We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

4

Learning Physically Grounded Lexicons from Spoken Utterances

Ryo Taguchi[1], Naoto Iwahashi[2], Kotaro Funakoshi[3], Mikio Nakano[3], Takashi Nose[4] and Tsuneo Nitta[5]
[1]*Nagoya Institute of Technology,*
[2]*National Institute of Information and Communications Technology,*
[3]*Honda Research Institute Japan Co., Ltd.,*
[4]*Tokyo Institute of Technology,*
[5]*Graduate School of Engineering, Toyohashi University of Technology*
Japan

1. Introduction

Service robots must understand correspondence relationships between things in the real world and words in order to communicate with humans. For example, to understand the utterance, "Bring me an apple," the robot requires knowledge about the relationship between the word "apple" and visual features of the apple, such as color and shape. Robots perceive object features with physical sensors. However, developers of service robots cannot describe all knowledge in advance because such robots may be used in situations other than those the developers assumed. In particular, household robots have many opportunities to encounter unknown objects. Therefore, it is preferable that robots automatically learn physically grounded lexicons, which consist of phoneme sequences and meanings of words, through interactions with users.

In the field of automatic speech recognition, several methods have been proposed for extracting out-of-vocabulary (OOV) words from continuous speech by using acoustic and grammatical models of OOV word classes such as personal names or place names (Asadi 1991; Schaaf, 2001; Bazzi & Glass, 2002). However, these studies have not dealt with the learning of physically grounded meanings.

Holzapfel et al. proposed a method for learning a phoneme sequence and the meaning of each word using pre-defined utterances in which unknown words are inserted, such as "my name is <*name*>", where any name can replace <*name*> (Holzapfel et al., 2008). Methods similar to Holzapfel's method have been used with many existing robots learning the names of humans or objects. However, these methods do not solve the problem of a robot's inability to learn words from undefined utterances.

Gorin et al., Alshawi, and Roy & Pentland conducted experiments to extract semantically useful phoneme sequences from natural utterances, but they have not yet been able to acquire the correct phoneme sequences with high accuracy (Gorin et al., 1999; Alshawi, 2003; Roy & Pentland, 2002). Since phoneme sequences obtained by recognizing utterances may

contain errors, it is difficult to correctly identify the word boundaries. For example, Roy and Pentalnd extracted keywords by using similarities of both acoustic features and meanings, but 70% of the extracted words contained insertion or deletion errors at either or both ends of the words. This method obtains many word candidates corresponding to each true word through learning. If robots speak words through speech synthesis, they have to select the word that has the most correct phoneme sequence from the candidates. However, this method does not have a selection mechanism because it is designed for speech recognition not for speech synthesis.

This chapter focuses on the task in which a robot learns the name of an object from a user's vocal instruction involving the use of natural expressions while showing the object to the robot. Through this learning, the robot acquires physically grounded lexicons for speech recognition and speech synthesis. User utterances for teaching may include words other than names of objects. For example, the user might say "this is James." In this paper, names of objects are called keywords, and words (or phrases) other than keywords are called non-keyword expressions. We assume that keywords and non-keyword expressions are independent of each other. Therefore, the same non-keyword expressions can be used in instruction utterances for different keywords. The robot in this task had never been given linguistic knowledge other than an acoustic model of phonemes. A robot can recognize user utterances as phoneme sequences with this model but cannot detect word boundaries. The robot must learn the correct phoneme sequences and the meanings of keywords from a set of utterance and object pairs. After learning, we estimate the learning results by investigating whether the robot can output the correct phoneme sequence corresponding to each object.

To solve this task, we propose a method for learning phoneme sequences of words and relationships between them and objects (hereafter *meanings*) from various user utterances, without any prior linguistic knowledge other than an acoustic model of phonemes. Roy and Petland's method focuses on acoustic and semantic information of each word, and ignores words other than keywords. However, we believe that insertion or deletion errors at the ends of the words can be decreased by learning and using grammatical relationships between each non-keyword expression and keywords. Therefore, we formulated the utterance-object joint probability model, which consists of three statistical models: acoustic, grammatical, and semantic. Moreover, by learning this model on the basis of the minimum description length principle (Rissanen, 1983), acoustically, grammatically, and semantically appropriate phoneme sequences can be acquired as words.

We describe the utterance-object joint probability model in Section 2 and explain how to learn and use the model in Section 3. We show and discuss the experimental results in Section 4 and conclude the paper in Section 5.

2. Utterance-object joint probability model

2.1 Joint probability model

The joint probability model of a spoken utterance and an object is formulated as follows.

Learning sample set **D** is defined in Eq. 1.

$$\mathbf{D} = \{ \mathbf{d}_i \mid 1 \le i \le M \}, \tag{1}$$

where $\mathbf{d}_i$ is the i-th learning sample and M is the number of samples. Each sample consists of a spoken utterance and an object, which are given at the same time.

$$\mathbf{d}_i = (\mathbf{a}_i, o_i), \tag{2}$$

where $\mathbf{a}_i$ is a sequence of feature vectors extracted from a spoken utterance. Each feature vector corresponds to a speech frame of tens of milliseconds. The notation o_i is an ID representing an object. In the real world, a computer vision technique is necessary for robots to identify objects. However, this chapter does not address the problem of computer vision for focusing on automatic segmentation of continuous speech into words. Therefore, we assume that objects can be visually identified without errors and a module for word acquisition can receive IDs of objects as the identification results. In the following explanation, we call $\mathbf{a}_i$ an utterance, and o_i an object, and we omit index i of each variable.

