, angry speech tends to be high pitch variance and fast speaking rate [15]. Thus this approach utilizes acoustic features that quantify prosodic information.

The traditional approaches are based on utterance-level heuristic features including the statistics of frame-level acoustic features such as pitch, power, and MFCC [8, 52]. However, it is difficult to create truly effective features because emotional cues exhibit great diversity. Most of these recent studies are based on DNN-based models. The posterior probabilities of the majority of the perceived emotions are estimated from each acoustic feature sequence. The estimation model consists of multiple DNN layers such as CNNs, Long Short-Term Memory (LSTM)-RNNs, attention mechanisms, and Fully-Connected (FC) layers [7, 9, 13]. Frame-level features like log power spectrum [23, 24], log-mel filterbank [53] or heuristic Low-Level Descriptors (LLDs) such as fundamental frequency f_o and zero-cross ratio [10, 54] are often used as the input. One of the advantages of this approach is that the model can automatically learn context-sensitive emotional cues. Our proposals are based on these successful DNN-based frameworks.

Approach 2: Multimodal-based Method

Humans express their emotion by not only prosody but also other aspects such as the linguistic contents, face, and dialog behaviors. Thus some of the conventional studies combine acoustic features with lexical, visual, and dialogic ones.

One of the multimodal-based approaches leverages linguistic features. Word features, e.g. bag-of-words and Term Frequency-Inverse Document Frequency (TF-IDF) [55], and word sequences themselves are used as a kind of the audio features in Figure 1.1 [11, 56]. In the extraction of linguistic features, automatic speech recognition or transcription is used [57]. Since many applications do not have transcriptions, it is more practical to use speech recognition results. However, the problem is that automatic recognition results may contain word errors, which degrade emotion recognition performances.

Another approach is to use visual features. Several methods have been proposed to estimate emotions using a DNN model with both audio and face images [58]. In general, emotion recognition based on face images tends to be more accurate than emotion recognition based on speech [59], so the combination with video information is expected to improve recognition accuracy. However, the use of visual features may restrict the application, and it may also cause recognition errors due to the bad position of the face and occlusion.

The other is to utilize dialogic features. It has been reported that the interlocutor's information is effective for the target speaker's emotion recognition in a two-speaker conversation [60, 61]. The use of dialogic features will be useful in such a limited environment.

Speaker and Listener Dependencies in Emotion Recognition

In order to improve the above two approaches, there are several methods that consider speaker and listener dependencies in SER.

Several studies have investigated speaker dependency on emotional expression. They indicate that the speaker differences significantly affect emotion representation. For example, each speaker exhibits different laryngealisation and pitch characteristics [62]. It has been suggested that speaker variability is a more serious factor than linguistic content [63]. Therefore, a lot of speaker adaptation methods for SER have been developed. Some attempt feature-level normalization; a speaker-dependent utterance feature is transformed into its speaker-independent equivalent [64]. Another approach is model-level adaptation. A speaker-independent emotion recognition model can, with a small amount of adaptation data, be adjusted to yield a speaker-dependent model [65].

Recent studies employ multi-task learning to construct gender-dependent models without inputting speaker attributes [53, 66]. Personal profiles have also been utilized to estimate speaker-dependent emotion recognition [67].

It has also been reported that emotion perception varies with the listener. Younger listeners tend to perceive emotions more precisely than their elders [68]. It is reported that female listeners are more sensitive to emotion than males [69]. The perception also depends on culture [70]. Even though listener variability affects the majority decision as to emotion perception, there is little work that considers the listener in SER. One related work is soft-label / multi-label emotion recognition; it models the distribution of emotion perception of listeners [24, 54, 71]. However, it is impossible with their approaches to model the bias of emotion perception for a particular listener since it cannot distinguish individuals. In music emotion recognition, several studies have tackled listener-wise perception [72, 73]. However, to the best of our knowledge, there is little work to utilize listener variability for SER. We thus aim to make listener-dependent emotion recognition models in this thesis.

It is considered that constructing listener-oriented emotion recognition models is strongly related to the frameworks created for domain or speaker adaptation. As mentioned with regard to speaker adaptation in emotion recognition, adaptation has two approaches: feature-based and model-based adaptation. In recent speech processing methods such as those for speech recognition and speech synthesis, model-based adaptation is dominant because it is very powerful in handling complex changes in domains or listeners. One of the common adaptation approaches is updating the parameters of a pre-trained model by using a target domain dataset [74]. Another approach is developing a recognition model that includes the domain-dependent part. Technologies along these lines such as switching domain-dependent layers [75], projections with auxiliary input [76, 77] or summation of multiple projection outputs with speaker-dependent weights [78] have been proposed. Inspired by these successful frameworks, our proposal yields listener-dependent emotion recognition models.

2.4 Summary

This chapter described related studies on speech emotion recognition. The general information about speech emotion recognition was summarized; the description of emotion, the fundamental tasks, and the difference from other information contained in the speech. Conventional methods for the two speech emotion recognition tasks, customer satisfaction estimation and basic emotion classification, are then introduced.

3 Customer Satisfaction Estimation in Contact Center Calls based on a Hierarchical Multi-Task Model

3.1 Introduction

Contact centers are one of the most important channels between companies and their customers. They are regarded as far more than merely replying to customer's requests but are also seen as great opportunities for increasing sales. Customer's discontents in calls will indicate new ideas for products [79, 80]. Kind and supportive responses by agents improve the brand by directly promoting loyalty [81]. Therefore, most companies see a pressing need to improve contact center performance.

Customer satisfaction (CS) is the dominant quality attribute in contact centers. The ratio of satisfied/dissatisfied calls can indicate the performance of individual agents and contact centers. Identifying the customer's positive/negative reactions yields better understanding of the strengths and weaknesses of products and services. Furthermore, online monitoring of CS will enable supervisors to take over the calls as soon as customers start to exhibit negative responses [1]. Therefore, measuring the CS of each call and interval is essential for improving the quality of products and contact center itself.

However, as most current contact centers survey CS scores manually by sampling and listening to calls, the survey size is limited due to the high costs. Thus automatic CS estimation is an urgent requirement.

The tasks of CS estimation can be categorized into two groups: call-level and turn-level estimations. Call-level estimation is assessing the CS degree of an entire call. Various kinds of call-level features have been developed such as prosodic change and spoken words of the customer [1,12,36,38], opening/closing talk of the agent [1] and turn-taking characteristics [12]. Meta information of calls like previous inbound interactions or waiting time have also been employed [1, 39]. Recent approaches utilize deep neural networks to capture call-level features automatically from raw waveforms [37] or automatic speech recognition results [40]. Turn-level estimation, on the other hand, is to evaluate the CS in each customer turn during a call. A turn represents a segment as determined by speaker change information. Most conventional methods use hand-crafted prosodic and lexical features extracted from individual customer turns [41, 43, 48].

Though conventional methods improve estimation performances by means of developing features, there are two remaining problems. First, in both tasks, long-range sequential information is ignored. In turn-level estimation, conventional methods treat individual turns as being independent even though the CS in one turn is related to the surrounding agent/customer turns. Conventional methods for call-level estimation use statistics of hand-crafted features as determined for the entire call, not the sequences. Second, the relationship between turn-level and call-level CS is not considered. In fact, call-level CS tends to be positive if turn-level CS rises towards the end of the call. However, all conventional methods regard the two estimation tasks as being separate, and so model them independently.

In this chapter, we propose a new CS estimation method that utilizes long-range sequential information and the relationship between call-level and turn-level CS. Different from conventional methods, we assume that both call-level and turn-level CS labels are available. The proposed method, named a Hierarchical Multi-Task (HMT) model, is achieved by hierarchically stacking two-level CS estimation networks. Long short-term memory recurrent neural networks (LSTM-RNNs) are employed for both turn-level and call-level CS estimation to capture the long-range sequential contexts of calls. Both networks are hierarchically stacked so that turn-level estimation results can be used directly for call-level estimation. Furthermore, the proposed method uses joint optimization training to learn the relationship of the two tasks. Hand-crafted features derived from several conventional studies are used as input. Our HMT model is inspired by a Joint Many-Task model, which uses the estimated result of one task as the input of another task [82].

Our previous work [83] conducted preliminary experiments on acted calls to elucidate the effectiveness of the HMT model for both turn-level and call-level estimation. However, several issues remained outstanding; real calls were not used in the performance evaluations, and the features were not analyzed. Thus this thesis conducts further investigations to better elucidate the capability of our HMT model. The contributions of this study are as follows:

- Real contact center calls are used in addition to acted ones to evaluate the HMT model.

- Data analyses of turn-level and call-level CS are conducted to reveal their characteristics and relationship. To the best of our knowledge, this is the first work to investigate them.

- Feature comparisons are presented to measure the effectiveness of the hand-crafted turn-level features.

- All the ground truths are defined by harmonizing multiple third-person annotations. Furthermore, the effects of harmonization are investigated.

- In addition to the flat-start training of the HMT model, an adaptation technique is presented. Our adaptation method can improve CS estimation performance of not only the call-level task, but also the turn-task one even if only call-level labels are available.

This chapter is organized as follows. The task description of this chapter is shown in Section 3.2. Details of the datasets and analyses conducted are introduced in Section 3.3. The proposal, the HMT model, and its training methods are shown in Section 3.4. Evaluation experiments are reported in Section 3.5 and the conclusion is given in Section 3.6.

3.2 Task Descriptions

The tasks of this chapter are two: call-level and turn-level CS estimation. We treat both as a three-class classification task like several conventional studies [12,44,48]. The classes are *pos*, *neu* and *neg*. *pos* and *neg* represent the articulate positive/negative statements of a customer. *pos* includes emotions such as satisfied, happy and excited, while *neg* includes emotions like dissatisfied, disappointed, frustrated and angry. *neu* is neither of them, i.e., inarticulate positive/negative statements and neutral emotion. Such three class classification is desired for automatic quality measurement of products or agent's performance; articulate positive/negative calls probably derive from the quality of products or agent's performance, while inarticulate ones may be derived from the customer's characteristics. However, our definition means that *neu* class includes various types of emotion samples, which will complicate the solution of classification problems.

All turns and backchannels are detected automatically in a three step process. First, automatic voice activity detection based on switching Kalman filter [84] is applied to both agent and customer channels to obtain Inter-Pausal Units (IPUs) [85]. IPUs are defined as consecutive tokens with no gap greater than 200 ms. Second, IPUs less than 1 second or those that include contradictory IPU intervals are regarded as backchannels. Finally, a continuous string of IPUs without backchannels is taken as a turn. An example of detected turns and backchannels is shown in Figure 3.1. Turn-level estimation is the task of estimating CS class of each individual customer turn from itself and its surrounding turns and backchannels.

Different from several conventional studies, we define ground-truth CS classes from multiple annotators. It is reported that self-reported CS rating has several issues [39, 86]; human ratings of emotion do not follow an absolute scale, and customers may not provide enough objective evidence about subjective opinions. Thus we employ third-person labels to make reasonable ground-truths for most professional agents. In this chapter, all the annotators are contact center supervisors who are trained in the criteria of CS and evaluate call-level/turn-level CS themselves on a daily basis.

In turn-level estimation, we assume two situations: online and batch. Online demands evaluation from only the current and the past turns/backchannels of the agent and the customer, which is the same condition as the conventional studies. Batch is allowed to use all the turns and backchannels in a call to estimate each turn-level CS. Online estimation is more useful than batch situation because it can be used for wider applications such as real-time CS monitoring. However, it will be more difficult due to the limited amount of information of the agent and the customer available.

Furthermore, we set two training conditions: flat-start and adaptation. Flat-start

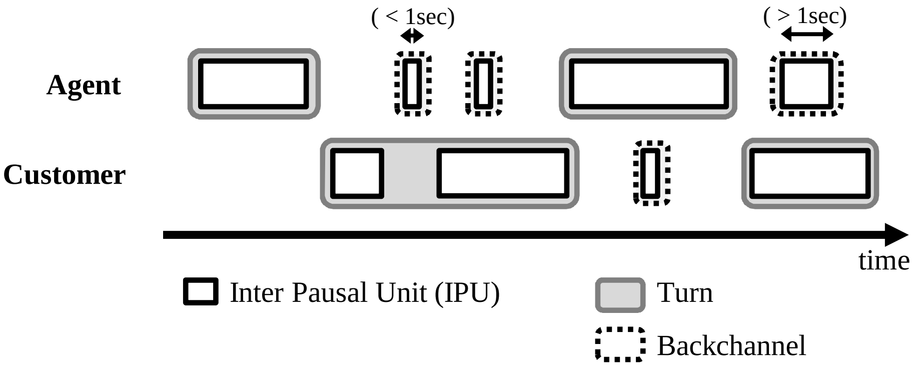


Figure 3.1: *An example of detected turns in a call.*

constructs estimation models from the beginning while adaptation adjusts the trained models to particular contact center calls. It is reported that there are some domain-dependent factors in CS estimation [44] and domain adaptation will be required in practice, as often mentioned in speech recognition or image classification studies. Furthermore, we consider two adaptation scenarios: adaptation with both level labels and that with just call-level labels. This is because the cost of turn-level label annotation is much higher than that for call-level annotation so adaptation with just call-level labels is desired in practice.

3.3 Datasets

This section describes the two Japanese call datasets, acted and real, used in this chapter. Furthermore, several analyses of turn-level/call-level CS that reveal their characteristics and relationship are discussed.

3.3.1 Recording and Annotation

We created an acted call dataset to collect a wide range of ‘realistic’ emotional calls. The calls were made by improvised conversations along with predetermined outlines. The domain was the customer center of a frozen food company and included several sub-tasks such as new orders, inquiries, and cancellations. First, we made various types of outlines, each consisting of a situation, desired results, persona of a customer, and change of emotion in each sub-task. Persona included information of gender, age, family structure and character. Emotional words like ‘gladly’ and ‘with annoyance’ were used to indicate customer’s attitudes. These outlines were strictly checked by real contact center agents and the calls that lacked naturalness (e.g. mismatch between customer’s character and emotion patterns) were eliminated. This process yielded 89 outlines. Then, a pair of speakers read the same outline and decided their roles as operator or customer before talking. They held an improvised conversation via phone sets while following the outline. All of the speakers were contact center agents (10 males, 19 females) and speaker pairs were selected randomly. Inadequate calls which include overacted utterances or those that deviated from the outline were rejected as agreed by the three inspectors. All of them were real contact center supervisors. The result was 306 recorded calls from 29 speakers that included a variety of satisfaction and dissatisfaction types. The total length was 29.7 hours; the calls had talk times ranging from 4 to 12 minutes. All were recorded in stereo, 8 kHz with 16 bit format.

The real calls came from a technical support center handling personal computers and network appliances. The tasks were mainly troubleshooting of IT devices. Agents and customers included both male and female, but no meta-data was available. We randomly picked approximately four hundred calls recorded in the period from August to September, 2017. Then the inadequate calls, e.g. those which were disconnected in the middle of the conversations, were eliminated with the agreement of multiple supervisors. Finally, we collected 391 valid calls. Total and average lengths of calls were 39.1 hours and 6.1 minutes, respectively. The recording format was the same as that of the acted dataset.

Both call-level and turn-level CS ground truths were decided by three annotators. All the annotators were contact center supervisors (same as the inspectors of inadequate calls) and every call was annotated by two of them. These two levels of ground truths were decided independently by harmonizing the call-level/turn-level annotations as described below. First, each annotator listened to each call twice; first time to assign CS degree to the entire call and the second time to give CS degrees to arbitrary intervals in the call. A 5-point Likert scale was used; *5:very positive, 4:positive, 3:neutral, 2:negative, 1:very negative.* Positive included satisfied, happy, excited while negative included dissatisfied, frustrated and cold/hot anger. To adjust the criteria among the annotators, all listened to 5 sample calls of each call-level CS degree before commencing the annotation. Sample calls were previously selected by one annotator from the acted calls, and they were not used thereafter. Second, the ground-truth decision step collapsed CS degrees to assign three CS classes: *pos* included *5:very positive* and *4:positive*, *neg* included *1:very negative* and *2:negative* and *neu* was *3:neutral.* With regard to turn-level CS classes, in addition to that criteria, if an annotator assigned *pos* or *neg* intervals with over 50 % overlaps of IPUs in a customer turn, we regarded that turn as *pos* or *neg*, respectively, while all the rest were *neu.* Ground truths were finally defined to harmonize annotator-wise CS classes. They were *pos* or *neg* if both annotators assigned *pos* or *neg* class to each call/turn, while the rest were *neu.* Because of these harmonization operations, *pos* and *neg* class well exhibited the characteristics of these classes while some of *neu* may shade into the *pos* or *neg* class. The numbers of calls or

Table 3.1: *CS label distributions in the acted and the real dataset.*

	# of calls			# of turns		
	neg	*neu*	*pos*	*neg*	*neu*	*pos*
acted	71	111	119	2162	4786	962
real	31	304	56	376	7603	227

Table 3.2: *Means and standard deviations of call/turn durations.* * *represents that its mean showed significant differences from those of the two other classes.*

	call-*neg*	call-*neu*	call-*pos*	turn-*neg*	turn-*neu*	turn-*pos*
acted	352.7 ± 50.4	371.8 ± 65.8	398.9 ± 81.8	6.4 ± 6.1	6.1 ± 6.5	5.6 ± 5.4
real	371.3 ± 127.9	332.6 ± 129.1	418.9 ± 256.4	10.4^* ± 11.7	5.5 ± 6.4	5.4 ± 4.5

turns in each dataset are shown in Table 3.1.

To measure pair-wise agreement of the two annotators, we evaluated Cohen's kappa. The average kappa coefficients of three pairs of annotators were 0.68 for call-level and 0.51 for turn-level in the acted data; 0.53 for call-level and 0.42 for turn-level in the real data. These values suggested moderate matches and that the ground truths were sufficiently reliable.

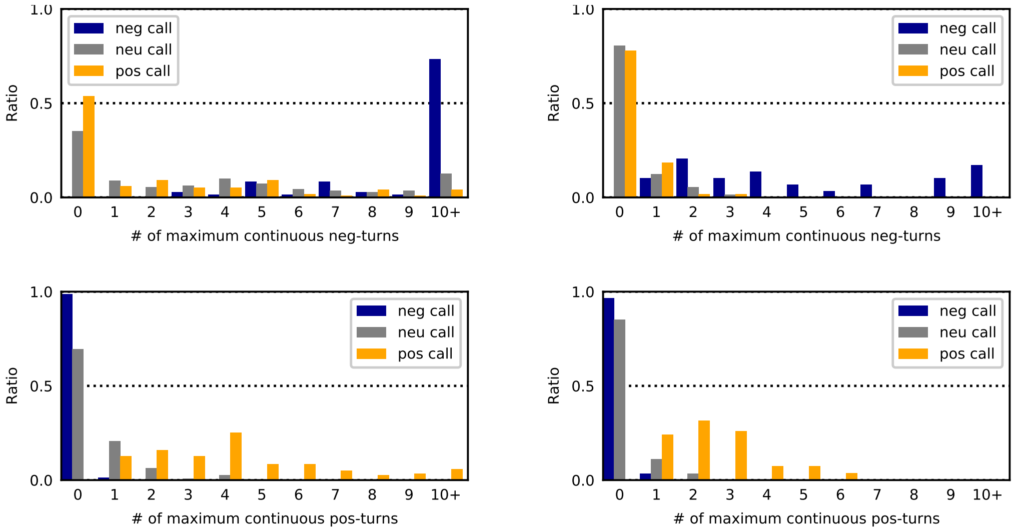
3.3.2 Analyses

We conducted several analyses to reveal the characteristics of call-level and turn-level CS.

First, the average lengths of calls and turns were investigated. The results are shown in Table 3.2. In both datasets, *pos* calls and *neg* turns were slightly longer than the others in call-level/turn-level comparisons. However, the distributions of the durations in each CS class were heavily overlapped.

Table 3.3: *Frequencies of neg, neu and pos turns in each call-level CS.*

	call-*neg*			call-*neu*			call-*pos*		
	turn-*neg*	turn-*neu*	turn-*pos*	turn-*neg*	turn-*neu*	turn-*pos*	turn-*neg*	turn-*neu*	turn-*pos*
acted	17.9 ± 8.3	7.8 ± 6.3	0.0 ± 0.1	5.1 ± 5.8	19.8 ± 7.5	0.7 ± 1.4	2.7 ± 4.2	17.1 ± 6.3	7.4 ± 4.1
real	8.7 ± 6.0	13.2 ± 7.6	0.0 ± 0.2	0.4 ± 0.9	19.5 ± 8.8	0.2 ± 0.6	0.5 ± 1.5	21.6 ± 14.2	3.8 ± 2.2

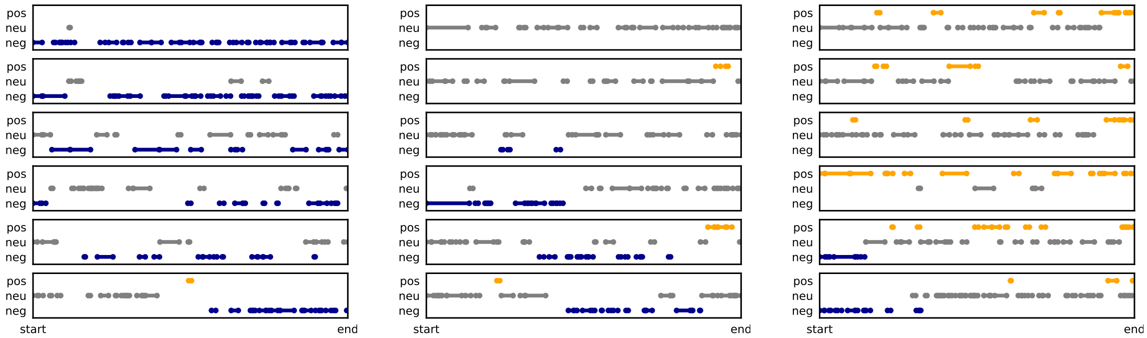


(a) *acted*

(b) *real*

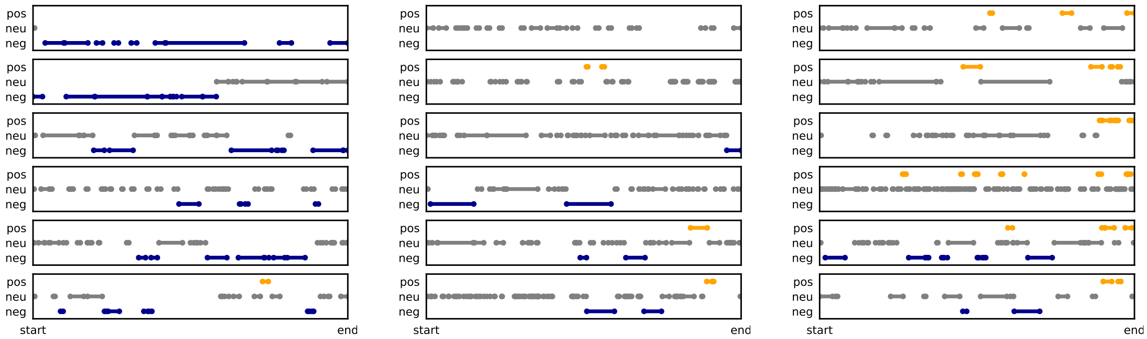
Figure 3.2: *Ratios of maximum continuous neg, pos turns in individual calls.*

To reveal the relationship between call-level and turn-level CS, average frequencies of *pos*, *neu* and *neg* turns were evaluated, see Table 3.3. The overall average turn frequencies in a call were 26.3 (acted) and 21.0 (real). Table 3.3 indicates that the distributions of turn-level CS labels were different in each call-level CS. Furthermore, several interesting characteristics were found between call-level and turn-level CS; disagreement of two-level CS values and proportion of turn-level CS. Though there were almost no *pos* turns in *neg* calls, there were a few *neg* turns in some *pos* calls. In terms of the proportion, the ratio of *pos* turns was usually less than 50 % even in *pos* calls. This was true in both the real and the acted datasets. However, the frequencies of



(a) *call-neg* (b) *call-neu* (c) *call-pos*

Figure 3.3: *Examples of turn sequences in the acted dataset.*



(a) *call-neg* (b) *call-neu* (c) *call-pos*

Figure 3.4: *Examples of turn sequences in the real dataset.*

pos and *neg* turns in the acted calls were slightly higher than those in the real calls. It is considered that the types of emotions appearing in real contact center calls may be biased. For example, most customers in technical support center tend to have low-arousal emotions, i.e. an emotion lying in the low-arousal area on Russell's circumplex model [18] like satisfied, neutral and disappointed, while the acted calls include a fuller variety of customer emotions such as delighted and enraged.

The frequencies of maximum continuous *pos* and *neg* turns were also investigated to reveal contextual characteristics, see Figure 3.2. There were also common cues in both the real and the acted dataset; *neg* turns in *neg* calls tended to continue longer than *pos* turns in *pos* calls. There were almost no *neu* and *neg* calls that had more than three continuous *pos* turns. However, the continuity of *neg* turns differed between the datasets. Several *neu* and *pos* calls containing *neg* turns continued for more than five turns in the acted dataset, while such sequences did not exist in the real calls. These findings suggest two hypotheses which will be useful in turn-level CS estimation. First, turn-level CS strongly depends on contextual information of the call. Second, call-level CS helps to identify turn-level CS because it is related to the continuity of turn-level CS.

Finally, we investigated the changes in turn-level CS. Figures 3.3 and 3.4 show examples of sequences of turn-level CS. A line connecting two dots represents a turn. The figure indicates that there were certain properties of turn-level CS changes in the datasets. In *neg* calls, several continuous *neg* turns appear in arbitrary positions of the calls. On the other hand, typical *pos* calls have *pos* turns in the middle and at the end of the calls. It is considered that customers in *pos* calls tend to show positive attitudes at the end of each topic and closing talk to express their gratitude, while in *neg* calls they verbalize the bad feelings over some duration at any time. The figures also show that there are several domain-independent categories in each call-level CS. For example, *neg* calls may have three types; the customer has bad emotion entirely (first row), in particular regions (from second to fifth rows) or with short positive response (sixth row). *neu* calls include few or no *pos* or *neg* turns in the calls (first to third rows), whose customers have negative feelings but recover (fourth row) or contain both *pos* and *neg* turns but overall are neutral (fifth and sixth rows). *pos* calls can be categorized into two; *pos* turns appear in the middle and end of calls (first to fourth rows), or customers have negative feelings at first but recover and finally acquire good emotion (fifth and sixth rows). These details indicate an important hypothesis for call-level

Table 3.4: *Results of N-gram modeling of turn-level CS.*

		Perplexity			
dataset	w/ call-level CS	1-gram	2-gram	3-gram	4-gram
acted		2.75	1.91	1.84	1.83
	✓	2.21	1.75	1.72	1.71
real		1.53	1.45	1.44	1.44
	✓	1.41	1.39	1.38	1.38

CS estimation; call-level CS can be identified only by the series of turn-level CS, and the identification rule is slightly domain dependent.

To confirm the hypotheses listed above, we conducted two pre-experiments. The first was N-gram modeling of the turn-level CS labels with or without call-level CS information. The purposes were to measure the valid lengths yielding consistent contextual information and the effectiveness of call-level CS. The task was predicting current turn-level label from the previous N-1 labels. Speaker-open 8-fold cross validation was employed. The N-gram model was trained by 7 folds and evaluated by the other 1 fold. A single N-gram was trained and evaluated without the call-level CS condition. In the case of with call-level CS information, N-gram models were trained and evaluated separately in each call-level CS. Lengths of N-grams ranged from 1 to 4. The evaluation measure was perplexity which reflects the ambiguity of turn-level CS prediction: lower is better. The results are listed in Table 3.4. In both acted and real datasets, perplexity decreased with longer context. There were significant differences from 1 to 3 grams in the acted and 1 to 2 grams in the real dataset ($p < 0.05$ in paired samples t-test). Call-level CS information also reduced perplexity ($p < 0.05$). These results demonstrate that utilizing certain length of context and call-level CS information is effective in the turn-level CS estimation task.

Table 3.5: *Results of call-level CS estimation from turn-level CS ground truths.*

		acted		real	
		Acc.	macroF1	Acc.	macroF1
Chance		0.333	0.329	0.333	0.263
MajorityClass		0.395	0.188	0.782	0.292
DominantTurn		0.548	0.513	0.808	0.467
FinalTurn		0.786	0.794	0.813	0.572
ULSTM	in-corpus	**0.880**	**0.889**	**0.903**	**0.840**
	cross-corpus	0.807	0.798	0.895	0.770

The second was call-level CS estimation from turn-level CS ground truths. This investigated whether call-level CS can be identified from just turn-level CS. Cross-corpus evaluations were also conducted to measure domain dependency. To learn a series of turn-level CS, 1-layer unidirectional LSTM (ULSTM) with 64 hidden units was used as the call-level CS estimation model. The input was the sequence of one-hot representations of turn-level CS ground truths. The estimation model was trained by minibatch training with a cross entropy-based loss function. The dropout ratio was 0.5. The minibatch size was 3 calls. Optimization rule was Adam [87] and learning rate was 0.001. Speaker-open 8-fold cross validation was used in the evaluation. One subset was test, another was development, and the remaining 6 were used as training data. Early-stopping was realized by the development subset [88]. Evaluation measures were accuracy (Acc.) and macroF1, which is the macro-average of the F1 values of each class. Results are shown in Table 3.5. *Chance* and *MajorityClass* mean the chance ratio and the results that all estimations lay in the majority class of the training set, respectively. *DominantTurn* and *FinalTurn* represent the results yielded by selecting the dominant class or the final turn's class in each call as call-level CS. The table shows that a simple rule such as selecting the dominant is insufficient. *FinalTurn* represents measurable performance in the acted dataset, but not in the real dataset. It is considered that this is due to the differences in the frequencies of low-arousal emotional calls. We found that some low-arousal calls like disappointed or frustrated calls showed *neg* turns only in the middle of calls, which make it impossible to estimate call-level CS by *FinalTurn*. Most of the calls in the real dataset were low-arousal ones, as shown in Table 3.3, which provides a plausible reason for the degraded performances of *FinalTurn* when applied to the real dataset. On the other hand, ULSTM-based estimation model trained by in-corpus data attained around 90% accuracy in both datasets. Furthermore, the performance still exceeded 80% with cross-domain training. These results indicate that call-level CS can be evaluated from just turn-level CS information, and that the estimation criteria from turn-level CS are slightly domain dependent.

3.4 Customer Satisfaction Estimation based on a Hierarchical Multi-Task Model

This section describes a CS estimation method based on the HMT model. It estimates both call-level and turn-level CS classes simultaneously. The key idea of the proposed method is utilizing the characteristics of call-level and turn-level CS as revealed in Section 3.3.2; contextual information, relationships of the two levels of CS values, and domain independency. Two LSTM-RNNs are employed to capture contextual information for turn-level and call-level CS. Estimated turn-level CS values are directly used for call-level estimation. The two estimation models, call-level and turn-level ones, are jointly optimized to acquire the relationship of two-level CS, similar to multi task learning. Furthermore, the estimation model is adapted to a particular domain by updating only the domain-dependent part of the model.

The HMT model is inspired by a Joint Many-Task model, which was originally proposed for natural language processing [82]. The difference is that the HMT model has a chained structure and the upper network utilizes only the output of the lower network, while the Joint Many-Task model is branched and the upper uses both lower-network outputs and input features. It is reasonable to employ the HMT model for two-level CS estimation because it is enough to evaluate call-level CS from turn-level CS information alone, which enables the number of model parameters to be reduced. Another advantage is that the training by the call-level label approach of the HMT model yields correction of the turn-level estimation results, which indicates that training by just call-level labels can improve not only call-level but also turn-level estimation.

3.4.1 The Hierarchical Multi-Task Model

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$ be a sequence of turn-level features in a call and N be the total number of customer turns. $\boldsymbol{x}_n$ includes information of not only the n-th customer turn but also the previous agent turn and the surrounding backchannels, as described in Section 3.4.4. Call-level and turn-level CS estimation are formulated as estimating call-level label $l^{(c)}$ and the series of turn-level labels $\boldsymbol{l}^{(t)} = [l_1^{(t)}, \cdots, l_N^{(t)}]$ corresponds to the turn-level features from $\boldsymbol{X}$. Individual labels $l_n^{(t)}, l^{(c)} \in L$ where L is a set of CS labels, e.g., $\{$*pos*, *neu*, *neg*$\}$. In the HMT model, estimated call-level and turn-level labels are decided by all the turn-level features in a call:

$$\hat{l}^{(c)} = \operatorname*{argmax}_{l^{(c)}} P(l^{(c)} \mid \boldsymbol{X}, \boldsymbol{\Theta}), \tag{3.1}$$

$$\hat{l}_n^{(t)} = \operatorname*{argmax}_{l_n^{(t)}} P(l_n^{(t)} \mid \boldsymbol{X}, \boldsymbol{\Theta}), \tag{3.2}$$

where $\boldsymbol{\Theta}$ is a set of the parameters in the HMT model[1].

[1] In this chapter, notation $^{(t)}$ means a variable for turn-level CS estimation. $^{(c)}$ is for call-level task.

To estimate the n-th turn-level labels from the n-th turn-level features and the surroundings, hidden representation of turn-level contextual information $\boldsymbol{h}_n^{(t)}$ is extracted first. The extraction formula depends on whether the turn-level estimation situation is online or batch (see Section 3.2). In the online situation, $\boldsymbol{h}_n^{(t)}$ is defined by the first to the n-th turn-level features,

$$\boldsymbol{h}_n^{(t)} = \mathsf{ULSTM}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n; \boldsymbol{\theta}_h^{(t)}), \tag{3.3}$$

where $\mathsf{ULSTM}()$ is a unidirectional LSTM-RNN function and $\boldsymbol{\theta}_h^{(t)}$ is the set of its parameters. In the batch situation, all turn-level features in a call are available,

$$\boldsymbol{h}_n^{(t)} = \mathsf{BLSTM}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N, n; \boldsymbol{\theta}_h^{(t)}), \tag{3.4}$$

where $\mathsf{BLSTM}()$ is a bidirectional LSTM-RNN function.

The posterior probabilities are defined by the hidden representation with a linear transformation layer and softmax function,

$$\boldsymbol{v}_n^{(t)} = \mathsf{LINEAR}(\boldsymbol{h}_n^{(t)}; \boldsymbol{\theta}_v^{(t)}), \tag{3.5}$$

$$\boldsymbol{y}_n^{(t)} = \mathsf{SOFTMAX}(\boldsymbol{v}_n^{(t)}), \tag{3.6}$$

where $\mathsf{LINEAR}()$ is a fully-connected linear transformation layer whose parameters are $\boldsymbol{\theta}_v^{(t)}$; $\mathsf{SOFTMAX}()$ is a softmax function. $\boldsymbol{y}_n^{(t)}$ represents the turn-level CS posterior probability vector of the n-th turn. $\boldsymbol{\theta}^{(t)} = \{\boldsymbol{\theta}_h^{(t)}, \boldsymbol{\theta}_v^{(t)}\}$ is the parameter set of turn-level CS estimation results yielded by turn-level features.

In call-level CS estimation, the HMT model utilizes estimated turn-level CS posteriors $\boldsymbol{y}_n^{(t)}$ rather than turn-level features because the series of turn-level CS has sufficient power to permit call-level estimation, as shown in Section 3.3.2. The hidden representation of contextual information of estimated turn-level CS sequence $\boldsymbol{h}^{(c)}$ is defined as,

$$\boldsymbol{h}^{(c)} = \mathsf{LSTM}(\boldsymbol{y}_1^{(t)}, \cdots, \boldsymbol{y}_N^{(t)}; \boldsymbol{\theta}_h^{(c)}), \tag{3.7}$$

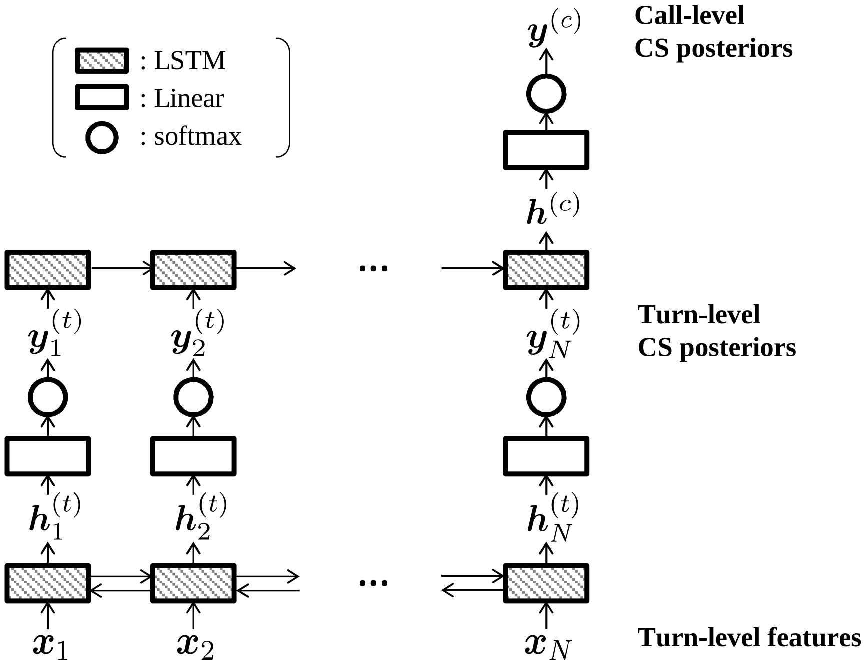


Figure 3.5: *The structure of the Hierarchical Multi-Task (HMT) model.*

where LSTM is either unidirectional or bidirectional LSTM-RNNs with parameters of $\boldsymbol{\theta}_h^{(c)}$. Similar to the turn-level estimation, posterior probabilities of call-level CS are estimated by linear projection with softmax function,

$$\boldsymbol{v}^{(c)} = \mathsf{LINEAR}(\boldsymbol{h}^{(c)}; \boldsymbol{\theta}_v^{(c)}), \tag{3.8}$$

$$\boldsymbol{y}^{(c)} = \mathsf{SOFTMAX}(\boldsymbol{v}^{(c)}), \tag{3.9}$$

where $\boldsymbol{y}^{(c)}$ is the call-level CS posterior probability vector and $\boldsymbol{\theta}_v^{(c)}$ is the parameter set of the linear layer. $\boldsymbol{\theta}^{(c)} = \{\boldsymbol{\theta}_h^{(c)}, \boldsymbol{\theta}_v^{(c)}\}$ is the parameter set of call-level CS estimation results yielded by estimated turn-level CS posteriors.