The joint probability of $\mathbf{a}$ and o is denoted by $P(A=\mathbf{a}, O=o)$, where A and O are random variables. We assume that A and O are conditionally independent given a word sequence $\mathbf{s}$. This means that an utterance is an acoustic signal made from a word sequence and that the word sequence indicates an object. Therefore, $P(A=\mathbf{a}, O=o)$ is defined as follows.

$$\begin{aligned} P(A=\mathbf{a}, O=o) &= \sum_s P(A=\mathbf{a}, O=o, S=\mathbf{s}) \\ &= \sum_s \{P(A=\mathbf{a} \mid S=\mathbf{s}) P(S=\mathbf{s}) P(O=o \mid S=\mathbf{s})\} \end{aligned} \tag{3}$$

We call $P(A=\mathbf{a}, O=o)$ utterance-object joint probability. Figure 1 shows a graphical model of $P(A,O)$. The notations S and W_j are random variables representing a word sequence and each word, respectively.

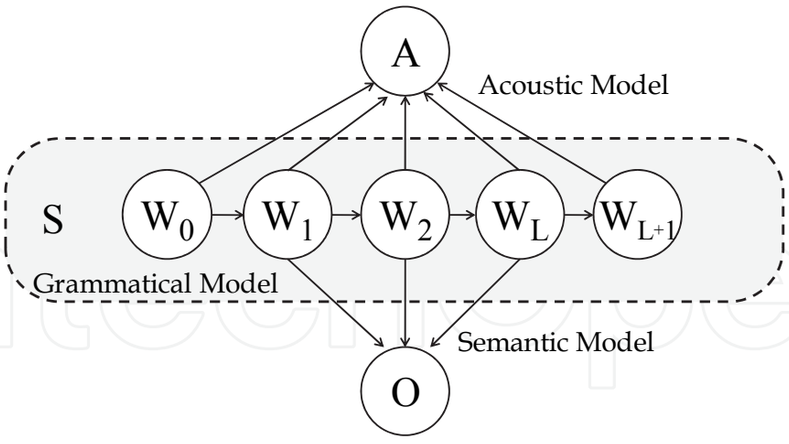

Fig. 1. Utterance-object joint probability model.

In the following explanation, we omit random variables to simplify formulas. The notation $P(\mathbf{a} \mid \mathbf{s})$ is the probability of an acoustic feature given a word sequence. $P(\mathbf{a} \mid \mathbf{s})$ is calculated from the phoneme acoustic model as usual speech recognition systems do. We use a hidden Markov model as the phoneme acoustic model. The learning of the phoneme acoustic model requires much more speech data. However the phoneme acoustic model can be learned before the lexical learning task because it does not depend on domains. $P(\mathbf{s})$ is the

probability of a word sequence, which we call the grammatical model, and $P(o \mid \mathbf{s})$ is the probability of an object given a word sequence. It represents a meaning of an utterance. We call it the semantic model.

In general statistical speech recognition algorithms, the acoustic and grammatical models are generally used. On the other hand, in the utterance-object joint probability model, the semantic model is also used.

Equation (3) requires a large amount of calculation because there are a large number of word sequences. Therefore, we approximate the summation by maximization as expressed by Eq. (4). This approximation enables efficient probability calculation using the beam search algorithm.

$$P(\mathbf{a}, o) = \max_{\mathbf{s}} \{ P(\mathbf{a} \mid \mathbf{s}) P(\mathbf{s}) P(o \mid \mathbf{s}) \} \tag{4}$$

The acoustic, grammatical, and semantic models differ in modeling accuracy. In statistical speech recognition algorithms, a weighting parameter is used to decrease a difference between the acoustic and grammatical models. In our method, we multiply the acoustic score by the weighting parameter α. We call α acoustic model weight.

The logarithm of utterance-object joint probability is defined as follows:

$$\log P(\mathbf{a}, o) \approx \max_{s} \{ \alpha \log P(\mathbf{a} \mid \mathbf{s}) + \log P(\mathbf{s}) + \log P(o \mid \mathbf{s}) \} \tag{5}$$

We verified practical effectiveness of weighting $P(\mathbf{s})$ or $P(o \mid \mathbf{s})$ through preliminary experiments, but they were not effective.

2.2 Grammatical model

We use a word-bigram model as the grammatical model.

$$P(\mathbf{s}) = \prod_{i=1}^{L+1} P(w_i \mid w_{i-1}), \tag{6}$$

where w_i is the i-th word in **s**, w_0 is the start point, and w_{L+1} is the end point. A general word-bigram model represents the relationship between two words. However, the bigram model used in our method represents the relationship between keywords and each non-keyword expression. The words that are considered as keywords are not distinguished each other and they are treated as the same word in the bigram model. Namely, this is a class bigram model in which keywords is considered as a class. A method for determining whether or not a word is a keyword is described in Section 2.4.

2.3 Semantic model

A word sequence consists of keywords and non-keyword expressions. In an ideal situation, **s** consists of a single keyword and some non-keyword expressions. However, in the initial stage of learning, some keywords can be wrongly divided into short keywords. In this case,

s can include several short keywords. Moreover, non-keyword expressions are independent of objects. Therefore, $P(o|\mathbf{s})$ is calculated from multiple keywords, as expressed by Eq. (7).

$$P(o|\mathbf{s}) = \sum_{i=1}^{L} \gamma(\mathbf{s},i)\, P(o|w_i) \tag{7}$$

where $P(o|w_i)$ represents the meaning of word w_i. Index i is from 1 to L because w_0 and w_{L+1} are independent of objects. The notation $\gamma(\mathbf{s},i)$ is the meaning weight of w_i and is calculated on the bases of the number of phonemes as follows:

$$\gamma(\mathbf{s},i) = \begin{cases} \dfrac{\mathrm{N}(w_i)}{\mathrm{N}(\mathbf{s})} & \text{if } w_i \text{ is a keyword} \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

where $N(w_i)$ is the number of phonemes of w_i, and $N(\mathbf{s})$ is the total amount of phonemes of keywords included in **s**. The meaning weight of w_i is assigned as zero when w_i is not a keyword. If **s** does not include any keyword, $P(o|\mathbf{s})$ is assigned as zero as a penalty for rejecting the recognition result.