The structure of the proposed HMT model is shown in Figure 3.5. The following sections show two types of training methodology for the HMT model: flat-start and domain adaptation.

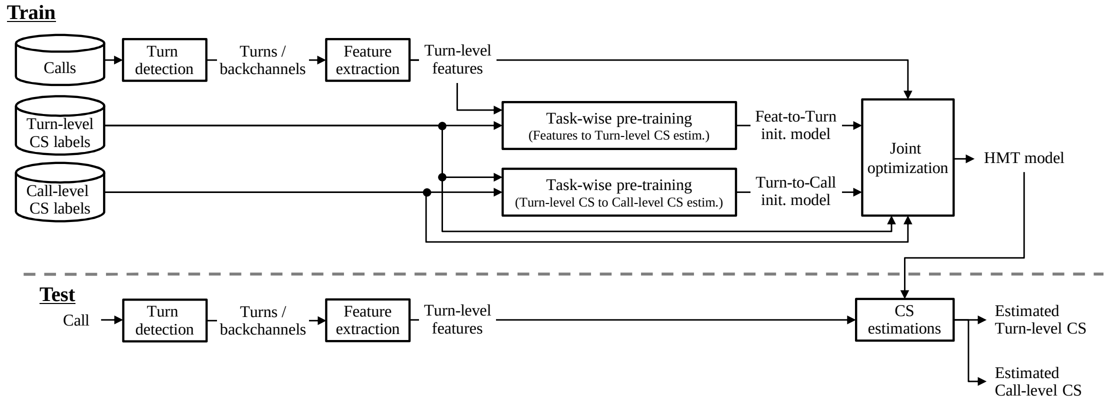


Figure 3.6: *The flow of joint optimization with task-wise pre-training of the HMT model.*

3.4.2 Training of the HMT Model

Two types of training, two-step training and joint optimization, are presented for training the HMT model from scratch. In both methods, the model parameters $\boldsymbol{\Theta} = \{\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)}\}$ are updated by stochastic gradient descent with loss functions. The loss functions of turn-level and call-level estimation are based on cross entropy,

$$\mathcal{L}_t(\boldsymbol{\theta}^{(t)}) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{l_n^{(t)}\in L} l_n^{(t)} \log P(l_n^{(t)} \mid \boldsymbol{X}, \boldsymbol{\theta}^{(t)}), \tag{3.10}$$

$$\mathcal{L}_c(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)}) = -\sum_{l^{(c)}\in L} l^{(c)} \log P(l^{(c)} \mid \boldsymbol{X}, \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)}). \tag{3.11}$$

The first, named two-step training, optimizes turn-level estimation parameters $\boldsymbol{\theta}^{(t)}$ and call-level parameters $\boldsymbol{\theta}^{(c)}$ sequentially and independently. $\boldsymbol{\theta}^{(t)}$ is optimized at first by turn-level estimation loss Eq. (3.10). Then $\boldsymbol{\theta}^{(c)}$ is updated with fixed turn-level estimation parameters $\bar{\boldsymbol{\theta}}^{(t)}$,

$$\mathcal{L}(\boldsymbol{\theta}^{(c)}) = \mathcal{L}_c(\bar{\boldsymbol{\theta}}^{(t)}, \boldsymbol{\theta}^{(c)}), \tag{3.12}$$

where $\mathcal{L}(\boldsymbol{\theta}^{(c)})$ is the loss function used to acquire $\boldsymbol{\theta}^{(c)}$.

The second is joint optimization with task-wise pre-training. All parameters are jointly optimized by both labels in order to utilize the relationship of two-level CS values to enhance estimation performance. The entire loss of the HMT model $\mathcal{L}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)})$ is represented as the weighted sum of Eq. (3.10) and (3.11),

$$\mathcal{L}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)}) = \alpha\mathcal{L}_t(\boldsymbol{\theta}^{(t)}) + (1-\alpha)\mathcal{L}_c(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(c)}), \tag{3.13}$$

where α is a loss weight that controls the convergence of the individual estimation. In preliminary works, we confirmed that the call-level estimation parts tend to converge faster than the turn-level ones because call-level estimation is much easier. To stabilize the HMT model training, a pre-training technique called task-wise pre-training is employed. First, turn-level estimations from turn-level features and call-level estimations from turn-level ground truth labels are trained separately as pre-training operations. The resulting pre-trained parameters are used as initial weights of the HMT model; the whole model is then fine-tuned by the entire loss (Eq. (3.13)). The flow of joint optimization with task-wise pre-training is shown in Figure 3.6.

3.4.3 Adaptation of the HMT Model

In domain adaptation, i.e., adjusting the HMT model to suit specific-domain calls, we hypothesize that it is effective to update just the domain-specific part of the model. Thus a new adaptation framework named call-net freezing is employed. It offers adaptation with both turn-level and call-level labels, and with call-level labels only. The outline of the adaptation by call-net freezing is shown in Figure 3.7.

The results in Section 3.3.2 indicate that call-level CS estimation from turn-level CS information is less domain independent. On the other hand, it is reported that turn-level CS estimation will be rather domain dependent [44]. From these findings,

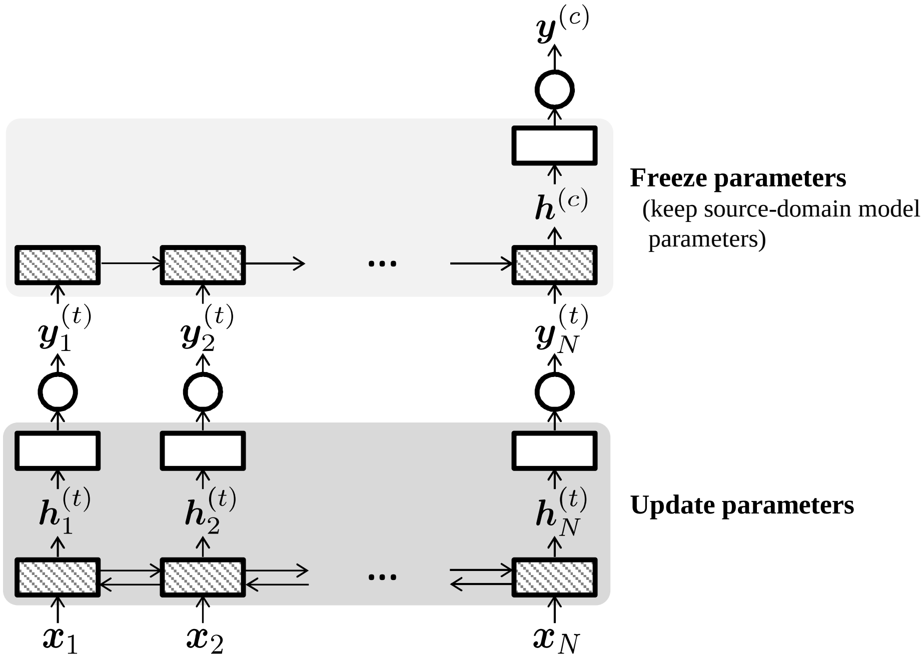


Figure 3.7: *Adaptation by the proposed call-net freezing.*

we consider that it is suitable for domain adaptation of the HMT model to adjust turn-level estimation criteria (domain dependent) while keeping constant the call-level criteria (domain independent). This will be effective, especially with limited adaptation data because the number of the parameters to be optimized is significantly decreased.

In call-net freezing, we use the well-known technique called layer freezing. Parameters of turn-level estimation $\boldsymbol{\theta}^{(t)}$ are updated with fixed call-level estimation parameters $\bar{\boldsymbol{\theta}}^{(c)}$. The loss function with call-level and turn-level labels is given by,

$$\mathcal{L}(\boldsymbol{\theta}^{(t)}) = \alpha\mathcal{L}_t(\boldsymbol{\theta}^{(t)}) + (1-\alpha)\mathcal{L}_c(\boldsymbol{\theta}^{(t)}, \bar{\boldsymbol{\theta}}^{(c)}), \tag{3.14}$$

which means that call-level loss is used for updating only turn-level estimation part. $\bar{\boldsymbol{\theta}}^{(c)}$ is constant before and after adaptation.

Furthermore, the HMT model adaptation is possible with just call-level labels. This is because the HMT model has chained structures and updating by the call-level loss has the effect of correcting the estimation results of turn-level CS. The loss function with just call-level labels is defined as follows,

$$\mathcal{L}(\boldsymbol{\theta}^{(t)}) = \mathcal{L}_c(\boldsymbol{\theta}^{(t)}, \bar{\boldsymbol{\theta}}^{(c)}). \tag{3.15}$$

This indicates that only turn-level estimation sub-networks are updated by call-level loss alone. Adaptation using only call-level labels is desirable in practice because call-level labels have much lower annotation cost than turn-level labels.

3.4.4 Turn-level Features

The turn-level features used in the HMT model include prosodic, lexical and interactive features extracted from both customer and operator turns and backchannels. Most are inspired by the conventional studies [1,12,46]. We use hand-crafted features in order to realize robustness even with limited amounts of training data.

Prosodic Features

Prosodic features include the information of fundamental frequency (F0), loudness, and speech rate of a current customer turn. They reflect changes in the customer's pitch, power, talking speed and end-of-sentence stretching. It has 21 dimensions: mean, std., max, min, range and ratio of start/end 500 ms mean to entire mean of customer turn log F0, mean, std., and max of loudness, mean, std., max, min of first derivative of log F0 and loudness, speech rate of customer turn and previous operator turn (mora/sec), and duration of the end of phoneme of the target customer turn. The speech rates and phoneme duration are obtained by automatic speech recognition.

Lexical Features

As the lexical features, we use Bag-of-Words (BoW) of specific words in current customer turn or the previous operator turn. All the words are obtained by automatic speech recognition. Target words are empirically selected by three real contact center supervisors from a previous study [46] to decrease domain dependency: automatic selection of class-specific words like the entropy-based method [1] depend on the training corpus. The lexical features have 12 dimensions: total number of words in the current customer turn or previous operator turn, number of filler words, backchannel words, appreciation and its emphasis words (e.g. *'kindly'*), personal pronoun in the current customer turn, number of filler words, backchannel words, appreciation, humility, and apology words in the previous operator turn. Filler words include *'uh'* or *'hmm'* and backchannel words are *'hai (yes in English)'*, *'wakarimashita (I see)'*, etc.

Interactive Features

Interactive features include turn-taking, pause and backchannel information. In addition to the conventional methods [12], the proposed method utilizes the characteristics of backchannels around current customer turn because backchannels are related to the interests, politeness, and apology of customers and agents. The proposed method uses 11 dimensional interactive features: length of the target customer turn and the previous operator turn, length of pause between the target customer turn and the previous/next operator turn, length of interval between the target and previous customer turn, the ratio of length of the target customer turn to the sum of previous operator and target customer turn, frequency of customer and agent backchannels, average durations and number of repeated words in customer backchannels, and the ratio of average F0 of customer backchannels to customer turn. Backchannels are automatically determined

Table 3.6: *Overall accuracies and macroF1s of turn-level and call-level CS estimations.*

		acted				real			
		Turn		Call		Turn		Call	
	train method	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1
Chance		0.333	0.303	0.333	0.329	0.333	0.211	0.333	0.263
MajorityClass		0.605	0.251	0.395	0.188	**0.920**	0.319	**0.782**	0.292
SVM [12, 36, 44, 48]		0.556	0.528	0.631	0.627	0.555	0.345	0.638	0.462
FCNN [38, 50]		0.572	0.539	0.641	0.631	0.711	0.401	0.640	0.459
ULSTM		0.616	0.588	0.704	0.706	0.719	0.402	0.740	0.487
BLSTM		0.676	0.645	0.678	0.683	0.739	0.444	0.722	0.444
HMT (ULSTM + ULSTM)	two-step	0.616	0.588	0.711	0.718	0.719	0.402	0.753	0.523
	joint optim.	0.647	0.612	0.724	0.729	0.763	0.415	0.735	0.542
HMT (BLSTM + ULSTM)	two-step	0.676	0.645	**0.728**	0.724	0.739	0.444	0.724	0.555
	joint optim.	**0.681**	**0.646**	**0.728**	**0.730**	0.774	**0.459**	0.740	**0.571**

by the segmentation method shown in Section 3.2. The customer backchannels of the target turn are defined as those between previous and target customer turns.

3.5 Experiments

We conducted two evaluation experiments; flat-start and domain adaptation. In the flat-start evaluation, feature comparison and annotation harmonization assessment were also conducted to investigate the effects of the input turn-level features and ground-truth information.

3.5.1 Common Setups

The datasets shown in Section 3.3 were used in both evaluations. Performance was evaluated by customer-open 8-fold cross validation. Each fold contained 3 or 4 unique customers in the acted dataset, and the same number of unique customers as calls in the real dataset. In each fold, the combinations of customers were selected by hand in order to keep the similar proportions of *pos* / *neu* / *neg* calls as those of the entire corpus.

Table 3.7: *Comparisons of input turn-level features for turn-level and call-level CS estimation.*

	acted				real			
	Turn		Call		Turn		Call	
	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1
Chance	0.333	0.303	0.333	0.329	0.333	0.211	0.333	0.263
Pro	0.612	0.575	0.631	0.634	0.748	0.403	0.651	0.422
Lex	0.649	0.617	0.654	0.657	0.732	0.443	0.693	0.571
Int	0.597	0.547	0.538	0.550	**0.777**	0.428	0.688	0.469
Pro + Lex	0.665	0.635	0.704	0.702	0.765	0.441	0.711	0.559
Pro + Int	0.659	0.614	0.664	0.668	0.772	0.419	0.711	0.483
Lex + Int	0.664	0.628	0.664	0.673	0.774	**0.466**	0.695	**0.572**
Pro + Lex + Int	**0.681**	**0.646**	**0.728**	**0.730**	0.774	0.459	**0.740**	0.571

We used one fold as the test, another one as the development, and the remaining six folds as the training set.

In turn-level feature extraction, frame length of F0 and loudness were set at 64 ms and 5 ms shift, respectively. The F0 extraction method was based on dominant harmonic components [89]. A DNN-HMM acoustic model with a large-vocabulary weighted finite-state transducer language model was used to obtain words for lexical features. The acoustic model had 8 fully-connected hidden layers with 2048 nodes and 3072 outputs. Both acoustic and lexical models were trained by several hundred hours of transcribed real contact center calls. The speech recognition decoder was VoiceRex [90, 91]. All turn-level features were normalized against the training set to zero mean and unit variance.

We employed overall accuracy and macroF1, the macro-averaged F-measures of all classes, as evaluation measures. We considered that macroF1 was a more important indicator of performance than accuracy because both datasets were highly class-imbalanced. In class imbalanced datasets, accuracy is generally high when all the estimated results lie in the majority class, but the estimator lacks estimation robustness [92].

3.5.2 Flat-start Evaluation

A Support Vector Machine (SVM) and Fully-Connected Neural Networks (FCNN) were employed as baseline classifiers, following several studies [12,36,38,44,48,50]. The inputs were individual turn-level features for turn-level estimation and their statistics (mean and standard deviation in a call) for call-level estimation. The kernel of SVM was a radial basis function and its hyper-parameters were optimized in each validation by grid-search with the training and development sets. The structure of FCNN was 3-layer fully-connected with 128 units. The activation function was rectified linear. Minibatch size was 16 and optimization method was Adam [87]. The learning rate was 0.0005. SVM and FCNN were implemented on LIBSVM [93] and pytorch [94], respectively. To mitigate the class imbalance problem [95], the inverse values of the class frequencies were used as class weight in the training step of both methods[2].

Four model variants were compared to the baselines: ULSTM, BLSTM and two HMT models (ULSTM+ULSTM, BLSTM+ULSTM). The first two estimated two-level CS independently from turn-level features. The structure was 1-layer unidirectional or bidirectional LSTM with 128 units, 1-layer fully-connected with 3 units (output dimensions) and softmax layer. The last two stacked turn-level and call-level estimation models hierarchically, as shown in Figure 3.5. They were constructed by 1-layer unidirectional or bidirectional LSTM with 128 units, 1-layer fully-connected with 3 units, softmax (these are for turn-level estimation), 1-layer unidirectional LSTM with 64 units, 1-layer fully-connected with 3 units and softmax layer (for call-level estimation). The HMT models were trained by two-step training or joint optimization, see Section 3.4.2. In joint optimization of the HMT models, training and development data were the

[2] Preliminary experiments found that most of the estimation results without class weight became *neu* in the acted dataset.

same in both task-wise pre-training and fine-tuning. Two-step training acquired one-way feedback from turn-level to call-level CS, while joint optimization employed mutual interaction. Note that ULSTM and HMT (ULSTM+ULSTM) were suitable for on-line estimation of turn-level CS, while BLSTM and HMT (BLSTM+ULSTM) were effective for batch estimation, as described in Section 3.4.1. The dropout ratio was 0.5. Minibatch size was 3 calls in all conditions. Adam [87] was used for optimization. Learning rates were 0.0001 in joint optimization and 0.0005 in all other training approaches. These hyper-parameters were evaluated by grid-search and the combination that yielded the highest accuracies over all validations in both datasets were used as the final results. The variants of evaluated hyper-parameters were as follows; 32 / 64 / 128 / 256 hidden units and 1 / 2 / 3 layers in each ULSTM and BLSTM, 0.0 / 0.5 dropout ratio, 3 / 6 / 10 calls in minibatch size, Adam / momentumSGD optimization with 0.0001 / 0.0002 / 0.0005 / 0.001 learning rate. We tried small models, small batch-sizes and strong regularization because the amount of training data was quite limited. Loss weight was 0.6 as it yielded the highest performance among weights ranging from 0.1 to 0.9. Inverse class frequency in the training data was used as the class weight of the cross-entropy based loss function [96]. Early-stopping was triggered by the losses of the development set. All the proposed model variants were implemented by pytorch [94].

Results are shown in Table 3.6. *Chance* and *MajorityClass* take the same meanings as in Section 3.3.2; the chance ratio and the results that all estimations lay in the majority class of the training set, respectively. Note that turn-level results of ULSTM/BLSTM and HMT with two-step training were the same because they used the same model structure, input and training condition.

The results of call-level estimation are discussed first. Comparing SVM, FCNN, UL-

STM and BLSTM, all variants of LSTM-based models achieved higher performance in both datasets; the difference was statistically significant ($p < 0.05$ in McNemar's test). These indicate that contextual information was clearly effective. However, BLSTM was worse than ULSTM in call-level task. It is considered that the amount of training data for call-level estimation was so limited that ULSTM was more suitable as its parameters are small. Compared to ULSTM, the HMT variants with two-step training showed better performance. This indicates that estimated turn-level CS values are capable of call-level estimation even though they have quite limited dimensionality. Finally, the training methods of the HMT model were compared. Joint optimization attained higher macroF1s than two-step training in both HMT models and both tasks. However, the differences were not significant ($p > 0.05$). One possible explanation is that the call-level estimation data was limited. Further investigation with a greater volume of calls is a future task.

Next, turn-level estimation performances were compared. Similar to call-level estimation, all LSTM-based methods outperformed SVM with statistically significant differences, especially BLSTM-based models. It is considered that contextual information is useful even in turn-level CS estimation, and that batch estimation offers better performance than online evaluation. The improvements offered by joint optimization to two-step learning were small in the acted calls but significantly large ($p < 0.05$) in the real calls. It is considered that joint optimization may give some feedback from call-level CS labels, which made the HMT model more robust.

We summarize here the results of the flat-start evaluation: Contextual information is clearly effective in both types of estimation. The series of turn-level CS estimation results matches the discrimination performance of call-level CS. Joint optimization, which attempts to learn the relationships of two-level tasks, is useful in enhancing the

Table 3.8: *Annotation harmonization results. Bold means the best results in each CS estimation model.*

		acted				real			
		Turn		Call		Turn		Call	
	ground truths	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1
FCNN	1 annot.	0.545	0.520	0.588	0.563	0.680	0.393	**0.648**	0.463
	w/o harmo.	0.550	0.525	0.618	0.599	0.695	0.396	**0.648**	**0.469**
	w/ harmo.	**0.572**	**0.539**	**0.641**	**0.631**	**0.711**	**0.401**	0.640	0.460
HMT	1 annot.	0.649	0.628	0.724	0.717	**0.777**	0.447	**0.743**	0.549
(BLSTM + ULSTM)	w/o harmo.	0.671	0.642	0.714	0.715	0.773	0.451	0.721	0.551
	w/ harmo.	**0.681**	**0.646**	**0.728**	**0.730**	0.774	**0.459**	0.740	**0.571**

Table 3.9: *Domain adaptation results. Target domain is the real dataset and source domain is the acted dataset.*

		real-full				real-half			
		Turn		Call		Turn		Call	
label		Acc.	macroF1	Acc.	macroF1	Acc.	macroF1	Acc.	macroF1
-	*No adapt*	0.466	0.336	0.533	0.465	0.466	0.336	0.533	0.465
call + turn	Flat-start	0.774	0.459	0.740	0.571	0.788	0.432	0.714	0.490
	Fine-tune	0.767	0.469	0.772	0.619	0.739	0.448	0.724	0.557
	Call-net freezing	0.780	0.474	0.764	0.606	0.753	0.457	0.719	**0.568**
call	Fine-tune	0.783	0.465	0.774	0.597	0.764	0.459	0.735	0.558
	Call-net freezing	**0.801**	**0.481**	**0.780**	**0.620**	**0.793**	**0.475**	**0.738**	0.558

HMT model performance in both tasks.

3.5.3 Feature Comparison Evaluation

To reveal the ability of each type of turn-level feature, feature comparison evaluations were conducted. Prosodic (Pro), Lexical (Lex), Interactive (Int) features and their combinations were taken as the turn-level input features. The classifier was for the batch estimation; the HMT model with BLSTM for turn-level estimation and ULSTM for call-level estimation. The model structure and the training conditions were the same as in the previous evaluation.

Table 3.7 shows the feature comparison results. Among the individual features, lexical features showed the best macroF1 value. We consider that some words appeared infrequently but were decisive cues in terms of estimation. Prosodic features were worst for the real dataset though they were better than interactive features in acted calls. This indicates that real calls may have much wider variation in customer speaking style than the acted calls. This also was found in the combinations of the feature subsets; The macroF1 values with and without prosodic features were close for the real dataset. However, most combinations of features yielded better performance than the features in isolation.

The conclusions of this evaluation are as follows. The lexical and interactive features are the first and the second most effective pieces of information in both tasks, respectively. The prosodic features may be of little use in analyzing real calls.

3.5.4 Annotation Harmonization Evaluation

The effects of harmonizing annotations were also investigated. If the harmonization only slightly impacts CS estimations, a single annotator is sufficient to train the classification model, which is desired to reduce annotation cost.

Three types of the ground truths were compared; 1 annotator, without harmonization, and with harmonization. For 1 annotator, one of two annotators was randomly selected for every call and their call-level/turn-level CS classes were used as the ground truth. This is equivalent to having several annotators but every call is annotated by just one of them. Without harmonization meant that all calls have two ground truths and they treated as different data. With harmonization was described in Section 3.3.1 with the same results in Section 3.5.2. These ground truths were used only in the training and the development set and those in the test set were harmonized. FCNN and HMT (BLSTM+ULSTM) with joint optimization were used as the CS estimation model. All hyper-parameters were the same as those in Section 3.5.2.

Table 3.8 presents the results. In the acted dataset, harmonization yielded the best accuracies and macroF1s. The differences in accuracy between 1 annotator and with harmonization were significant except for call-level estimation by the HMT model ($p < 0.05$). On the other hand, in the real dataset the macroF1s with harmonization were mostly better than those of 1 annotator and without harmonization, the differences were not significant. Comparing the results of harmonization with those of 1 annotator and without harmonization, harmonization attained higher precision but lower recall in both *pos* and *neg* classes. It is considered that harmonization increased *neu* samples and decreased *pos* and *neg* samples in the training data, which is problematic in the real dataset because the number of *pos* and *neg* samples were limited. This issue may be resolved if more *pos* and *neg* calls and turns are available.

These results indicate that the harmonization of annotations improve CS estimation performance. However, the effects of the harmonization are limited if the dataset contains few *pos* and *neg* samples.

3.5.5 Domain Adaptation Evaluation

Finally, we evaluated the domain adaptation performance. The source domain was the acted dataset and the target was the real domain. This was done because the acted calls contained a wide range of satisfied, neutral and dissatisfied calls, while the real dataset exhibited a relatively small range of emotion. Two adaptation dataset sizes, full and half, were used to measure the performance with large and small amounts of adaptation data, respectively. Full meant the entire training set and half meant a random sampling. Furthermore, two types of ground-truth information, with call-level and turn-level labels and with call-level label only, were given in the adaptation scenario.

The baselines were no adaptation, flat-start training, and fine-tuning of the HMT model. No adaptation presented the model trained by the source domain dataset. Flat-start was the model trained by the adaptation data from scratch. Fine-tuning was updating all the parameters of the model by the adaptation data. The proposed adaptation was the call-net freezing shown in Section 3.5.5. We evaluated batch estimation situation, thus the model was the HMT with BLSTM for turn-level task and ULSTM for call-level. The learning rate was 0.0005 in flat-start training and 0.0001 in fine-tuning and call-net freezing. The remaining conditions such as minibatch size followed those of Section 3.5.2.

Table 3.9 presents the adaptation results. First, no-adaptation models yielded low accuracy and macroF1 in both call-level and turn-level estimation. This indicates a significant model mismatch given the different domains. In the case of adaptation with call-level and turn-level labels, macroF1s of the fine-tuning and call-net freezing were higher than those achieved by flat-start training. This was observed in both full and half adaptation sets. It is considered that the HMT model could learn certain general characteristics of call-level and turn-level CS (both acted and real calls), which were utilized in adaptation. Comparing call-net freezing with fine-tuning by two-level labels, call-net freezing showed better macroF1s except for call-level estimation with full dataset. This suggests that call-net freezing worked well, especially if the adaptation data is limited. Finally, we mention adaptation with just call-level labels. Interestingly, fine-tuning and call-net freezing with call-level labels improved not only call-level estimation performance but also that of the turn-level task. This indicates that the HMT model trained by the acted dataset learnt the relationship between call-level and turn-level CS, and thus corrected the turn-level estimation results yielded by the call-level labels. Note that in the full dataset, call-net freezing with call-level labels alone showed better performance than that with both levels of labels. One possible explanation is that the turn-level labels were extremely class-imbalanced (92% were *neu* turns) and the use of turn-level labels may lead to overfitting of *pos* and *neg* turns. Further evaluation by more class-balanced calls is desirable as a future task.

3.6 Summary

In this chapter, we presented a novel CS estimation method that evaluated both call-level and turn-level CS simultaneously to better analyze contact center dialogues. Data analyses using acted and real calls elucidated three important characteristics; dependency of context, relationship between call-level and turn-level CS, and domain independency in call-level CS estimation from the series of turn-level CS degree. The proposed method, called the Hierarchical Multi-Task (HMT) model, well utilized these characteristics. It employs two LSTM-RNNs to capture contexts in turn-level and call-level estimations. These two models are hierarchically stacked and jointly optimized to learn the relationship between the two tasks. Domain adaptation uses call-net freezing, which maintains the call-level CS estimation part of the model, because it was less domain dependent. Three experiments, flat-start evaluation, feature comparison and domain-adaptation, were conducted. The HMT model was clearly superior to the conventional SVM or fully-connected NN-based classifier for both call-level and turn-level CS estimation. A feature comparison revealed that lexical features yielded the greatest contribution, while dialog features were also useful. Adaptation experiments confirmed that different domains incurred significant model mismatch, and the proposed adaptation approach achieved the highest performance. Future work includes evaluations with larger quantities of more balanced contact center calls or self-reported labels. Other directions are improving the online estimation performance of turn-level CS and introducing new lexical features without heuristics.

4 Basic Emotion Classification in Natural Speech based on Listener-Dependent Emotion Perception Models

4.1 Introduction

Human speech is the most basic and widely-used form of daily communication. Speech conveys not only linguistic information but also other factors such as speaker and emotion, all of which are essential for human interaction. Thus, speech emotion recognition (SER) is an important technology for natural human-computer interaction. There are a lot of SER applications such as voice-of-customer analysis in contact center calls [44, 97], driver state monitoring [2] and human-like responses in spoken dialog systems [4].

SER can be categorized into two tasks: dimensional and categorical emotion recognition. Dimensional emotion recognition is the task of estimating the values of several emotion attributes present in speech [5]. Three primitive emotion attributes, i.e. valence, arousal, and dominance are commonly used [6]. Categorical emotion recognition is the task of identifying the speaker's emotion from among a discrete set of emotion categories [7]. The ground truth is defined as the majority of perceived emotion class as determined by multiple listeners. Comparing these two tasks, categorical emotion recognition is more suitable for most applications because it is easy to interpret. This chapter aims to improve categorical emotion recognition accuracy.

A large number of SER methods have been proposed. One of the basic approaches is based on utterance-level heuristic features including the statistics of frame-level acoustic features such as fundamental frequency, power, and Mel-Frequency Cepstral Coefficients (MFCC) as determined by a simple classifier [8, 52]. Though they can recognize several typical emotions, their performance is still far from satisfactory because emotional cues exhibit great diversity, which demands the use of hand-crafted features with simple criteria. In contrast to this approach, several recent studies have achieved remarkable improvements through the use of Deep Neural Network (DNN)-based classifiers [9, 10, 13, 23, 24, 53, 98–101]. The main advantage of DNN-based classifiers is that they can learn complex cues of emotions automatically by combining several kinds of layers. Recurrent Neural Network (RNN)-layers have been used to capture the contextual characteristics of utterances [100, 101]. Attention mechanism has also been employed to focus on the local characteristics of utterances [9, 101]. Furthermore, DNN-based models can utilize low-level features, e.g. log power-spectrogram or raw waveform, which have rich but excessively complex information that simple classifiers are unable to handle [7, 13].

However, SER is still a challenging task despite these advances. One of the difficulties lies in handling two types of individuality: speaker and listener dependencies. The way in which emotions are presented strongly depends on the speaker. It is reported that prosodic characteristics such as pitch and laryngealization differ among speakers [62]. This is similar in emotion perceptions, and depends on age [68], gender [69] and cultures [70] of listeners. Given these issues, speaker dependency has often been considered for SER [64,65]. However, the dependency of listeners has received little attention in SER tasks even though it influences the determination of the majority-voted emotions.

This chapter presents a new SER framework based on Listener-Dependent (LD) models. The proposed framework aims to consider the individuality of emotional perceptions. In the training step of the proposed method, LD models are constructed so as to learn criteria for capturing the emotion recognition attributes of individual listeners. This allows the LD models to estimate the posterior probabilities of perceived emotions of specific listeners. Majority-voted emotions can be estimated by averaging these posterior probabilities as given by LD models. Inspired by domain adaptation frameworks in speech processing, three LD models are introduced: fine-tuning, auxiliary input, and sub-layer weighting. The fine-tuning method constructs as many LD models as listeners, while the remaining models cover all listeners by a single model. We also propose adaptation frameworks that allow the LD models to handle unseen listeners in the training data. Experiments on two emotional speech corpora show the individuality of listener perception and the effectiveness of the proposed approach. The main contributions of this chapter are as follows:

1. A scheme to recognize majority-voted emotions by leveraging the individuality of emotion perception is presented. To the best of our knowledge, this is the first work to take listener characteristics into consideration for SER.
2. The performance of listener-oriented emotion perception is evaluated in addition to that of majority-voted emotion recognition. The proposed LD models show better performance than the conventional method in both metrics, which indicates that the proposed scheme is suitable for estimating not only majority-voted emotions, but also personalized emotion perception.

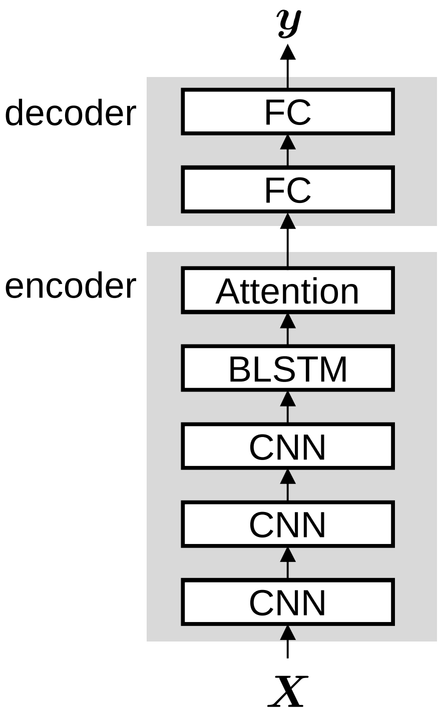
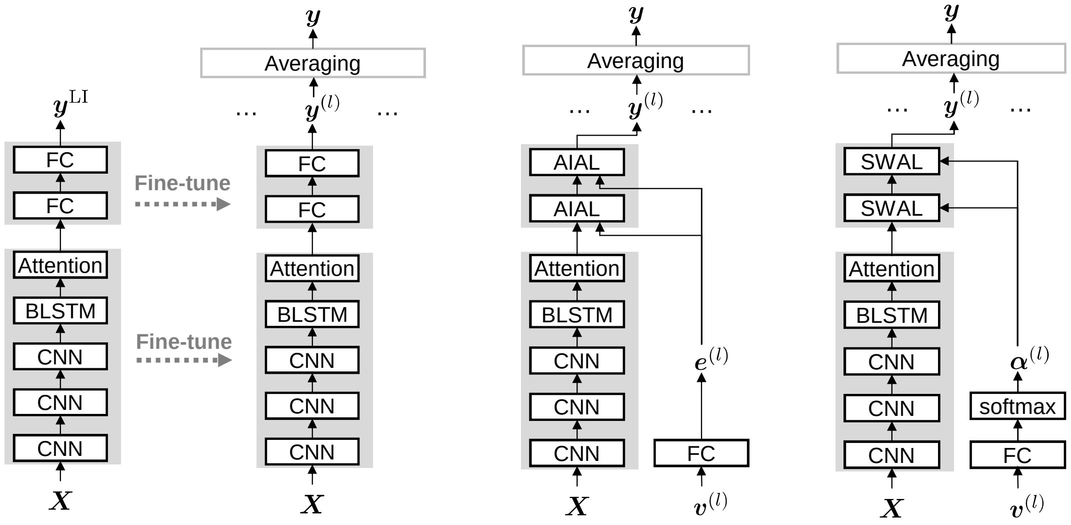


Figure 4.1: *An example of the conventional emotion recogntion model based on direct modeling of majority-voted emotion.*

This chapter is organized as follows. Conventional emotion recognition is shown in Section 4.2. The proposed framework based on LD models is shown in Section 4.3. Evaluation experiments are reported in Section 4.4 and the conclusion is given in Section 4.5.

4.2 Emotion Recognition by Majority-Voted Model

This section describes the conventional emotion recognition approach based on DNN model [9, 53]. In this chapter, we call this model the *majority-voted model* because it directly models majority-voted emotion of multiple listener perceptions.



(a) *Fine-tuning* (b) *Auxiliary input* (c) *Sub-layer weighting*

Figure 4.2: *Overview of the proposed majority-voted emotion recognition based on Listener-Dependent (LD) models.*

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T]$ be the acoustic features of an input utterance and T be their total length. $C = \{1, \cdots, K\}$ is the set of target emotion indices, e.g. 1 means neutral and 2 is happy. K is the total number of target emotions. The task of SER is formulated as estimating the majority-voted emotion of utterance $c \in C$ from $\boldsymbol{X}$,

$$\hat{c} = \operatorname*{argmax}_{c} P(c|\boldsymbol{X}), \tag{4.1}$$

where $\hat{c}$ is the estimated majority-voted emotion. $P(c|\boldsymbol{X})$ is the posterior probability indicated by the input utterance. The ground truth of majority-voted emotion c is defined as the dominant choice of multiple listener's perception results,

$$c \equiv \operatorname*{argmax}_{k} \sum_{l \in L} f(c^{(l)} = k), \tag{4.2}$$

where $c^{(l)} \in C$ is the perceived emotion of human listener l [1]. $f(\cdot)$ is a binary function of

[1] In this chapter, notation $^{(l)}$ means a listener-dependent variable.

emotion presence / absence, $f(c^{(l)} = k) = 1$ if l perceived the k-th target emotion from the utterance, otherwise 0. L is a set of the listeners annotated emotion perceptions given the input utterance, where $L \subset \mathbb{L}$ and $\mathbb{L}$ is a set of listeners in the training data. Note that the set of the listeners, L, can vary for each utterance in SER task.

The posterior probabilities of the majority-voted emotions $\boldsymbol{y} = [P(c = 1|\boldsymbol{X}), \cdots, P(c = K|\boldsymbol{X})]^\top$ are evaluated by the estimation model composed of an encoder and decoder. An example of the estimation model is shown in Fig. 4.1. The encoder projects an arbitrary length of acoustic features $\boldsymbol{X}$ into a fixed-length hidden representation in order to extract context-sensitive emotional cues. It consists of CNN, BLSTM, and self-attention layers such as a structured self-attention network [102]. The decoder estimates $\boldsymbol{y}$ from the hidden representation. It is composed of several Full-Connected (FC) layers.

The parameters of the estimation model are optimized by stochastic gradient descent with cross entropy loss,

$$\mathcal{L} = -\sum_c q(c) \log P(c|\boldsymbol{X}), \tag{4.3}$$

where $q(\cdot)$ is the reference distribution. $q(c = k)$ is 1 if the majority-voted emotion is the k-th target emotion, otherwise 0.

4.3 Emotion Recognition by Listener Dependent Models

This section proposes a majority-voted emotion recognition framework based on LD models. The key idea of our proposal is to consider the individuality of emotion perception. Every majority-voted emotion is determined from different sets of listeners in the SER task. However, the characteristics of emotional perceptions vary with the listener. Direct modeling of the majority-voted emotion will result in conflating multiple different emotion perception criteria, which may degrade estimation performance. To solve this problem, the proposed method constructs LD models to learn listener-specific emotion perception criteria.

This framework determines the posterior probability of majority-voted emotion by averaging the posterior probabilities of the listener-dependent perceived emotions,

$$P(c|\boldsymbol{X}) = \frac{1}{N_L} \sum_{l \in L} P(c^{(l)}|\boldsymbol{X}, l), \tag{4.4}$$

where N_L is the total number of listeners L. In vector representation,

$$\boldsymbol{y} = \frac{1}{N_L} \sum_{l \in L} \boldsymbol{y}^{(l)}, \tag{4.5}$$

where $\boldsymbol{y}^{(l)} = [P(c^{(l)} = 1|\boldsymbol{X}, l), \cdots, P(c^{(l)} = K|\boldsymbol{X}, l)]^\top$ is the listener-dependent posterior probability vector evaluated by the LD model.