$\gamma(\mathbf{s},i)$ is a heuristics. However, when **s** includes several keywords, the negative effects of short keywords, which are wrongly divided, are reduced by using the heuristics in which relatively long keywords are more effective for calculating $P(o|\mathbf{s})$.

2.4 Keyword determination

To determine whether or not a word is a keyword, the difference between the entropy of o and its conditional entropy given a word w is calculated as follows:

$$I(w) = -\sum_{o} P(o)\log P(o) + \sum_{o} P(o|w)\log P(o|w) \tag{9}$$

If w is a non-keyword expression, the conditional probability distribution $P(O|W{=}w)$ and probability distribution $\mathrm{P}(O)$ are approximately the same because w is independent of objects.

On the other hand, if w is a keyword, the entropy of $P(O|W{=}w)$ is lower than that of $\mathrm{P}(O)$ because $P(O|W{=}w)$ is narrower than $\mathrm{P}(O)$.

If the difference $I(w)$ is higher than the threshold T, w is considered a keyword. The threshold was manually determined on the basis of preliminary experimental results.

2.5 Keyword output

To correctly speak the name of o, the robot has to choose keyword $\tilde{w}$, the best representation of o, from many keywords acquired though learning. The formula for choosing $\tilde{w}$ is defined as Eq. (10).

$$\begin{aligned}\tilde{w} &= \arg\max_{w\in\Omega} P(w \mid o) \\ &= \arg\max_{w\in\Omega} P(w,o) \\ &= \arg\max_{w\in\Omega} \{ \log P(w) + \log P(o \mid w) \}\end{aligned} \quad , \tag{10}$$

where Ω is the set of acquired keywords.

3. Lexical learning algorithm

Figure 2 gives an overview of lexical learning algorithm. The algorithm consists of four steps. In step 1, all user utterances are recognized as phoneme sequences. Then the initial word list is built based on statistics of sub-sequences included in the phoneme sequences. In step 2, all user utterances are recognized as word sequences using the word list. Parameters of the grammatical and semantic models are learned from the recognition results. In step 3, the word list is rebuilt using the models that have been learned. Specifically, word deletion based on the minimum description length (MDL) principle and word concatenation based on the word-bigram model are executed. By this process, unnecessary words are deleted and those wrongly divided into short words in step 1 are restored. In step 4, model parameters are re-learned using the word list, which has been rebuilt. By repeating word list rebuilding (step 3) and model parameter re-learning (step 4), more correct phoneme sequences of keywords are acquired. The details of each step are explained after the next section.

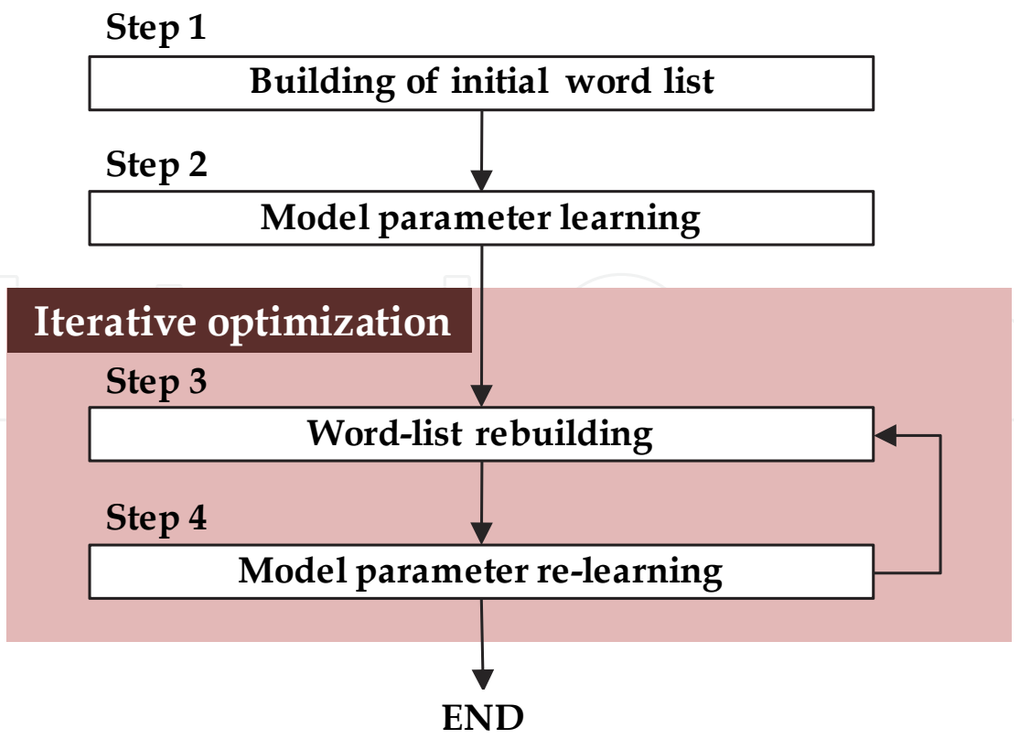

Fig. 2. Overview of lexical learning algorithm.

3.1 Step 1: building of initial word list

First, all user utterances are recognized as phoneme sequences by using the phoneme acoustic model. Next, a word list is built by extracting subsequences included in the phoneme sequences. The entropies of phonemes before or after each subsequence are calculated. If the boundary of a phoneme sequence equals the boundary of a true word, the entropies are high because varied phonemes, which are the start or end of other words, are observed before or after the sequence. If a word is divided into short sub-sequences, the entropies are low because specific phonemes, which are the start or end of the adjacent sub-sequences in the word, are observed before or after each sub-sequence. Many word candidates can be obtained with this algorithm when the entropies of a sub-sequence are not zero and its frequency is more than two, it is registered on the word list as a word candidate.

3.2 Step 2: model parameter learning

Utterances are recognized as word sequences using both the phoneme acoustic model and word list. Note that N-best hypotheses are output as a recognition result for each utterance in our algorithm. Parameters of the word-bigram and semantic models are learned from all word sequences included in the N-best hypotheses to improve the robustness of learning. Moreover, the backward bigram that predicts words before each word is also learned.

The word meaning model $P(o \mid w)$ is calculated as follows.

$$P(o \mid w) = \frac{F(o,w)}{\sum_{o} F(o,w)} \tag{11}$$

where o is an object, w is a word and $F(o, w)$ is a co-occurrence frequency of o and w. $F(o, w)$ is calculated as follows.

$$F(o,w) = \sum_{i=1}^{M} \frac{1}{N_i} \sum_{j=1}^{N_i} F(o,w,\mathbf{s}_j^i) \tag{12}$$

$$F(o,w,\mathbf{s}_j^i) = \begin{cases} 1 & \text{if } o = o_i \text{ and } w \in \mathbf{s}_j^i \\ 0 & \text{otherwise} \end{cases}, \tag{13}$$

where M is the number of learning samples, N_i is the number of hypotheses obtained by recognizing utterance $\mathbf{a}_i$ and $\mathbf{s}^i_{\mathrm{j}}$ is a word sequence of j-th hypothesis. The notation $F(o, w, \mathbf{s}^i_{\mathrm{j}})$ represents the co-occurrence of o and w in $\mathbf{s}^i_{\mathrm{j}}$. In this algorithm, the number of actual N-best hypotheses differs from utterance to utterance because the beam search algorithm is used. Therefore, $P(o \mid w)$ is calculated by normalizing the frequency of $F(o, w, \mathbf{s}^i_{\mathrm{j}})$ by N_i.

3.3 Step 3: word-list rebuilding

3.3.1 Word deletion using MDL

Unnecessary words in the word list are deleted based on the MDL principle (Rissanen, 1983). The sum of the description length of observed data by each model, and description

length of parameters of the model is calculated in this principle. Then, the model that has the minimum sum is chosen as the best.

In this algorithm, the description length of the model parameter set $\boldsymbol{\theta}$, which consists of the word list and parameters of each probability model, and learning sample set **D** is defined as follows:

$$DL(\boldsymbol{\theta}) = -L(\mathbf{D}\,|\,\boldsymbol{\theta}) + \frac{f(\boldsymbol{\theta})}{2}\log M\,, \tag{14}$$

where $L(\mathbf{D}\,|\,\boldsymbol{\theta})$ is a log likelihood of $\boldsymbol{\theta}$, $f(\boldsymbol{\theta})$ is the degree of freedom of $\boldsymbol{\theta}$, and M is the number of learning samples. $L(\mathbf{D}\,|\,\boldsymbol{\theta})$ and $f(\boldsymbol{\theta})$ are calculated using Eqs. (15) and (16), respectively.

$$\begin{aligned} L(\mathbf{D}\,|\,\boldsymbol{\theta}) &= \sum_{i=1}^{M} \log P(\mathbf{a}_i, o_i\,|\,\boldsymbol{\theta}) \\ &= \sum_{i=1}^{M} \log \left\{ \sum_{\mathbf{s}} P(\mathbf{a}_i, o_i, \mathbf{s}\,|\,\boldsymbol{\theta}) \right\} \\ &\approx \sum_{i=1}^{M} \log \left\{ \max_{\mathbf{s}\in\boldsymbol{\Psi}_i} P(\mathbf{a}_i, o_i, \mathbf{s}\,|\,\boldsymbol{\theta}) \right\} \end{aligned} \tag{15}$$

$$f(\boldsymbol{\theta}) = K + (K^2 + 2K) + CK\,, \tag{16}$$

where $\boldsymbol{\psi}_i$ is the N-best hypotheses obtained by recognizing utterance $\mathbf{a}_i$ ($\boldsymbol{\psi}_i = \{\ \mathbf{s}_j^i \,|\, 1 \le i \le N_i\ \}$), K is the number of words in the word list, and C is the number of object IDs.

The first term "K" in the right-hand side of Eq. (16) means the number of parameters of the word list, the second term "(K^2+2K)" means the number of parameters of the grammatical model, and the third term "CK" means the number of parameters of the semantic model. Note that $f(\boldsymbol{\theta})$ does not include the number of parameters of the acoustic model because it is not learned.

These definitions are not strict MDL because there are some approximations and the acoustic model weight α is used. However, we believe they work well.

The optimization of the word list requires calculating the log likelihoods in all combinations of possible word candidates. However, it is computationally expensive and not practical. Therefore, using the N-best hypotheses obtained in Step 2, we approximately calculate the difference in the description lengths of two models, one that includes w and the other that does not. This is done by computing the likelihood of the hypothesis that is the highest among those that do not include w.

The model obtained by subtracting word w from the original model $\boldsymbol{\theta}$ is denoted by $\boldsymbol{\theta}^{-w}$. The description length $DL(\boldsymbol{\theta}^{-w})$ is calculated by subtracting the difference from $DL(\boldsymbol{\theta})$. If

$DL(\boldsymbol{\theta}^{-w})$ is lower than $DL(\boldsymbol{\theta})$, w is removed from the original model $\boldsymbol{\theta}$. This word deletion is iterated in order of decreasing difference of DLs. When no w can be removed, the word deletion process finishes. A flowchart of word deletion is shown in Fig. 3.