In this thesis, three LD models are introduced: fine-tuning, auxiliary input, and sublayer weighting. All of them are inspired by adaptation techniques in speech processing. The proposed frameworks based on LD models are overviewed in Fig. 4.2.

4.3.1 Model Overview

Fine-Tuning based Model

A Listener-Independent (LI) model is retrained with specific listener training data to create a LD model. This is inspired by fine-tuning based domain adaptation in speech recognition [74].

Two-step training is employed. First, the LI model is trained with all utterances and their listeners in the training data. Listener-wise annotations are used for the reference distributions without distinguishing among listeners. The trained LI model outputs listener-independent posterior probabilities $\boldsymbol{y}^{\mathrm{LI}}$. Second, the LI model is retrained with a particular listener's labels and utterances. This yields as many isolated LD models as there are listeners in the training data.

The optimization methods of LI / LD models are cross-entropy loss with listener-dependent perceived emotion, see as Eq. (4.3),

$$\mathcal{L} = -\sum_{c^{(l)}} q(c^{(l)}) \log P(c^{(l)} | \boldsymbol{X}, l). \tag{4.6}$$

Auxiliary Input based Model

The second model adapts particular layers of the estimation model through the auxiliary use of listener information. This is inspired by speaker adaptation in speech recognition [76] and speech synthesis [77].

One-hot vector of listener l, $\boldsymbol{v}^{(l)}$, is used to enhance acoustic features. $\boldsymbol{v}^{(l)}$ is projected into listener embedding vector $\boldsymbol{e}^{(l)}$ by an embedding layer,

$$\boldsymbol{e}^{(l)} = \sigma(\boldsymbol{W}_e \boldsymbol{v}^{(l)} + \boldsymbol{b}_e), \tag{4.7}$$

where $\boldsymbol{W}_e, \boldsymbol{b}_e$ are the parameters of the embedding layer. σ is an activation function such as hyperbolic tangent. Then $\boldsymbol{e}^{(l)}$ is used as the auxiliary input of the adaptation layers, named Auxiliary Input-based Adaptation Layers (AIALs), in the decoder so as to adjust the decoder to the chosen listener,

$$\boldsymbol{h}_{a,o} = \boldsymbol{W}_a \left[\boldsymbol{h}_{a,i}^{\top}, \boldsymbol{e}^{(l)\top}\right]^{\top} + \boldsymbol{b}_a, \tag{4.8}$$

where $\boldsymbol{h}_{a,i}, \boldsymbol{h}_{a,o}$ are the input and the output of the AIAL, respectively, and $\boldsymbol{W}_a, \boldsymbol{b}_a$ are the parameters. Note that the embedding layer, the encoder and decoder are optimized jointly.

The advantage of the auxiliary input based approach is that it offers greater stability than fine-tuning based models. There are two reasons for this. First, it has fewer parameters than fine-tuning based models. The fine-tuning models have to store as many encoders and decoders as there are listeners. However, auxiliary input based models share the encoder and decoder among all listeners, which suppresses the number of parameters. Second, the auxiliary input model can utilize the similarity of listeners. The fine-tuning models learn for just particular listeners. On the other hand, similar listeners will be mapped into similar latent vectors by the projection function, which reinforces the encoder's ability to learn listener-dependent emotion perception.

Note that only the decoder of the LD model is adapted to the selected listener. We consider that every listener perceives the same emotional cues from acoustic features, e.g. pitch raise / fall and fast-talking, and decision making from the emotional cues depends on listeners.

Sub-Layer Weighting based Model

The sub-layer weighting approach combines multiple projection functions to adapt to the listener. This is inspired by Context Adaptive DNN (CADNN) proposed for source separation [78].

Sub-layer Weighting-based Adaptation Layers (SWALs) are used to adapt the decoder to the selected listener. SWAL consists of multiple FC sub-layers,

$$\boldsymbol{h}_{s,o} = \sum_{m=1}^{M} \alpha_m^{(l)} \left(\boldsymbol{W}_{s,m} \boldsymbol{h}_{s,i} + \boldsymbol{b}_{s,m} \right), \tag{4.9}$$

where $\boldsymbol{h}_{s,i}, \boldsymbol{h}_{s,o}$ are the input and output of the SWAL. $\boldsymbol{W}_{s,m}, \boldsymbol{b}_{s,m}$ is the parameters of the m-th sub-layer and M is the total number of sub-layers. $\alpha_m^{(l)}$ is the adaptation

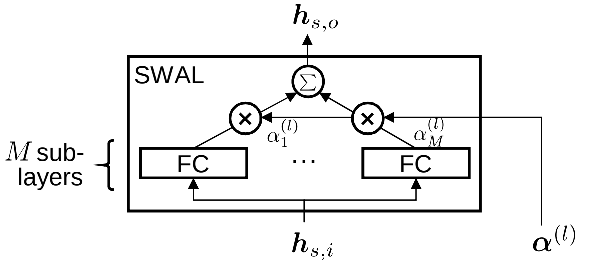


Figure 4.3: *Structure of the Sub-layer Weighting-based Adaptation Layer (SWAL).*

weight associated with the selected listener,

$$\boldsymbol{\alpha}^{(l)} = \mathsf{SOFTMAX}(\boldsymbol{W}_e \boldsymbol{v}^{(l)} + \boldsymbol{b}_e), \tag{4.10}$$

where $\boldsymbol{\alpha}^{(l)} = [\alpha_1^{(l)}, \cdots, \alpha_M^{(l)}]^\top$ is an adaptation weight vector determined by listener representation $\boldsymbol{v}^{(l)}$ and $\mathsf{SOFTMAX}(\cdot)$ is the softmax function. The structure of the SWAL is shown in Fig. 4.3. The model parameters including the embedding layer, Eq. (4.10), are jointly optimized as the auxiliary input approach.

The main advantage of sub-layer weighting is that it is more expressive than auxiliary input based models. Listener-dependent estimation is conducted by means of combining the perception rule of embedded listeners. However, it will require more training data than the auxiliary input-based approach because it has more parameters.

4.3.2 Adaptation to a New Listener

The LD models can be directly applied to listener-closed situations, i.e. evaluation listeners are present in the training data. Though common SER tasks are listener-closed, SER in practice is listener-opened so evaluation listeners are not included in the training set. Our solution is to propose adaptation methods that allow the LD models to handle open listeners using a small amount of adaptation data.

The adaptation for the fine-tuning based LD model can be achieved by the retraining method that is the same as the second step of the flat-start training of the model. The LI model constructed by the training set is fine-tuned using the adaptation utterances of the particular listener.

The auxiliary input and sub-layer weighting based LD models adapt to a new listener to estimate the most similar listener code in the training listeners. Let $\hat{\boldsymbol{v}}^{(l)}$ and $\boldsymbol{u}^{(l)}$ be the estimated listener code and its indicator whose sizes are the same as $\boldsymbol{v}^{(l)}$. The initial value of $\boldsymbol{u}^{(l)}$ is a zero vector. $\boldsymbol{u}^{(l)}$ is updated by backpropagating the loss of the adaptation data while freezing all the model parameters, as shown in Fig. 4.4. Note that the proposed adaptation does not update $\hat{\boldsymbol{v}}^{(l)}$ directly so as to restrict that the sum of $\hat{\boldsymbol{v}}^{(l)}$ to be 1 and all the dimensions to be non-negative, which is the same constraint as $\boldsymbol{v}^{(l)}$ in the training step. After the estimated listener vector $\hat{\boldsymbol{v}}^{(l)}$ is obtained from the adaptation data, it is fed to the LD models as the listener code, and the posterior probabilities of the perceived emotion of the new listener are derived. This approach is similar to those proposed in speech recognition [103].

4.4 Experiments

We evaluated the proposed LD models in two scenarios. The first was a flat-start evaluation. The estimation models were trained from scratch and evaluated by listeners present in the training dataset, i.e. a listener-closed condition. The second was an adaptation evaluation. It was a listener-open condition; the utterances and listeners separated from the training data were used for the adaptation and evaluation data to investigate estimation performance for unseen listeners.

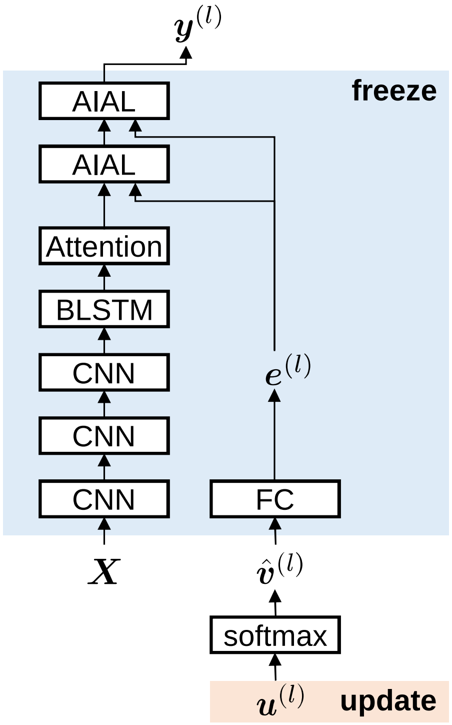


Figure 4.4: *Adaptation for the auxiliary input-based LD model.*

4.4.1 Datasets

Two large SER datasets, Interactive Emotional Dyadic Motion Capture (IEMOCAP) [104] and MSP-Podcast [105], were used in evaluating the proposal. IEMOCAP and MSP-Podcast contains acted and natural emotional speech, respectively. We selected four target emotions, neutral (*Neu*), happy (*Hap*), sad (*Sad*) and angry (*Ang*). All non-target emotion classes in the datasets were set as other (*Oth*) class.

IEMOCAP contains audiovisual data of 10 skilled actors (5 males and 5 females) in five dyadic sessions. The database consists of a total of 12 hours of English utterances generated by improvised or scripted scenarios specifically written to represent the emotional expressions. As in several conventional studies [9, 23, 54, 66, 99], we used only

Table 4.1: *Number of utterances in IEMOCAP.*

		Neu	*Hap*	*Sad*	*Ang*	*Oth*	Total
Majority		1099	947	608	289	0	2943
Listener	1	412	1166	589	284	456	2907
	2	951	876	586	269	99	2781
	3	1225	717	324	155	150	2571
	rest	226	119	113	56	56	570

audio tracks of the improvised set since scripted data may contain undesired contextual information. There are six listeners in the corpus and every utterance was annotated by three of them. The annotated categorical emotion labels are 10: neutral, happy, sad, angry, disgusted, excited, fearful, frustrated, surprised, and other. We combine happy and excited into *Hap* class in accordance with conventional studies [10, 53]. Though listeners were allowed to give multiple emotion labels to each utterance, to evaluate listener-wise emotion perception performance we unified them so that all listeners labeled one emotion per utterance. The unification rule was to select the majority-voted emotion if it is included in the multiple annotations, otherwise the first annotation is the unique perceived emotion. The listeners who gave fewer than 500 annotations were clustered as the "rest listeners" because they provided too little information to support learning listener-dependent emotion perception characteristics. Finally, the utterances whose majority-voted emotion is one of the target emotions were used to form the evaluation dataset. The numbers of utterances are shown in Table 4.1. The estimation performances were compared by leave-one-speaker-out cross-validation; one speaker was used for testing, another for validation, and the other 8 speakers were used for training.

MSP-Podcast contains English speech segments from podcast recordings. Collected

Table 4.2: *Number of utterances in MSP-Podcast.*

		Neu	*Hap*	*Sad*	*Ang*	*Oth*	Total
Majority		22681	12302	2351	2893	0	40227
Listener	1	5475	380	27	59	45	5986
	2	1130	1026	120	69	800	3145
	3	421	1072	191	128	440	2252
	...						...
	154	78	37	4	2	27	148
	rest	74524	57200	12459	14891	44470	203544

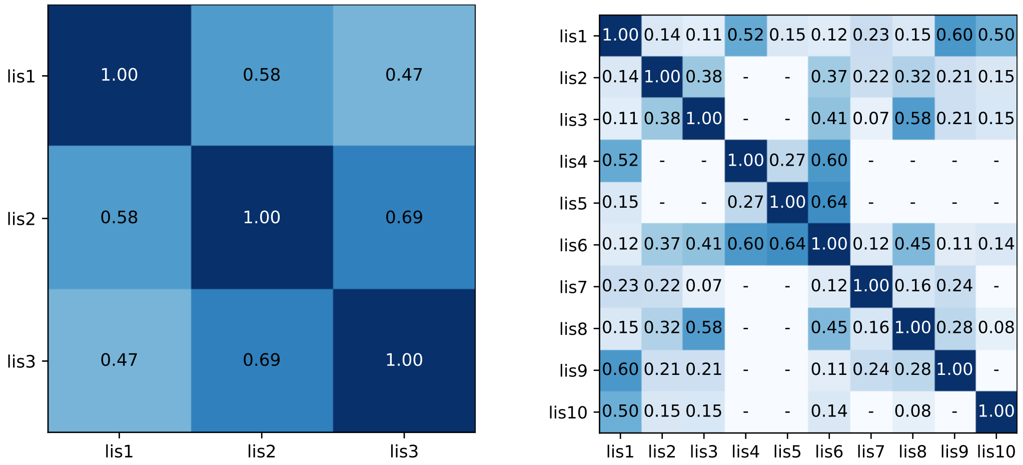
from online audio shows, they cover a wide range of topics like entertainment, politics, sports, etc. We used Release 1.7 which contains approximately 100 hours of speaking turns. Annotations were conducted by crowdsourcing. There are 11010 listeners and each utterance was annotated by at least three listeners (6.7 listeners per utterance on average). This dataset has two types of emotion annotations, primary and secondary emotions; we used only the primary emotions as listener-wise perceived emotions. The variety of annotated primary emotions consisted of neutral, happy, sad, angry, disgust, contempt, fear, surprise, and other. We used the utterances whose majority-voted emotions were one of the target emotions. A predetermined speaker-open subset was used in the flat-start evaluation; 8215 segments from 60 speakers for testing, 4418 segments from 44 speakers for validation, and the remaining 25332 segments from more than 1000 speakers for training. The listeners who gave fewer than 100 annotations in the training set were clustered as “rest listeners”, same as IEMOCAP. The total numbers of emotional utterances are shown in Table 4.2.

To clarify the impact of listener dependency on emotion perception, we first investigated the similarity of listener annotations. Fleiss’ and Cohen’s kappa coefficients were employed as the similarity measures of the overall and the individual pairs of listeners, respectively. The coefficients were calculated through 5-class matching (4 target emotions + *Oth*) from only the utterances in the evaluation dataset. Cohen's kappa coefficients of the listener pairs in which both listeners annotated less than the same 20 utterances were not evaluated ('-' in results). Fleiss' kappa were 0.57 in IEMOCAP and 0.35 in MSP-Podcast. There are two reasons for the lower consistency rate of MSP-Podcast. First, MSP-Podcast speech segments are completely natural, unlike IEMOCAP utterances which contained acted speech; this increased the ambiguous emotional speech in MSP-Podcast. Second, MSP-Podcast listeners will have larger diversity than those of IEMOCAP. All the listeners in IEMOCAP are students in the same university [104]. Cohen's kappa coefficients of IEMOCAP and MSP-Podcast listeners are shown in Fig. 4.5. It is shown that listener 2 showed relatively high similarity with listeners 1 and 3, while listeners 1 and 3 showed low similarity in IEMOCAP. The MSP-Podcast result showed the same property. Listener 1 showed high similarity with listeners 4, 9, 10, but low similarity with the remaining listeners. Listener 6 was similar to listeners 4 and 5. These indicate that emotion perception depends on listeners, and that there are several clusters of emotion perception criteria.

4.4.2 Flat-Start Evaluation

Setups

Log power spectrograms were used as acoustic features. The conditions used in extracting spectrograms followed those of conventional studies [23,106]. Frame length and frame shift length were 40 ms and 10 ms, respectively. The window type was Hamming window. DFT length was 1600 (10Hz grid resolution) and we used 0-4 kHz frequency range, which yielded 400-dimensional log power spectrograms. All the spectrograms



	lis1	lis2	lis3
lis1	1.00	0.58	0.47
lis2	0.58	1.00	0.69
lis3	0.47	0.69	1.00

(a) *IEMOCAP*

	lis1	lis2	lis3	lis4	lis5	lis6	lis7	lis8	lis9	lis10
lis1	1.00	0.14	0.11	0.52	0.15	0.12	0.23	0.15	0.60	0.50
lis2	0.14	1.00	0.38	-	-	0.37	0.22	0.32	0.21	0.15
lis3	0.11	0.38	1.00	-	-	0.41	0.07	0.58	0.21	0.15
lis4	0.52	-	-	1.00	0.27	0.60	-	-	-	-
lis5	0.15	-	-	0.27	1.00	0.64	-	-	-	-
lis6	0.12	0.37	0.41	0.60	0.64	1.00	0.12	0.45	0.11	0.14
lis7	0.23	0.22	0.07	-	-	0.12	1.00	0.16	0.24	-
lis8	0.15	0.32	0.58	-	-	0.45	0.16	1.00	0.28	0.08
lis9	0.60	0.21	0.21	-	-	0.11	0.24	0.28	1.00	-
lis10	0.50	0.15	0.15	-	-	0.14	-	0.08	-	1.00

(b) *MSP-Podcast*

Figure 4.5: *Cohen's kappa coefficients of listener annotations.*

were z normalized using the mean and variance of the training dataset.

The baseline was the majority-voted emotion recognition model described in Section 4.2. An ensemble of multiple majority-voted models with different initial parameters was also employed to compare with the proposed method that unifies several outputs of LD models. The number of ensembles was the average number of listeners per utterance, i.e. 3 and 7 in IEMOCAP and MSP-Podcast, respectively. The structure of the baseline is shown in Table 4.3. Each CNN layer was followed by batch normalization [107], rectified linear activation function, and 2×2 max pooling layers. Early stopping was performed using development set loss as the trigger. The optimization method was Adam [87] with a learning ratio of 0.0001. In the training step, inverse values of the class frequencies were used as class weights to mitigate the class imbalance problem [95]. Minibatch size was 8 in IEMOCAP and 16 in MSP-Podcast evaluations. Data augmentation was performed by means of speed perturbation with speed factors of 0.9, 0.95, 1.05, and 1.1 [7]. SpecAugment [108] was also applied with two time and frequency masking. The ensemble of multiple majority-voted models with different ini-

Table 4.3: *Network architectures of emotion recognition model.*

	Layer-type	Parameters
Encoder	CNN	16 ch, [12×16] kernel, [2×2] stride
	CNN	24 ch, [4×6] kernel, [1×1] stride
	CNN	32 ch, [3×4] kernel, [1×1] stride
	BLSTM	1 layer, 128 dim.
	attention	structured self-attention [102], 4 head
Decoder	FC / AIAL / SWAL	1 layer, 64 dim.
	FC / AIAL / SWAL	1 layer, 4 dim.

Table 4.4: *Number of model parameters.*

			IEMOCAP	MSP-Podcast
Baseline	Majority		1.04M	
	Majority (ens.)		3.11M	7.26M
Proposed	Fine-tuning	LI	1.04M	
		LD	4.15M	160.68M
	Auxiliary		1.04M	
	Weighting		1.09M	

tial parameters was also compared because the proposed method unifies the multiple outputs of LD models. The number of ensembled models was the average number of listeners per utterance, i.e. 3 and 7 in IEMOCAP and MSP-Podcast, respectively.