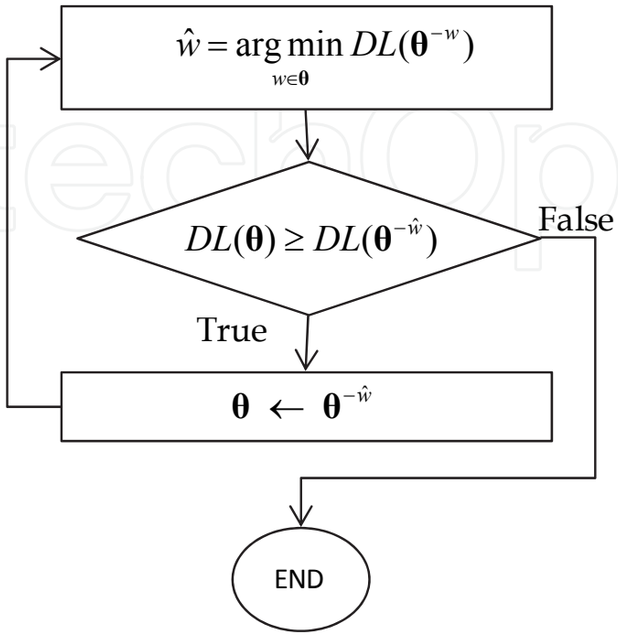

Fig. 3. Flowchart of word deletion.

3.3.2 Word concatenation using word-bigram model

If forward or backward bigram probability of two words is higher than a certain threshold (0.5 in this work), a new word candidate is generated by concatenating them into one word. This leads to recovering the erroneous dividing of words in Step 1. A new word list is built by merging the word-deletion and word-concatenation results.

3.4 Step 4: model parameter re-learning

The parameters of the word bigram and semantic models have to be re-learned because the composition of the word list changed in step 3. Therefore, they are learned using the same algorithm as in step 2.

3.5 Iterative optimization of steps 3 and 4

The new word candidates obtained by word concatenation are not based on the MDL principle because they are generated using the word-bigram model. Moreover, the words that have already been concatenated may not be removed. The necessity of each word has to be determined using the MDL principle. Therefore, word deletion and word concatenation are re-executed in step 3. Through the iteration of steps 3 and 4, acoustically, grammatically, and semantically useful words are acquired. However, word deletion in step 3 is local optimum. For this reason, after some iterations, the result that has the minimum DL is chosen as the best.

4. Experimental results

4.1 Conditions

To verify the effectiveness of the proposed method, we conducted experiments in which a navigation robot learns the names of locations in an office from Japanese utterances of a user. There were ten locations and each location had an object ID. The keywords corresponding to the locations are listed in Table 1. Six non-keyword expressions were used such as "kokowa <*keyword*> desu", which means "this is <*keyword*>" in English, and "konobasyowa <*keyword*>", which means "this place is <*keyword*>" in English, where each keyword can replace <*keyword*>. The sixty utterances, which consisted of all combinations, were recorded in a noiseless environment. Speakers of the utterances were seventeen Japanese men.

After learning from the data set of each speaker, the robot output ten keywords representing each location based on Eq. (10). The phoneme accuracy for the keywords was estimated using Eq. (17).

$$Acc = \frac{N - D - S - I}{N}, \tag{17}$$

where N is the number of the phonemes of true keywords, D is the number of deleted phonemes, S is the number of substituted phonemes, and I is the number of inserted phonemes. ATR Automatic Speech Recognition (ATRASR) (Nakamura et al., 2006) was used for phoneme recognition and connected word recognition. An acoustic model and finite-state automaton for Japanese phonemes were given, but the knowledge of words was not. By using ATRASR, the average phoneme accuracy was 81.4%, the best phoneme accuracy was 90.4%, and the worst phoneme accuracy was 71.8% for the seventeen speakers' data.

In the first experiment to determine an acoustic model weight α, we investigated the effect of the acoustic model weight using spoken utterance data from one person. In the second experiment, we investigated the effectiveness of iterative optimization using spoken utterance data of sixteen speakers.

Object ID	Keyword (in Japanese)	in English
1	/kaigishitsunomae/	the front of a meeting room
2	/tsuzinosaNnobuusu/	Tsuzino's booth
3	/furoanomaNnaka/	the center of a floor
4	/gakuseebeyanomae/	the front of a student room
5	/ochanomiba/	a lounge
6	/takeuchisaNnobuusunominami/	the south of Takeuchi's booth
7	/koosakushitsu/	a workshop
8	/ashimonoheya/	Ashimo's room
9	/sumaatoruumu/	Smart room
10	/sumaatoruumunoiriguchi/	the entrance of smart room

Table 1. Keywords used in experiments.

Non-keyword expressions (in Japanese)	in English
/kokononamaewa/ <*keyword*>	This place is called <*keyword*>.
/kokowa/ <*keyword*> /desu/	This is <*keyword*>.
/konobashowa/ <*keyword*>	<*keyword*> is here.
<*keyword*> /notokoroniiqte/	Please go to <*keyword*>.
<*keyword*> /eonegai/	Take me to <*keyword*>, please.
/imakara/ <*keyword*> /eiqte/	Go to <*keyword*> now.

Table 2. Non-keyword expressions used in experiments.

4.2 Effect of acoustic model weight

To determine an acoustic model weight α, we investigated its effect using spoken utterance data from one person picked at random. The phoneme accuracy was 86.8% for utterances of this person. After repeating word-list rebuilding (step 3) and model parameter re-learning (step 4) nine times, the model that had the minimum DL was chosen. Ten keywords corresponding to the ten objects were output using this model. We calculated the average phoneme accuracy for the output keywords. We call this accuracy output keyword phoneme accuracy.

The effect of the acoustic model weight α on output keyword phoneme accuracy is shown in Figure 4. When $\alpha = 10^{-4}$ or $\alpha = 10^{-5}$, the output keyword phoneme accuracy was the best (90.7%). If the weight was reduced too much, output keyword phoneme accuracy decreased because the acoustic adequacy of each word was ignored.

Figure 5 shows the number of words registered in the word list and the number of keywords determined using Eq. (9). In this experiment, the correct number of words was eighteen and the correct number of keywords was ten. When $\alpha = 10^{-4}$ or $\alpha = 10^{-5}$, the number of words and keywords were correct.

Figures 4 and 5 show that $\alpha = 10^{-4}$ or $\alpha = 10^{-5}$ is the best. Therefore, we set $\alpha = 10^{-5}$ in the second experiment.

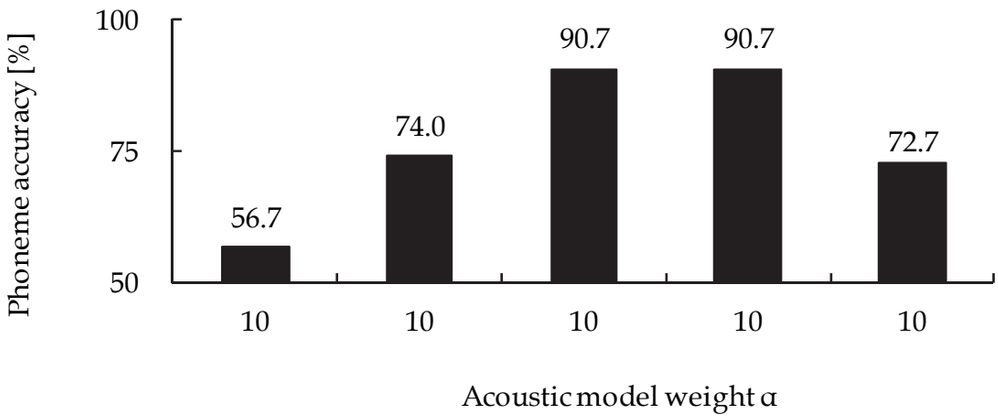

Fig. 4. Effects of acoustic model weight on optimum keyword phoneme accuracy.

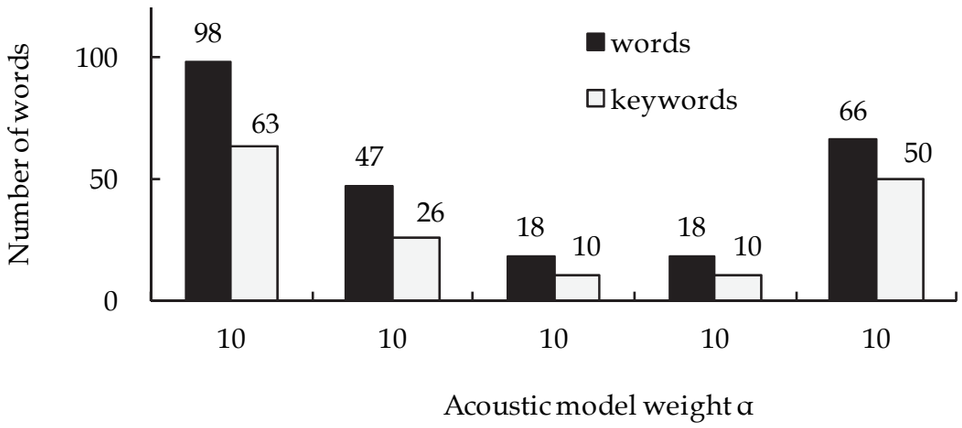

Fig. 5. Effects of acoustic model weight on number of acquired words.

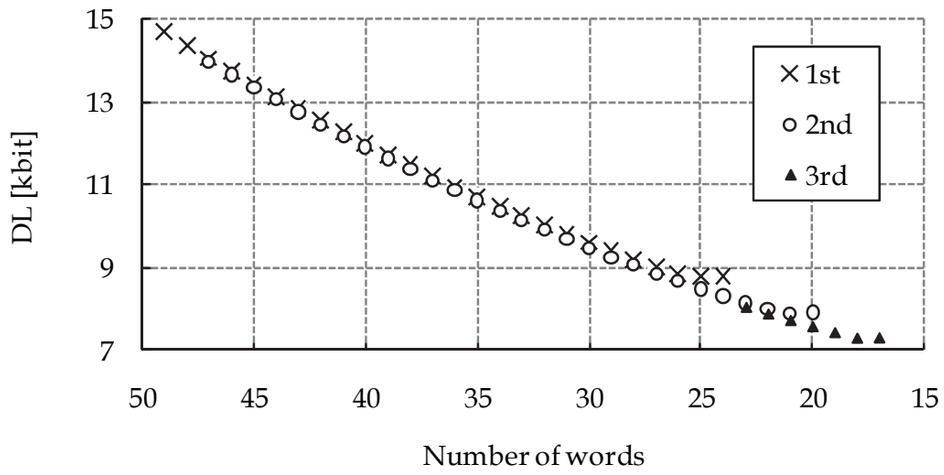

Fig. 6. Variation of description length in iterative optimization process.

4.3 Effects of iterative optimization

4.3.1 Variation in description length in iterative optimization process

To explain how the MDL principle works in iterative optimization, Figure 6 shows the variation of the DL in the above-mentioned experiment (more than 50 words were omitted). The initial word list, which consisted of 215 words, was constructed in step 1. The word-bigram and semantic models of these words were learned through step 2. Then, the first word deletion ("1st" in this figure) was executed. This word deletion was halted at 25 words because the DL of 24 words was higher than that of 25 words. A new word list consisting of 46 words was constructed by integrating the 25 words and the 22 words made by word concatenation. After model parameter re-learning, the second word deletion was executed ("2nd" in this figure). Through the iterations of steps 3 and 4, the number of newly added words gradually decreased, and the number of words was convergent.