The proposals were LD models by fine-tuning, auxiliary input, and sub-layer weighting. LI model, the base model of the fine-tuning based LD model, was also compared to investigate the difference before and after fine-tuning. These model structures were the same as the baseline except for FC layers in the decoder, which were replaced with AIALs or SWALs. The numbers of listener embedding vector dimensions and sub-layers were 2, 3, 4, 8, 16 and we selected the best parameters for each dataset.

Table 4.5: *Estimation accuracies of the majority-voted emotions.*

			IEMOCAP		MSP-Podcast	
			WA	UA	WA	UA
Baseline	Majority		59.1	62.5	47.8	47.0
	Majority (ens.)		61.0	64.7	47.8	47.0
Proposed	Fine-tuning	LI	61.2	63.7	54.9	**48.9**
		LD	**62.9**	**65.2**	56.6	**48.9**
	Auxiliary		**62.9**	**65.2**	57.4	47.0
	Weighting		62.3	64.0	**58.7**	46.3

The learning ratio was 0.0001 and 0.00005 in flat-start and fine-tuning, respectively. The class weights were calculated by each listener in LD model training. The other training conditions and data augmentation setup were those of the baseline. All the baseline and the proposed methods were implemented by PyTorch [94]. Comparisons of the model parameters are shown in Table 4.4. The numbers of dimensions of listener embedding vector dimensions and sub-layers shown in the Table were 4.

Two evaluation metrics common in emotion recognition studies were employed; Weighted Accuracy (WA) and Unweighted Accuracy (UA). WA is the classification accuracy of all utterances and UA is the macro average of individual emotion class accuracies. We evaluated not only the performances of majority-voted emotion estimation but also those of listener-wise emotion recognition to investigate the capability of the proposed LD models.

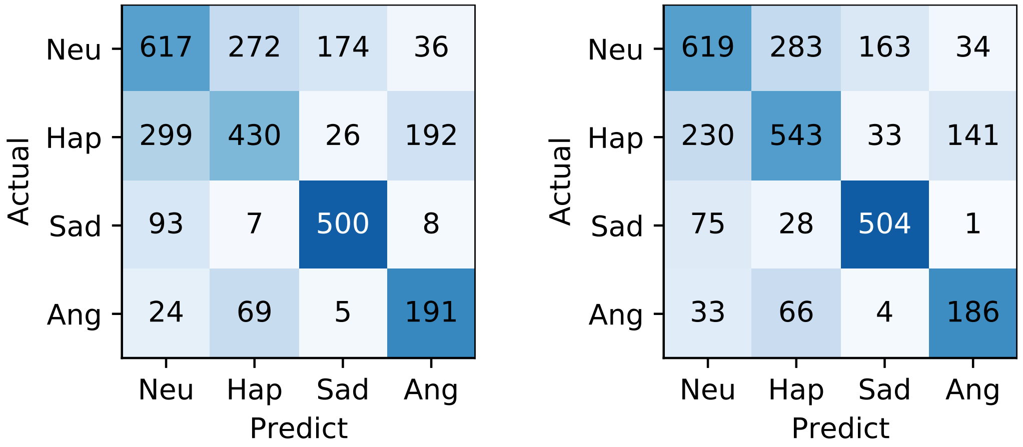
Results

The results of majority-voted emotion estimation are shown in Table 4.5. The notation Majority (ens.) means the ensemble result of the majority-voted models. Com-

Table 4.6: *Macro average of estimation accuracies of the listener-dependent perceived emotions.*

			IEMOCAP		MSP-Podcast	
			WA	UA	WA	UA
Baseline	Majority		57.0	62.0	44.9	43.2
	Majority (ens.)		58.9	64.1	44.0	43.3
Proposed	Fine-tuning	LI	59.7	63.6	50.0	44.3
		LD	**61.6**	63.3	51.8	44.8
	Auxiliary		61.5	**64.7**	52.1	**45.1**
	Weighting		60.9	63.8	**53.1**	**44.9**

paring the two datasets, MSP-Podcast yielded lower overall accuracy than IEMOCAP. It is considered that MSP-Podcast contains natural speech with a large number of speakers, which makes it more difficult to recognize emotion than IEMOCAP, which holds acted utterances from limited speakers. The LD models showed significantly better WAs ($p < .05$ in paired t-test) as almost the same or better UAs than the baselines on both datasets. For example, fine-tuning based LD models achieved 3.8 % and 2.7 % improvements from the single majority-voted model in WA and UA for IEMOCAP, 8.8% and 1.9% for MSP-Podcast. These results indicate that majority-voted emotion recognition based on LD models is more effective than the conventional majority-voted emotion modeling framework. Fig. 4.6 and 4.7 show the confusion matrices of the baseline and the auxiliary input-based LD model. Comparing numbers of the corrected samples for each emotion, *Hap* was improved on both IEMOCAP and MSP-Podcast, while *Sad* and *Ang* were degraded on MSP-Podcast. One possible reason for the degradation is data imbalance. These two emotions were hardly observed by some listeners, e.g. listener 154 annotated only 2 utterances with *Ang* emotion as shown in Table 4.2, which leads to overfitting in the LD model. Comparing the LD



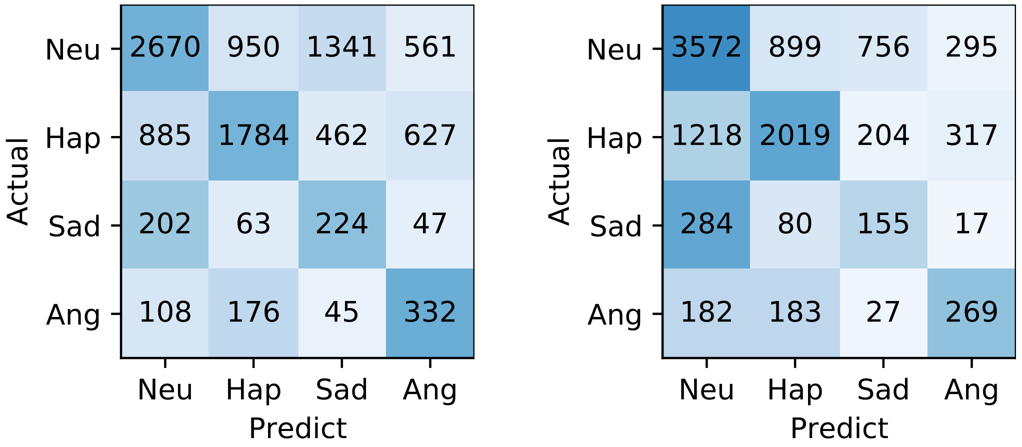
(a) *Majority-voted model*

(b) *LD model*

Figure 4.6: *Confusion matrices for IEMOCAP.*

models, there were no significant differences ($p \geq .05$), while fine-tuning and auxiliary input were slightly better for IEMOCAP, while sub-layer weighting yielded the best WA and fine-tuning attained the best UA for MSP-Podcast. Taking the number of parameters, see Table 4.4, into consideration, the auxiliary input based model is suitable for all conditions, while sub-layer weighting may become better for large datasets. Note that even the LI model significantly outperformed the model ensemble baseline in MSP-Podcast ($p < .05$). One possible reason is that training by listener-specific labels allows the model to learn inter-emotion similarities. For example, a set of listener-wise labels {*neu*, *neu*, *hap*} indicates that the speech may contain both *neu* and *hap* cues. On the other hand, its majority-voted label just indicates the speech has *neu* characteristics.

Macro averages of listener-wise emotion recognition performances are shown in Table 4.6. In this evaluation, WA / UAs of all the listeners except for "rest listeners" were averaged to compare overall performance. Table 4.6 represents that all LD models showed better performance than the baseline. The improvements were significant in

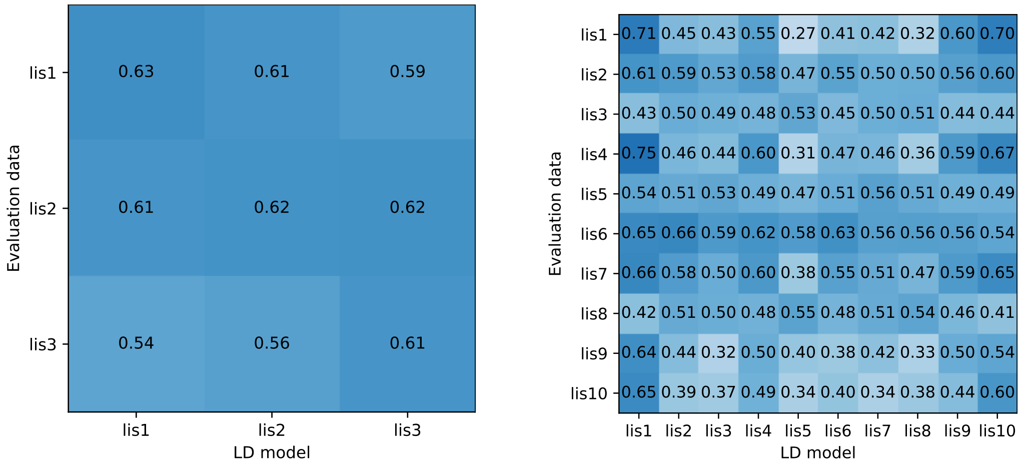


(a) *Majority-voted model*

(b) *LD model*

Figure 4.7: *Confusion matrices for MSP-Podcast.*

MSP-Podcast ($p < .05$ in paired t-test), while not in IEMOCAP. It is considered that IEMOCAP has only 3 listeners, which is too few samples for a paired t-test. Note that there are no significances among the three proposed LD models. The matrices of listener-wise WA with LD models are also shown in Fig. 4.8. All the LD models were constructed by fine-tuning. Comparing the matrices to Fig. 4.5, the evaluations of high similarity listener pairs tend to show relatively high WAs. For example, LD models of listeners 1, 4, 9, 10 showed higher WAs than the remaining LD models for listener 1 evaluation data. These results indicate that LD models can accurately learn listener-dependent emotion perception characteristics. Note that there are several listeners in which the listener-mismatched LD model showed better WAs than the listener-matched model. One possible reason is the difference in the amount of training data in listeners. For example, listener 1 has several times of training data compared with other listeners, which yields a better emotion perception model in spite of listener-mismatched conditions.



(a) *IEMOCAP* (b) *MSP-Podcast*

Figure 4.8: *WAs of listener-wise emotion recognitions with LD models.*

4.4.3 Adaptation Evaluation

Setups

We resegmented MSP-Podcast evaluation subsets to create an utterance and listener open dataset. First, the utterances contained only "rest listeners" were selected from the original training, validation, and testing dataset as the open data candidates. Second, the listeners who annotated more than 2 utterances with each target emotion and 30 utterances in total of the candidates were selected as the open listeners. Finally, the utterances that had one or more open listeners in the candidates were regarded as the open dataset, while the remaining candidates were returned to the original training, validation, and test sets. We selected 24 listeners with 1080 utterances for the open dataset. The average number of utterances per listener was 42.8. Note that we did not use IEMOCAP in the adaptation evaluation because no open utterances were available.

The baseline method was the majority-voted emotion recognition model without adaptation. It was trained by the resegmented training and validation set. The proposed was the auxiliary input-based LD model with adaptation. The LD model was trained by the resegmented training and validation set first, then adapted to the specific listener in the open set with adaptation data. 5-fold cross-validation was used in the LD model adaptation; 80 % of the open dataset was used for adaptation and the rest 20 % was for the evaluation. To evaluate the performance of the listener code estimation alone, we also ran a comparison with the auxiliary input-based LD model in the oracle condition in which the one-hot listener code that showed the highest geometric mean of WA and UA was selected for each open listeners. We used the same LD model in adaptation and oracle conditions. For the adaptation, the minibatch size was the same as the amount of listener-wise utterances in the adaptation set. The learning rate was 0.05. Earlystopping was not used and the adaptation was stopped at 30 epochs.

Evaluation metrics were macro averages of the listener-wise WAs and UAs. Note that we did not evaluate the performance of the majority-voted emotion recognition because the majority-voted emotions were not *open*; the listeners of the utterances in the open dataset were almost "rest listeners" who included in the training subset and the majority-voted emotions were mostly determined by them.

Results

The macro average of listener-wise WAs and UAs are shown in Table 4.7. Relative to the baseline, the auxiliary input-based LD model with adaptation achieved significantly better WA ($p < .05$ in paired t-test) with the same level of UA ($p > .05$). Furthermore, the oracle of the auxiliary model showed very high WA and UA ($p < .05$ compared with the auxiliary model with adaptation). These indicate that the auxiliary model is capable of listener-dependent emotion recognition and the proposed adaptation is

Table 4.7: *Macro average of WAs and UAs in listener-open dataset.*

			MSP-Podcast	
			WA	UA
Baseline	Majority		41.4	42.0
Proposed	Auxiliary	Adapted	48.4	44.2
		Oracle	58.6	52.7

effective for unseen listeners, while there is room for improvement to estimate better listener code from a limited adaptation set. The same trend is present in the examples of the listener-wise WAs and UAs shown in Fig. 4.9. The LD model with adaptation showed the same or better performances than the majority model for all listeners, and the auxiliary model in the oracle setup greatly outperformed the adapted model for some listeners such as listener B.

Table 4.6 and 4.7 show that the auxiliary model with oracle evaluation in the open set attained higher accuracies than those with listener-closed training in the test set. One possibility is that there are some listeners who gave noisy annotations, which degrades estimation performance even in listener-closed conditions. It has been reported that there are several noisy annotators in crowdsourced data like MSP-Podcast [109].

4.5 Summary

This chapter proposed an emotion recognition framework based on listener-dependent emotion perception models. The conventional approach ignores the individuality of emotional perception. The key idea of the proposal lies in constructing LD models that account for individuality. Three LD models were introduced: fine-tuning, auxiliary input, and sub-layer weighting. The last two models can adapt to a wide range

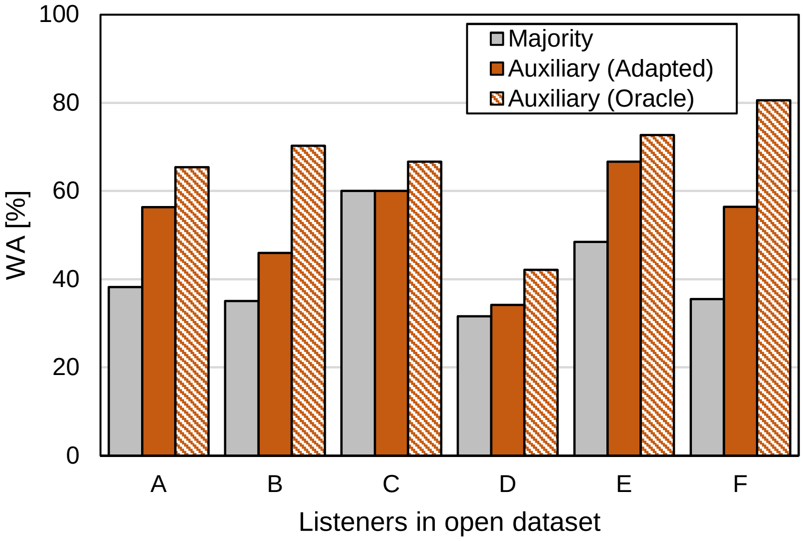


Figure 4.9: *WA for each listeners in open dataset.*

of listeners with limited model parameters. Experiments on two large emotion speech corpora revealed that emotion perception depends on listeners and that the proposed framework outperformed the conventional method by means of leveraging listener dependencies in majority-voted emotion recognition. Furthermore, the proposed LD models attained higher accuracies in listener-wise emotion recognition, which indicates that the LD models were successful in learning the individuality of emotion perception.

Future work includes investigating the effectiveness of the proposed approach in other languages and cultures, improving the adaptation framework to unseen listeners, and combining the LD models with the speaker adaptation frameworks.

5 Conclusions

5.1 Summary of This Thesis

Speech emotion recognition is an important technology to realize natural communication between humans and machines. However, research on speech emotion recognition is scarce, and there are few practical examples compared to other speech processing. This thesis aimed to develop speech emotion recognition frameworks that can be applied to the real world.

In this thesis, we attempted to solve the following two fundamental problems in speech emotion recognition. The first was that emotional expressions are very complex and diverse that their recognition is difficult. The second was that emotions are influenced not only by the speaker but also by the listener, and thus differences in the listener's perception of emotions need to be taken into account. We took two steps to solve each issue. The first step was to recognize only a limited number of emotions in a limited sound environment; customer satisfaction estimation in contact center calls. The second step was to recognize several basic emotions in natural speech; basic emotion classification in natural speech.

In Chapter 2, general information of emotions and the tasks of speech emotion recognition were described. We also presented the conventional methods of two-step studies, customer satisfaction estimation and basic emotion classification.

In Chapter 3, we described the first task: customer satisfaction estimation in contact centers. In order to capture a variety of emotional expressions to evaluate customer satisfactions, the proposed method utilized the characteristics of customer satisfaction expressions. We discovered that there was a contextual dependency in customer satisfaction, and there was a relationship between the satisfaction of the entire call and individual turns. The proposed hierarchical multi-task model had hierarchically stacked RNN layers with multi-task loss to learn these characteristics. Three types of heuristic features were also developed to capture complex cues of customer satisfactions. Furthermore, we proposed a new domain adaptation of the proposed model from call-level satisfaction labels alone, which substantially decreases annotation cost. The evaluation experiments based on two call datasets showed that the proposed model improved the estimation accuracy of two-level customer satisfactions, and the proposed adaptation achieved high estimation accuracies.

In Chapter 4, basic emotion recognition for natural speech was presented as the second step of this thesis. We proposed a new emotion recognition technique that modeled the emotion perception criteria of each listener, developing the conventional deep learning-based emotion recognition. Three types of the proposed listener-dependent emotion recognition models were described, one was the simple fine-tuning-based model and the rest two of which used auxiliary features to represent different emotion perception criteria for each listener in a single model. These proposed methods were based on domain adaptation techniques that have been successfully applied in speech processing. Evaluation experiments using two datasets revealed that the emotion perception criteria tend to be different for different listeners in natural speech. Furthermore, the proposed listener-dependent models improved estimation accuracies of both the listener-dependent and the majority-voted emotions.

In conclusion, this thesis demonstrated several speech emotion recognition methods that can be applied to the real world. Two emotion recognition frameworks for customer satisfaction estimation and basic emotion classification were proposed. The most important contribution of this thesis was that we revealed that there are certain trends in the expression and perception of emotional information, and that emotion recognition performance can be improved to utilize these trends.

5.2 Future Work

Although the proposed method has contributed to the improvement of recognition accuracy in existing speech emotion recognition tasks, there are several challenges that should be addressed in the future.

5.2.1 Utilization of Unlabeled Data

A limited amount of labeled data is usually used to train emotion recognition model because annotation of emotion labels requires a lot of cost. In order to construct a robust recognition model under such conditions, it is desirable to use unlabeled speech that is easy to collect. For example, self training and unsupervised pre-training should be incorporated.

5.2.2 Combination of Multimodal Features

Emotions are expressed not only in acoustic aspects but also in various aspects such as linguistic information. However, we do not use linguistic features, or we use only a limited vocabulary in this thesis. It is important to use multi-modal features including linguistic ones to improve the base accuracy of emotion recognition.

5.2.3 Simultaneous Modeling of Speaker and Listener Dependencies

In the second step of this paper, we proposed a method to deal with the dependency of emotion perception of listeners. However, as mentioned in Chapter 4, emotion is also dependent on the expression of speakers. A framework that handles the dependency of both the speaker and the listener in the single model is required.

5.2.4 Language Dependencies

Japanese calls and English speech are used in the first and the second step of the thesis. It is desirable to evaluate the proposed method in more languages because some studies indicate that there is a possibility that the emotion expressions depend on language.

Acknowledgments

本研究を遂行するにあたり，多くの方々にお世話になりました．この場を借りて，感謝の意を述べさせて頂きます．

主指導教員である名古屋大学大学院 情報基盤センター教授 戸田智基先生には，ひとかたならぬご指導を頂きました．社会人学生として快く研究室に受け入れて頂き，さらに研究室に通うことが困難である点を汲んで頂きリモートミーティングをご用意頂くなど，格別のご配慮を頂きました．また，細部にわたる技術・論文のご指導やご討論，数多くのあたたかい激励を頂きました．心から感謝の意を表します．

名古屋大学大学院 情報学研究科教授 武田一哉先生には学部学生から博士課程まで永年にわたり多くのご指導を賜りました．また，本研究の副査として多くのご助言を賜りました．具体的な研究指導だけでなく，研究の意義と楽しさを教えてくださり，私が研究開発の道に進むきっかけを与えてくださったことに深く感謝いたします．

名古屋大学大学院 情報学研究科 准教授 笹野遼平先生には副査として有意義なご助言とご指導を賜りました．厚く御礼申し上げます．

名古屋大学大学院 情報学研究科准教授 竹内栄二朗先生，同准教授 藤井慶輔先生，同特任准教授 石黒祥生先生を始めとした武田研究室の教官には本研究において多くのご助言とご指導を賜りました．また，戸田研究室並びに武田研究室の学生及びスタッフ各位には研究遂行にあたり有益なご討論と多大なサポートを頂きました．ここに感謝の意を表します．

日本電信電話株式会社 NTT メディアインテリジェンス研究所 (現 人間情報研究所)

青野裕司氏，同 宮崎昇氏には日頃より業務活動と博士課程活動の両立について格別のご配慮とご理解を賜りました．ここに深く感謝いたします．また，同 特別研究員 井島勇祐氏，同 (現 コンピュータ&データサイエンス研究所) 増村亮氏をはじめ，NTT メディアインテリジェンス研究所の研究員各位には日頃より有益なご討論ご助言を頂きました．ここに感謝の意を表します.

最後に，これまで私をあたたかく応援してくれた両親，私を励まし続け家庭を支えてくれた妻 莉絵，日々の笑顔で私に活力をくれた息子 聡志に心から感謝します.

References

[1] Y. Park and S. C. Gates, "Towards real-time measurement of customer satisfaction using automatically generated call transcripts," in *Proc. of ACM*, 2009, pp. 1387–1396.

[2] A. Tawari and M. Trivedi, "Speech based emotion classification framework for driver assistance system," in *Proc. of IEEE IVS*, 2010, pp. 174–178.

[3] K. hao Chang, D. Fisher, and J. Canny, "AMMON: A speech analysis library for analyzing affect, stress, and mental health on mobile phones," in *Proc. of PhoneSense*, 2011.

[4] J. C. Acosta, "Using emotion to gain rapport in a spoken dialog system," in *Proc. of NAACL HLT Student Research Workshop and Doctoral Consorium*, 2009, pp. 49–54.

[5] V. Kowtha, V. Mitra, C. Bartels, E. Marchi, S. Booker, W. Caruso, S. Kajarekar, and D. Naik, "Detecting emotion primitives from speech and their use in discerning categorical emotions," in *Proc. of ICASSP*, 2020, pp. 7164–7168.

[6] S. Parthasarathy and C. Busso, "Jointly predicting arousal, valence and dominance with multi-task learning," in *Proc. of INTERSPEECH*, 2017, pp. 1103–1107.

[7] M. Sarma, P. Ghahremani, D. Povey, N. K. Goel, K. K. Sarma, and N. Dehak, “Emotion identification from raw speech signals using DNNs,” in *Proc. of INTERSPEECH*, 2018, pp. 3097–3101.

[8] B. Schuller, A. Batliner, S. Steidl, and D. Seppi, “Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge,” *Speech Communication*, vol. 53, pp. 1062–1087, 2011.

[9] P. Li, Y. Song, I. McLoughlin, W. Guo, and L. Dai, “An attention pooling based representation learning method for speech emotion recognition,” in *Proc. of INTERSPEECH*, 2018, pp. 3087–3091.

[10] M. Neumann and N. T. Vu, “Attentive convolutional neural network based speech emotion recognition: A study on the impact of input features, signal length, and acted speech,” in *Proc. of INTERSPEECH*, 2017, pp. 1263–1267.

[11] B. Schuller, G. Rigoll, and M. Lang, “Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture,” in *Proc. of ICASSP*, 2004, pp. 577–580.

[12] S. A. Chowdhury, E. A. Stepanov, and G. Riccardi, “Predicting user satisfaction from turn-taking in spoken conversations,” in *Proc. of INTERSPEECH*, 2016.

[13] P. Tzirakis, J. Zhang, and B. Schuller, “End-to-end speech emotion recognition using deep neural networks,” in *Proc. of ICASSP*, 2018, pp. 5089–5093.

[14] H. Gunes, B. S. Schuller, M. Pantic, and R. Cowie, “Emotion representation, analysis and synthesis in continuous space: A survey,” in *Proc. of Automatic Face and Gesture Recognition*, 2011, pp. 827–834.

[15] M. El Ayadi, M. S. Kamel, and F. Karray, “Survey on speech emotion recognition: Features, classification schemes, and databases,” *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[16] P. Ekman, “An argument for basic emotions,” *Cognition and Emotion*, vol. 6, no. 3–4, pp. 169–200, 1992.

[17] R. Plutchik, “The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice,” *American Scientist*, vol. 89, no. 4, pp. 344–350, 2001.

[18] J. A. Russell, “A circumplex model of affect,” *Journal of Personality and Social Psychology*, vol. 39, no. 6, pp. 1161–1178, 1980.

[19] A. Mehrabian, *Basic Dimensions for a General Psychological Theory Implications for Personality, Social, Environmental, and Developmental Studies*, 1980.

[20] H. Fujisaki, “Prosody, information, and modeling—with emphasis on tonal features of speech,” in *Proc. of Speech Prosody*, 2004, pp. 1–10.

[21] D. Erickson, “Expressive speech: Production, perception and application to speech synthesis,” *Acoustical Science and Technology*, vol. 26, no. 4, pp. 317–325, 2005.

[22] N. Campbell, “Developments in corpus-based speech synthesis: Approaching natural conversational speech,” *IEICE TRANSACTIONS on Information and Systems*, vol. E88-D, no. 3, pp. 376–383, 2005.

[23] A. Satt, S. Rozenberg, and R. Hoory, “Efficient emotion recognition from speech using deep learning on spectrograms,” in *Proc. of INTERSPEECH*, 2017, pp. 1089–1093.

[24] A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, “Speech emotion recognition based on multi-label emotion existence model,” in *Proc. of INTERSPEECH*, 2019, pp. 2818–2822.

[25] J. Ang, R. Dhillon, A. Krupski, E. Shriberg, and A. Stolcke, “Prosody-based automatic detection of annoyance and frustration in human-computer dialog,” *Proc. of ICSLP*, 2002.

[26] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M. A. Nicolaou, B. Schuller, and S. Zafeiriou, “Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network,” in *Proc. of ICASSP*, 2016, pp. 5200–5204.

[27] M. Schmitt, N. Cummins, and B. W. Schuller, “Continuous Emotion Recognition in Speech — Do We Need Recurrence?” in *Proc. of INTERSPEECH*, 2019, pp. 2808–2812.

[28] K. R. Scherer, “Vocal communication of emotion: A review of research paradigms,” *Speech Communication*, vol. 40, no. 1, pp. 227–256, 2003.

[29] H. Mori, K. Maekawa, and H. Kasuya, *What does speech convey? : speech science of emotion, paralinguistic information, and speaker individuality.* CORONA PUBLISHING, 2014.

[30] M. Gamon, “Sentiment classification on customer feedback data: Noisy data, large feature vectors, and the role of linguistic analysis,” in *Proc. of COLING*, 2004.

[31] P. Gupta and N. Rajput, “Two-stream emotion recognition for call center monitoring,” in *Proc. of INTERSPEECH*, 2007, pp. 2241–2244.

[32] S. Godbole and S. Roy, “Text classification, business intelligence, and interactivity: Automating c-sat analysis for services industry,” in *Proc. of KDDM*, 2008, pp. 911–919.

[33] E. A. Hassan, N. E. Gayer, and M. M. Ghanem, “Emotions analysis of speech for call classification,” in *Proc. of ISDA*, 2010, pp. 242–247.

[34] Q. Llimona, J. Luque, X. Anguera, Z. Hidalgo, S. Park, and N. Oliver, “Effect of gender and call duration on customer satisfaction in call center big data,” in *Proc. of INTERSPEECH*, 2015.

[35] R. Chakraborty, M. Pandharipande, and S. Kopparapu, “Event based emotion recognition for realistic non-acted speech,” in *Proc. of TENCON*, 2015, pp. 1–5.

[36] J. Sun, W. Xu, Y. Yan, C. Wang, Z. Ren, P. Cong, H. Wang, and J. Feng, “Information fusion in automatic user satisfaction analysis in call center,” in *Proc. of IHMSC*, 2016, pp. 425–428.

[37] C. Segura, D. Balcells, M. Umbert, J. Arias, and J. Luque, “Automatic speech feature learning for continuous prediction of customer satisfaction in contact center phone calls,” in *Proc. of IberSPEECH*, 2016, pp. 255–265.

[38] P. Cong, C. Wang, Z. Ren, H. Wang, Y. Wang, and J. Feng, “Unsatisfied customer call detection with deep learning,” in *Proc. of ISCSLP*, 2016, pp. 1–5.

[39] J. Bockhorst, S. Yu, L. Polania, and G. Fung, “Predicting self-reported customer satisfaction of interactions with a corporate call center,” in *Proc. of ECML PKDD*, 2017, pp. 179–190.

[40] J. Luque, C. Segura, A. Sánchez, M. Umbert, and L. A. Galindo, "The role of linguistic and prosodic cues on the prediction of self-reported satisfaction in contact centre phone calls," in *Proc. of INTERSPEECH*, 2017, pp. 2346–2350.

[41] L. Devillers and I. Vasilescu, "Reliability of lexical and prosodic cues in two real-life spoken dialog corpora," in *Proc. of LREC*, 2004, pp. 1423–1426.

[42] L. Vidrascu and L. Devillers, "Five emotion classes detection in real-world call center data: the use of various types of paralinguistic features," in *Proc. of PARALING*, 2007, pp. 11–16.

[43] D. Morrison, R. Wang, and L. C. D. Silva, "Ensemble methods for spoken emotion recognition in call-centres," *Speech Communication*, vol. 49, pp. 98–112, 2007.

[44] L. Devillers, C. Vaudable, and C. Chastagnol, "Real-life emotion-related states detection in call centers: a cross-corpora study." in *Proc. of INTERSPEECH*, 2010, pp. 2350–2353.

[45] T. Polzehl, A. Schmitt, F. Metze, and M. Wagner, "Anger recognition in speech using acoustic and linguistic cues," *Speech Communication*, vol. 53, pp. 1198–1209, 2011.

[46] N. Nomoto, M. Tamoto, H. Masataki, O. Yoshioka, and S. Kobashikawa, "Anger recognition in spoken dialog using linguistic and para-linguistic information," in *Proc. of INTERSPEECH*, 2011, pp. 1545–1548.

[47] M. Erden and L. M. Arslan, "Automatic detection of anger in human-human call center dialogs," in *Proc. of INTERSPEECH*, 2011, pp. 81–84.

[48] C. Vaudable and L. Devillers, "Negative emotions detection as an indicator of dialogs quality in call centers," in *Proc. of ICASSP*, 2012, pp. 5109–5112.

[49] D. Galanis, S. Karabetsos, M. Koutsombogera, H. Papageorgiou, A. Esposito, and M.-T. Riviello, "Classification of emotional speech units in call centre interactions," in *Proc. of CogInfoCom*, 2013, pp. 403–406.

[50] R. Chakraborty, M. Pandharipande, and S. K. Kopparapu, "Mining call center conversations exhibiting similar affective states," in *Proc. of PACLIC*, 2016, pp. 545–553.

[51] K. P. Seng and L.-M. Ang, "Video analytics for customer emotion and satisfaction at contact centers," *IEEE Trans. Human-Machine Systems*, vol. 48, no. 3, pp. 266–278, 2018.

[52] I. Luengo, E. Navas, I. Hernàez, and J. Sànchez, "Automatic emotion recognition using prosodic parameters," in *Proc. of INTERSPEECH*, 2005, pp. 493–496.

[53] Y. Li, T. Zhao, and T. Kawahara, "Improved end-to-end speech emotion recognition using self attention mechanism and multitask learning," in *Proc. of INTERSPEECH*, 2019, pp. 2803–2807.

[54] A. Ando, S. Kobashikawa, H. Kamiyama, R. Masumura, Y. Ijima, and Y. Aono, "Soft-target training with ambiguous emotional utterances for dnn-based speech emotion classification," in *Proc. of ICASSP*, 2018, pp. 4964–4968.

[55] H. Lane, H. Hapke, and C. Howard, *Natural Language Processing in Action: Understanding, analyzing, and generating text with Python.* Manning Publications, 2019.

[56] Z. Lu, L. Cao, Y. Zhang, C. Chiu, and J. Fan, "Speech sentiment analysis via pre-trained features from end-to-end asr models," in *Proc. of ICASSP*, 2020, pp. 7149–7153.

[57] L. Pepino, P. Riera, L. Ferrer, and A. Gravano, "Fusion approaches for emotion recognition from speech usinh acoustic and text-based features," in *Proc. of ICASSP*, 2020, pp. 6484–6488.

[58] H. Ranganathan, S. Chakraborty, and S. Panchanathan, "Multimodal emotion recognition using deep learning architectures," in *Proc. of WACV*, 2016, pp. 1–9.

[59] Y. Gan, J. Chen, and L. Xu, "Facial expression recognition boosted by soft label with a diverse ensemble," *Pattern Recognition Letters*, vol. 125, pp. 105–112, 2019.

[60] R. Zhang, A. Ando, S. Kobashikawa, and Y. Aono, "Interaction and transition model for speech emotion recognition in dialogue," in *Proc. of INTERSPEECH*, 2017, pp. 1094–1097.

[61] S.-L. Yeh, Y.-S. Lin, and C.-C. Lee, "An interaction-aware attention network for speech emotion recognition in spoken dialogs," *Proc. of ICASSP*, pp. 6685–6689, 2019.

[62] A. Batliner and R. Huber, "Speaker characteristics and emotion classification," *Springer*, vol. 4343, pp. 138–151, 2007.

[63] V. Sethu, E. Ambikairajah, and J. Epps, "Phonetic and speaker variations in automatic emotion classification," in *Proc. of INTERSPEECH*, 2008, pp. 617–620.

[64] V. Sethu, J. Epps, and E. Ambikairajah, "Speaker variability in speech based emotion models - analysis and normalisation," in *Proc. of ICASSP*, 2013, pp. 7522–7526.

[65] J. Kim, J.-S. Park, and Y.-H. Oh, "Speaker-characterized emotion recognition using online and iterative speaker adaptation," *Cognitive Computation*, vol. 4, pp. 398–408, 2012.

[66] A. Nediyanchath, P. Paramasivam, and P. Yenigalla, "Multi-head attention for speech emotion recognition with auxiliary learning," in *Proc. of ICASSP*, 2020, pp. 7179–7183.

[67] J.-L. Li and C.-C. Lee, "Attentive to individual: A multimodal emotion recognition network with personalized attention profile," in *Proc. of INTERSPEECH*, 2019, pp. 211–215.

[68] B. M. Ben-David, S. Gal-Rosenblum, P. H. H. M. van Lieshout, and V. Shakuf, "Age-related differences in the perception of emotion in spoken language: The relative roles of prosody and semantics," *Journal of speech, language, and hearing research*, vol. 62, pp. 1188–1202, 2019.

[69] Y. Zhao, A. Ando, S. Takaki, J. Yamagishi, and S. Kobashikawa, "Does the lombard effect improve emotional communication in noise? - analysis of emotional speech acted in noise -," in *Proc. of INTERSPEECH*, 2019, pp. 3292–3296.

[70] J. Dang, A. Li, D. Erickson, A. Suemitsu, M. Akagi, K. Sakuraba, N. Minematsu, and K. Hirose, "Comparison of emotion perception among different cultures," *Acoustical Science and Technology*, vol. 31, no. 6, pp. 394–402, 2010.

[71] H. M. Fayek, M. Lech, and L. Cavedon, "Modeling subjectiveness in emotion recognition with deep neural networks: Ensembles vs soft labels," in *Proc. of IJCNN*, 2016, pp. 566–570.

[72] Y. Chen, J. Wang, Y. Yang, and H. H. Chen, “Component tying for mixture model adaptation in personalization of music emotion recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 7, pp. 1409–1420, 2017.

[73] J. Wang, Y. Yang, H. Wang, and S. Jeng, “Personalized music emotion recognition via model adaptation,” in *Proc. of APSIPA*, 2012, pp. 1–7.

[74] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, “Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition,” in *Proc. of ICASSP*, 2013, pp. 7893–7897.

[75] T. Ochiai, S. Matsuda, X. Lu, C. Hori, and S. Katagiri, “Speaker adaptive training using deep neural networks,” in *Proc. of ICASSP*, 2014, pp. 6349–6353.

[76] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, “Robust i-vector based adaptation of dnn acoustic model for speech recognition,” in *Proc. of INTERSPEECH*, 2015, pp. 2877–2881.

[77] N. Hojo, Y. Ijima, and H. Mizuno, “Dnn-based speech synthesis using speaker codes,” *IEICE Transactions on Information and Systems*, vol. E101.D, no. 2, pp. 462–472, 2018.

[78] M. Delcroix, K. Zmolikova, K. Kinoshita, A. Ogawa, and T. Nakatani, “Single channel target speaker extraction and recognition with speaker beam,” in *Proc. of ICASSP*, 2018, pp. 5554–5558.

[79] G. Forman, E. Kirshenbaum, and J. Suermondt, “Pragmatic text mining: Minimizing human effort to quantify many issues in call logs,” in *Proc. of KDDM*, 2006, pp. 852–861.

[80] I. Bhattacharya, S. Godbole, A. Gupta, A. Verma, J. Achtermann, and K. English, "Enabling analysts in managed services for crm analytics," in *Proc. of KDDM*, 2009, pp. 1077–1086.

[81] R. Hallowell, "The relationships of customer satisfaction, customer loyalty, and profitability: an empirical study," *International journal of service industry management*, vol. 7, no. 4, pp. 27–42, 1996.

[82] K. Hashimoto, C. Xiong, Y. Tsuruoka, and R. Socher, "A joint many-task model: Growing a neural network for multiple NLP tasks," in *Proc. of EMNLP*, 2017, pp. 1923–1933.

[83] A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, "Hierarchical lstms with joint learning for estimating customer satisfaction from contact center calls," in *Proc. of INTERSPEECH*, 2017, pp. 1716–1720.

[84] M. Fujimoto and K. Ishizuka, "Noise robust voice activity detection based on switching kalman filter," *IEICE Trans. information & systems*, vol. 91, no. 3, pp. 467–477, 2008.

[85] H. Koiso, Y. Horiuchi, S. Tutiya, A. Ichikawa, and Y. Den, "An analysis of turn-taking and backchannels based on prosodic and syntactic features in japanese map task dialogs," *Language and speech*, vol. 41, no. 3–4, pp. 295–321, 1998.

[86] J. Auguste, D. Charlet, G. Damnati, F. Bechet, and B. Favre, "Can we predict self-reported customer satisfaction from interactions?" in *Proc. of ICASSP*, 2019, pp. 7385–7389.

[87] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of ICLR*, 2015.

[88] L. Prechelt, "Automatic early stopping using cross validation: quantifying the criteria," *Neural Networks*, vol. 11, pp. 761–767, 1998.

[89] T. Nakatani and T. Irino, "Robust and accurate fundamental frequency estimation based on dominant harmonic components," *The Journal of the Acoustical Society of America*, vol. 116, no. 6, pp. 3690–3700, 2004.

[90] H. Masataki, D. Shibata, Y. Nakazawa, S. Kobashikawa, A. Ogawa, and K. Ohtsuki, "VoiceRex – spontaneous speech recognition technology for contact-center conversation," *NTT Technical Review*, vol. 5, no. 1, pp. 22–27, 2007.

[91] T. Hori, C. Hori, Y. Minami, and A. Nakamura, "Efficient WFST-based one-pass decoding with on-the-fly hypothesis rescoring in extremely large vocabulary continuous speech recognition," *IEEE Trans. Audio, Speech & Language Processing*, vol. 15, no. 4, pp. 1352–1365, 2007.

[92] J. S. Akosa, "Predictive accuracy: A misleading performance measure for highly imbalanced data," in *Proc. of SAS Global Forum*, 2017.

[93] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Trans. Intelligent Systems & Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[94] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *Advances in NIPS*, 2017.

[95] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. of IJCAI*, 2001, pp. 973–978.

[96] M. Buda, A. Maki, and M. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, 2017.

[97] A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, Y. Aono, and T. Toda, “Customer satisfaction estimation in contact center calls based on a hierarchical multi-task model,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 715–728, 2020.

[98] K. Han, D. Yu, and I. Tashev, “Speech emotion recognition using deep neural network and extreme learning machine,” in *Proc. of INTERSPEECH*, 2014, pp. 223–227.

[99] J. Lee and I. Tashev, “High-level feature representation using recurrent neural network for speech emotion recognition,” in *Proc. of INTERSPEECH*, 2015.

[100] S. Mirsamadi, E. Barsoum, and C. Zhang, “Automatic speech emotion recognition using recurrent neural networks with local attention,” in *Proc. of ICASSP*, 2017, pp. 2227–2231.

[101] C. W. Huang and S. S. Narayanan, “Attention assisted discovery of sub-utterance structure in speech emotion recognition,” in *Proc. of INTERSPEECH*, 2016, pp. 1387–1391.

[102] Z. Lin, M. Feng, C. N. dos Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, “A structured self-attentive sentence embedding,” in *Proc. of ICLR*, 2017.

[103] O. Abdel-Hamid and H. Jiang, “Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code,” in *Proc. of ICASSP*, 2013, pp. 7942–7946.

[104] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. Chang, S. Lee, and S. S. Narayanan, "IEMOCAP: Interactive emotional dyadic motion capture database," *Journal of Language Resources and Evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[105] R. Lotfian and C. Busso, "Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings," *IEEE Transactions on Affective Computing*, vol. 10, no. 4, pp. 471–483, 2019.

[106] X. Ma, Z. Wu, J. Jia, M. Xu, H. Meng, and L. Cai, "Emotion recognition from variable-length speech segments using deep learning on spectrograms," in *Proc. of INTERSPEECH*, 2018, pp. 3683–3687.

[107] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. of ICLR*, 2015, pp. 448–456.

[108] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. of INTERSPEECH*, 2019, pp. 2613–2617.

[109] R. Lotfian and C. Busso, "Curriculum learning for speech emotion recognition from crowdsourced labels," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 4, pp. 815–826, 2019.

List of Publications

Journal Papers

1. A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, Y. Aono, and T. Toda, "Customer Satisfaction Estimation in Contact Center Calls Based on a Hierarchical Multi-Task Model," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 715–728, 2020.

2. A. Ando, T. Mori, S. Kobashikawa, and T. Toda, "Speech emotion recognition based on listener-dependent emotion perception models," APSIPA Transactions on Signal and Information Processing, vol. 10, no. E6, 2021.

3. H. Kamiyama, A. Ando, R. Masumura, S. Kobashikawa, and Y. Aono, "Likability estimation for contact center agents by selecting annotators based on binomial distribution," Acoustical Science and Technology, vol. 41, no. 6, pp. 826–828, 2020.

International Conferences

1. A. Ando, R. Masumura, H. Sato, T. Moriya, T. Ashihara, Y. Ijima, and T. Toda, "Speech Emotion Recognition Based on Listener Adaptive Models," Proc. ICASSP, pp. 6274–6278, 2021.

2. A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, "Speech Emotion Recognition Based on Multi-Label Emotion Existence Model," Proc. INTERSPEECH, pp. 2818–2822, 2019.

3. A. Ando, S. Kobashikawa, H. Kamiyama, R. Masumura, Y. Ijima, and Y. Aono, "Soft-target training with ambiguous emotional utterances for dnn-based speech emotion classification," Proc. ICASSP, pp. 4964–4968, 2018.

4. A. Ando, R. Asakawa, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, "Automatic Question Detection from Acoustic and Phonetic Features Using Feature-wise Pre-training," Proc. INTERSPEECH, pp. 1731–1735, 2018.

5. A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, "Hierarchical LSTMs with Joint Learning for Estimating Customer Satisfaction from Contact Center Calls," Proc. INTERSPEECH, pp. 1716–1720, 2017.

6. A. Ando, T. Asami, Y. Yamaguchi, and Y. Aono, "Speaker recognition in duration-mismatched condition using bootstrapped i-vectors," Proc. APSIPA, pp. 1–4, 2016.

7. A. Ando, T. Asami, M. Okamoto, H. Masataki, and S. Sakauchi, "Agreement and disagreement utterance detection in conversational speech by extracting and integrating local features," Proc. INTERSPEECH, 2015.

8. Y. Kitagishi, H. Kamiyama, A. Ando, N. Tawara, T. Mori, and S. Kobashikawa, "Speaker Age Estimation Using Age-Dependent Insensitive Loss," Proc. APSIPA, pp. 319–324, 2020.

9. R. Masumura, M. Ihori, A. Takashima, T. Moriya, A. Ando, and Y. Shinohara, "Sequence-level consistency training for semi-supervised end-to-end automatic speech recognition," Proc. ICASSP, pp. 7054–7058, 2020.

10. H. Kamiyama, A. Ando, R. Masumura, S. Kobashikawa, and Y. Aono, “Urgent Voicemail Detection Focused on Long-term Temporal Variation,” Proc. APSIPA, pp. 917–921, 2019.

11. H. Kamiyama, A. Ando, R. Masumura, S. Kobashikawa, and Y. Aono, “Likability Estimation of Call-center Agents by Suppressing Annotator Variability,” Proc. APSIPA, pp. 911–916, 2019.

12. R. Masumura, M. Ihori, T. Tanaka, A. Ando, R. Ishii, T. Oba, and R. Higashinaka, “Improving speech-based end-of-turn detection via cross-modal representation learning with punctuated text data,” Proc. ASRU, pp. 1062–1069, 2019.

13. R. Masumura, T. Tanaka, A. Ando, H. Kamiyama, T. Oba, S. Kobashikawa, and Y. Aono, “Improving Conversation-Context Language Models with Multiple Spoken Language Understanding Models,” Proc. INTERSPEECH, pp. 834–838, 2019.

14. Y. Zhao, A. Ando, S. Takaki, J. Yamagishi, S. Kobashikawa, “Does the Lombard Effect Improve Emotional Communication in Noise? — Analysis of Emotional Speech Acted in Noise,” Proc. INTERSPEECH, pp. 3292–3296, 2019.

15. R. Masumura, S. Yamada, T. Tanaka, A. Ando, H. Kamiyama, and Y. Aono, “Online call scene segmentation of contact center dialogues based on role aware hierarchical LSTM-RNNs,” Proc. APSIPA, pp. 811–815, 2018.

16. R. Masumura, T. Tanaka, A. Ando, H. Masataki, Y. Aono, “Role Play Dialogue Aware Language Models Based on Conditional Hierarchical Recurrent Encoder-Decoder,” Proc. INTERSPEECH, pp. 1259–1263, 2018.

17. R. Masumura, T. Tanaka, A. Ando, R. Ishii, R. Higashinaka, and Y. Aono, "Neural dialogue context online end-of-turn detection," Proc. Annual SIGdial Meeting on Discourse and Dialogue, pp. 224–228, 2018.

18. H. Kamiyama, A. Ando, S. Kobashikawa, and Y. Aono, "Robust children and adults speech identification and confidence measure based on DNN posteriorgram," Proc. APSIPA, pp. 502–505, 2017.

19. R. Zhang, A. Ando, S. Kobashikawa, and Y. Aono, "Interaction and Transition Model for Speech Emotion Recognition in Dialogue," Proc. INTERSPEECH, pp. 1094–1097, 2017.

Domestic Conferences

1. 安藤 厚志, 森 岳至, 小橋川 哲, 戸田 智基, "聴取者ごとの感情知覚モデルに基づく音声感情認識," 日本音響学会講演論文集 (秋), 3-2-1, pp. 777–778, 2020.

2. 北岸 佑樹, 神山 歩相名, 安藤 厚志, 俵 直弘, 森 岳至, 小橋川 哲, "話者性別推定とのマルチタスク学習による話者年齢推定," 日本音響学会講演論文集 (秋), 3-T2-8, pp. 909–910, 2020.

3. 岡田 慎太郎, 安藤 厚志, 戸田 智基, "発話感情認識における音韻・話者情報の低減," 日本音響学会講演論文集 (春), 1-4-3, pp. 873–874, 2020.

4. 増村 亮, 庵 愛, 高島 瑛彦, 森谷 崇史, 安藤 厚志, 篠原 雄介, "半教師あり End-to-End 音声認識のための系列単位 Consistency Training の検討," 日本音響学会講演論文集 (春), 2-4-3, pp. 889–890, 2020.

5. 神山 歩相名, 安藤 厚志, 増村 亮, 小橋川 哲, 青野 裕司, “話速の変動を捉える特徴量に基づく留守録音声の緊急度推定,” 日本音響学会講演論文集 (秋), 1-3-4, pp. 1185–1186, 2019.

6. 安藤 厚志, 増村 亮, 神山 歩相名, 小橋川 哲, 青野 裕司, 戸田 智基, “コンタクトセンタ顧客満足度推定におけるドメイン適応の検討,” 日本音響学会講演論文集 (秋), 2-Q-3, pp. 885–886, 2019.

7. 神山 歩相名, 安藤 厚志, 増村 亮, 小橋川 哲, 青野 裕司, “ラベラーの安定性を考慮した潜在変数モデルに基づく電話応対の好感度推定,” 日本音響学会講演論文集 (春), 1-9-13, pp. 1353–1356, 2019.

8. 安藤 厚志, 神山 歩相名, 小橋川 哲, 青野 裕司, “逆教師学習に基づく音声感情分類,” 日本音響学会講演論文集 (春), 1-9-14, pp. 1357–1358, 2019.

9. 岡田 慎太郎, 安藤 厚志, 戸田 智基, “音素事後確率を利用した表現学習に基づく発話感情認識,” 日本音響学会講演論文集 (春), 2-9-7, pp. 881–882, 2019.

10. 増村 亮, 田中 智大, 安藤 厚志, 石井 亮, 東中 竜一郎, 青野 裕司, “対話コンテキストを扱うターン交替点検出の検討,” 日本音響学会講演論文集 (春), 3-9-2, pp. 889–890, 2019.

11. 安藤 厚志, 増村 亮, 神山 歩相名, 小橋川 哲, 青野 裕司, “Feature-wise Pre-trainingを用いた音声・言語特徴からの質問発話検出,” 日本音響学会講演論文集 (秋), 2-Q-5, pp. 1049–1050, 2018.

12. 神山 歩相名, 安藤 厚志, 増村 亮, 小橋川 哲, 青野 裕司, “ラベラーの安定性を考慮した電話応対者の好感度推定,” 日本音響学会講演論文集 (秋), 2-Q-6, pp. 1051–1052, 2018.

13. 安藤 厚志, 神山 歩相名, 小橋川 哲, 増村 亮, 青野 裕司, “曖昧感情発話を活用したソフトターゲット学習に基づく音声感情分類,” 日本音響学会講演論文集 (春), 2-8-5, pp. 41–42, 2018.

14. 安藤 厚志, Zhang Ruo, 小橋川 哲, 青野 裕司, “感情の自己/相互作用モデルを用いた対話音声の感情分類,” 日本音響学会講演論文集 (春), 2-8-6, pp. 43–44, 2018.

15. 神山 歩相名, 安藤 厚志, 小橋川 哲, 青野 裕司, “電話応対音声における好感度推定の検討,” 日本音響学会講演論文集 (春), 2-Q-6, pp. 149–150, 2018.

16. 安藤 厚志, 増村 亮, 神山 歩相名, 小橋川 哲, 青野 裕司, “階層マルチタスク学習を用いたコンタクトセンタ通話からの顧客満足度推定,” 日本音響学会講演論文集 (秋), 1-10-13, pp. 37–38, 2017.

17. 神山 歩相名, 安藤 厚志, 小橋川 哲, 青野 裕司, “性別年代識別における DNN 事後確率系列を用いた信頼度尺度,” 日本音響学会講演論文集 (秋), 2-Q-15, pp. 163–164, 2017.

18. 神山 歩相名, 安藤 厚志, 浅見 太一, 小橋川 哲, 山口 義和, 青野 裕司, “DNN における偏在データの影響を考慮した性別・年代識別手法,” 日本音響学会講演論文集 (春), 1-Q-17, pp. 127–128, 2017.

19. 安藤 厚志, 神山 歩相名, 小橋川 哲, 青野 裕司, “コンタクトセンタ通話における顧客満足度推定の検討,” 日本音響学会講演論文集 (春), 2-P-4, pp. 145–146, 2017.

20. 安藤 厚志, 浅見 太一, 山口 義和, 青野 裕司, “短い発話での話者識別における話者の血縁関係の影響分析,” 日本音響学会講演論文集 (秋), 2-4-5, pp. 15–16, 2016.

21. 安藤 厚志, 浅見 太一, 山口 義和, 青野 裕司, “登録発話分割を用いた短い発話に頑健な話者識別,” 日本音響学会講演論文集 (春), 1-1-6, pp. 11–12, 2016.

22. 安藤 厚志, 浅見 太一, 岡本 学, 政瀧 浩和, 阪内 澄宇, “韻律と言語の局所的特徴に基づく会議音声からの肯定/否定発話の抽出,” 日本音響学会講演論文集 (秋), 2-1-9, pp. 1323–1324, 2015.

23. 安藤 厚志, 宮島 千代美, 北岡 教英, 武田一哉, “音声認識のための特徴量領域音源分離,” 日本音響学会講演論文集 (秋), 3-9-12, 2012.

24. 安藤 厚志, 大橋 宏正, 原 直, 北岡 教英, 武田一哉, “周波数帯域ごとの音源分離信頼度を利用したマルチバンド音声認識,” 日本音響学会講演論文集 (春), 1-P-15, pp. 153–156, 2012.

25. 安藤 厚志, “[招待講演] 音声感情認識の分野動向と実用化に向けたNTTの取り組み,” 情報処理学会研究報告, 2020-SLP-133, no. 16, pp. 1-1, 2020.

26. 高木 信二, 安藤 厚志, 越智 景子, 沢田 慶, 塩田 さやか, 鈴木 雅之, 玉森 聡, 俵 直弘, 福田 隆, 増村 亮, “国際会議Interspeech2018報告,” 情報処理学会研究報告, 2019-SLP-126, no. 10, pp. 1-9, 2019.

27. 秋田 祐哉, 安藤 厚志, 岡本 拓磨, 小川 厚徳, 神田 直之, 倉田 岳人, 郡山 知樹, 篠崎 隆宏, 高島 遼一, 太刀岡 勇気, 藤本 雅清, 増村 亮, “国際会議Interspeech2018報告,” 情報処理学会研究報告, 2018-SLP-123, no. 2, pp. 1-7, 2018.

28. 藤田 健一, 安藤 厚志, 井島 勇祐, “音素継続時間長のモデル化のための発話リズムに基づく話者埋め込みの検討,” 電子情報通信学会技術研究報告, Vol. 120, IEICE-SP-399, pp. 103–108, 2021.

29. 岡田 慎太郎, 安藤 厚志, 戸田 智基, “発話感情認識における音素事後確率を利用した表現学習とデータ拡張の評価,” 電子情報通信学会技術研究報告, Vol. 119, IEICE-SP-321, pp. 91–96, 2019.

30. 安藤 厚志, 増村 亮, 神山 歩相名, 小橋川 哲, 青野 裕司, “マルチラベル感情表出推定に基づく音声感情分類,” 電子情報通信学会技術研究報告, Vol. 119, IEICE-SP-188, pp. 39–44, 2019.

31. 増村 亮, 田中 智大, 安藤 厚志, 神山 歩相名, 大庭 隆伸, 青野 裕司, “対話コンテキストを考慮したニューラル通話シーン分割,” 電子情報通信学会技術研究報告, Vol. 118, IEICE-NLC-439, pp. 21–26, 2019.

32. 増村 亮, 田中 智大, 安藤 厚志, 大庭 隆伸, 青野 裕司, “条件付き階層再帰型エンコーダデコーダに基づく複数人会話音声認識向け言語モデル,” 電子情報通信学会技術研究報告, Vol. 118, IEICE-NLC-439, pp. 21–26, 2019.

33. 神山 歩相名, 安藤 厚志, 増村 亮, 小橋川 哲, 青野 裕司, “アノテータのラベル付与能力を考慮した電話応対音声の好感度推定モデル学習法の検討,” 電子情報通信学会技術研究報告, Vol. 118, IEICE-SP-497, pp. 197–202, 2019.

34. Y. Zhao, A. Ando, S. Takaki, J. Yamagishi, S. Kobashikawa, “Initial analysis of emotional speech acted in noise,” 電子情報通信学会技術研究報告, Vol. 118, IEICE-SP-497, pp. 125–130, 2019.

35. 渡部 瑞季, 安藤 厚志, 神山 歩相名, 小橋川 哲, 青野 裕司, 大庭 隆伸, 礒田 佳徳, “対面式の窓口会話に対する話者の出現パターンに着目したダイアライゼーション,” 電子情報通信学会技術研究報告, Vol. 117, IEICE-SP-160, pp. 21–26, 2017.

36. 安藤 厚志, 丹羽 健太, 北岡 教英, 武田 一哉, “特徴量領域音源分離のためのクロススペクトル抑圧,” 電子情報通信学会技術研究報告, Vol. 112, no. 369, pp. 107–112, 2012.

37. 安藤 厚志, 大橋 宏正, 原 直, 北岡 教英, 武田 一哉, “ブラインド音源分離の信頼度を用いたマルチバンド音声認識,” 電子情報通信学会技術研究報告, Vol. 111, no. 431, pp. 219–224, 2012.

Awards

1. 2019 年 日本音響学会 粟屋潔学術奨励賞 受賞