4.3.2 Evaluation of iterative optimization process

We evaluated the effectiveness of the iterative optimization process from experiments using a sample data set of sixteen speakers other than the speaker of the above experiment. Figure 7 shows the average results among all speakers. The horizontal axis represents the number of iterations. The histogram indicates the number of acquired words and keywords included in the word list. We can see that the iterations decreased the number of words. Finally, an average of thirteen keywords was obtained. This number is close to ten, which is the correct number of keywords in the training utterance set. The dashed line in this figure represents phoneme accuracy for manually segmented keywords, which were obtained by manually segmenting phoneme sequences of all utterances into the correct word sequence. This accuracy was 81.5%. The solid line in this figure represents the output keyword phoneme accuracy of each learning result. This accuracy was 49.8% without optimization. In contrast, by iterating steps 3 and 4 accuracy increased up to 83.6%. This accuracy was slightly above the phoneme accuracy for manually segmented keywords.

Figure 8 shows the correct-segmentation, insertion error, and deletion error rates of output keywords. Correct segmentation means that there is no insertion error or deletion error at the start and end of an output keyword. The insertion and deletion error rates are the percentages of insertion errors and deletion errors occurring at the start or end of the output keywords. Many deletion errors occurred at the beginning of the iterations, but they decreased by iterative optimization. Finally, the correct-segmentation rate improved to 97%.

Table 3 lists examples of obtained keywords before and after iterative optimization. We can see that keyword segmentation errors were corrected. Table 4 lists examples of acquired non-keyword expressions after iterative optimization. We can also see that non-keyword expressions can be learned with high accuracy. These results prove that the proposed method makes it possible to appropriately determine the boundary of keywords.

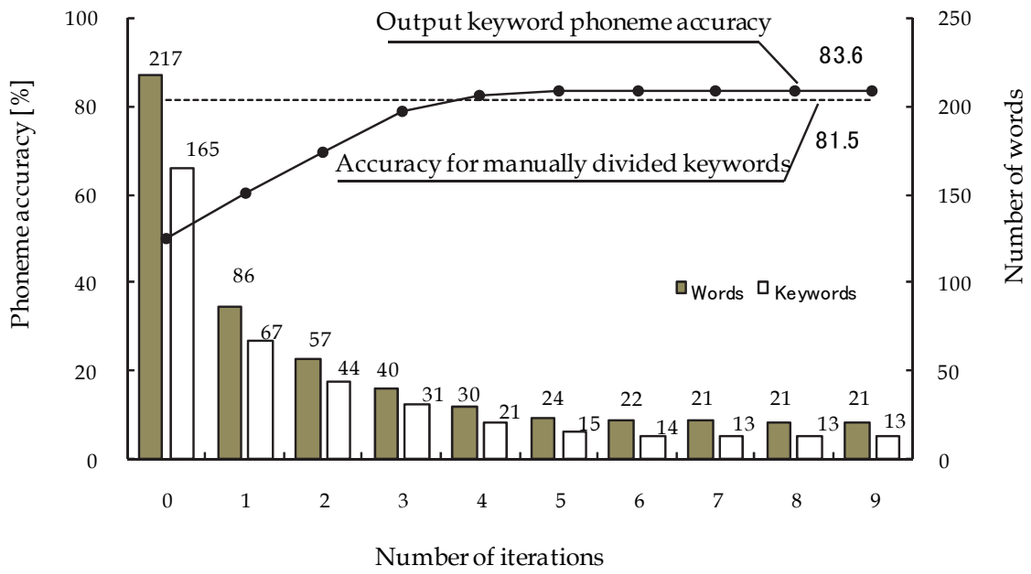

Fig. 7. Effects of iterative optimization on phoneme accuracy and number of words.

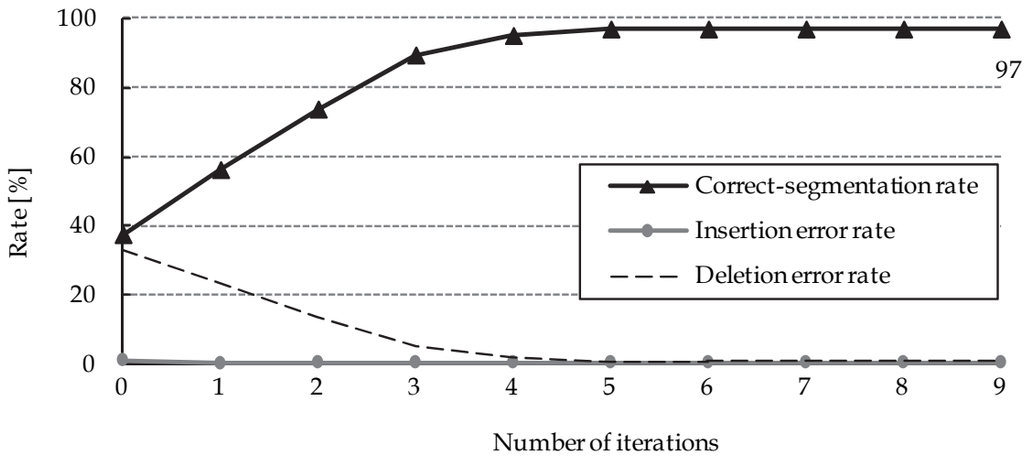

Fig. 8. Effects of iterative optimization on word segmentation.

ID	Correct keyword	Output keyword before iterative optimization	Output keyword after iterative optimization
1	/kaigishitsunomae/	/ka/	/kaigishitsugamae/
2	/tsuzinosaNnobuusu/	/tsuzinasaNnobuusu/	/tsuzinasaNnobuusu/
3	/furoanomaNnaka/	/furoanamaNnaka/	/furoanamaNnaka/
4	/gakuseebeyanomae/	/kuseebeyanamae/	/gakuseebeyanamae/
5	/ochanomiba/	/ba/	/watanamiba/
6	/takeuchisaNno buusunominami/	/taikee/	/taikeechisaNno buusunaminami/
7	/koosakushitsu/	/koosakushitsu/	/koosakushitsu/
8	/ashimonoheya/	/ashima/	/ashimanoheya/
9	/sumaatoruumu/	/mu/	/sumaatoruumu/
10	/sumaatoruumuno iriguchi/	/riguchi/	/sumaatoruguna iriguchi/

Table 3. Examples of output keywords before and after iterative optimization.

Correct non-keyword expression	Acquired non-keyword expression
/kokononamaewa/	/kokonagamaewa/
/kokowa/	/kokowa/
/desu/	/gesu/
/konobashowa/	/konabashowa/
/notokoroniiqte/	/notokoroniiqte/
/eonegai/	/eonegai/
/imakara/	/imakara/
/eiqte/	/ereiqke/

Table 4. Examples of acquired non-keyword expressions after iterative optimization.

4.4 Discussion

Experimental results show that the method acquired phoneme sequences of object names with 83.6% accuracy and a 97% correct-segmentation rate. The deletion error rate at the ends of words was 3%. These results suggest that keywords can be acquired with high accuracy. The phoneme accuracy of output keywords was slightly above the phoneme accuracy for manually segmented keywords. In manual segmentation, the average phoneme accuracy of each keyword was calculated from six keyword segments manually extracted from six utterances for learning the keyword. Therefore, the effect of variations in each utterance was included in the accuracy. For example, even if there is mispronunciation of one utterance, the average phoneme accuracy decreases. In word deletion using MDL, the acoustic score of each word was calculated from multiple utterances, and the words with high acoustic scores were kept. Keyword candidates extracted from utterances including mispronunciations were deleted because they had low acoustic scores. Therefore, such mispronunciations were corrected by word deletion, and the phoneme accuracy of output keywords improved.

In the real world, a computer vision technique is necessary for robots to identify objects. However, in our experiments, we assumed that objects can be visually identified without errors and a module for word acquisition can receive IDs of objects as the identification results. We believe that it is easy to extend the word meaning model. In fact, we proposed a method for automatically classifying continuous feature vectors of objects in parallel with lexical learning (Taguchi et al., 2011). In those experiments, a mobile robot learned ten location-names from pairs of a spoken utterance and a localization result, which represented the current location of the robot. The experimental results showed that the robot acquired phoneme sequences of location names with about 80% accuracy, which was nearly equal to the experiments in this chapter. Moreover, the area represented by each location-name was suitably learned.

5. Conclusions

We proposed a method for learning a physically grounded lexicon from spontaneous speeches. We formulated a joint probability model representing the relationship between an utterance and an object. By optimizing this model on the basis of the MDL principle, acoustically, grammatically, and semantically appropriate phoneme sequences were acquired as words. Experimental results show that, without a priori word knowledge, the method can acquire phoneme sequences of object names with 83.6% accuracy. We expect that the basic principle presented in this study will provide us with a clue to resolving the general language acquisition problem in which morphemes of spoken language are extracted using only non-linguistic semantic information related to each utterance.

6. References

Alshawi, H. (2003). *Effective utterance classification with unsupervised phonotactic models*, Proc. NAACL 2003.

Asadi, A. Schwartz, R. & Makhoul, J. (1991). *Automatic Modeling for Adding New Words to a Large Vocabulary Continuous Speech Recognition System*, Proc. ICASSP91, pp. 305--308.

Bazzi, I. & Glass, J. (2002). *A multi-class approach for modelling out-of-vocabulary words*, Proc. ICSLP02, pp. 1613--1616.

Gorin, A. L., Petrovska-Delacretaz D., Wright, J. H. & Riccardi, G. (1999). *Learning spoken language without transcription*, Proc. ASRU Workshop.

Holzapfel, H. Neubig, D. & Waibel, A. (2008). *A Dialogue Approach to Learning Object Descriptions and Semantic Categories*, Robotics and Autonomous Systems, Vol. 56, Issue 11, pp. 1004—1013.

Nakamura, S., Markov, K., Nakaiwa, H., Kikui, G., Kawai, H., Jitsuhiro, T., Zhang, J., Yamamoto, H., Sumita, E. & Yamamoto, S. (2006). *The ATR multilingual speech-to-speech translation system*, IEEE Trans. on Audio, Speech, and Language Processing, vol. 14, no. 2, pp. 365--376.

Rissanen, J. (1983). *A universal prior for integers and estimation by minimum description length*, The Annals of Stat., Vol. 11, No. 2, pp.416--431.

Roy, D. & Pentland, A. (2002). *Learning words from sights and sounds: A computational model*, Cognitive Science, 26, pp. 113--146.

Schaaf, T. (2001). *Detection of OOV Words Using Generalized Word Models And A Semantic Class Language Model*, Proc. Eurospeech 2001.

Taguchi, R., Yamada, Y., Hattoki, K., Umezaki, T., Hoguro, M., Iwahashi, N., Funakoshi, K. & Nakano, M. (2011). *Learning Place-Names from Spoken Utterances and Localization Results by Mobile Robot*, Proc. of INTERSPEECH2011, pp.1325--1328.

Human Machine Interaction - Getting Closer

Edited by Mr Inaki Maurtua

ISBN 978-953-307-890-8
Hard cover, 260 pages
Publisher InTech
Published online 25, January, 2012
Published in print edition January, 2012

In this book, the reader will find a set of papers divided into two sections. The first section presents different proposals focused on the human-machine interaction development process. The second section is devoted to different aspects of interaction, with a special emphasis on the physical interaction.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ryo Taguchi, Naoto Iwahashi, Kotaro Funakoshi, Mikio Nakano, Takashi Nose and Tsuneo Nitta (2012). Learning Physically Grounded Lexicons from Spoken Utterances, Human Machine Interaction - Getting Closer, Mr Inaki Maurtua (Ed.), ISBN: 978-953-307-890-8, InTech, Available from: http://www.intechopen.com/books/human-machine-interaction-getting-closer/learning-physically-grounded-lexicons-from-spoken-utterances

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen