Regular paper http://dx.doi.org/10.13067/JKIECS.2014.9.11.1221

음성 인식에서 음소 클러스터 수의 효과

이창영*

The Effect of the Number of Phoneme Clusters on Speech Recognition

Chang-Young Lee*

요 약

본 논문에서는 음성 인식의 효율을 높이기 위하여 음소 클러스터 개수의 효과에 대해 연구하였다. 이를 위하여 음소 클러스터 개수를 바꾸어 가면서 수정된 k-평균 군집 알고리듬을 사용하여 코우드북을 작성하였다. 그런 다음, 퍼지 벡터 양자화와 은닉 마코브 모델을 사용하여 음성인식 테스트를 수행하였다. 실험 결과 두 개의 영역이 구분되어 나타났다. 음소 클러스터 개수가 클 때 인식 성능은 대체로 그와 무관하지만, 개수가 작을 때에는 그 감소와 더불어 인식 오류율이 비선형적으로 증가하는 것으로 나타났다. 수치 해석적 계산으로부터, 이 비선형 영역은 멱승함수에 의해 모델링 될 수 있었다. 또한 300개의 고립단어 인식의 경우에, 166개의 음소 클러스터가 최적의 수임을 보일 수 있었다. 이는 음소당 3개 정도의 변화에 해당하는 값이다.

ABSTRACT

In an effort to improve the efficiency of the speech recognition, we investigate the effect of the number of phoneme clusters. For this purpose, codebooks of varied number of phoneme clusters are prepared by modified k-means clustering algorithm. The subsequent processing is fuzzy vector quantization (FVQ) and hidden Markov model (HMM) for speech recognition test. The result shows that there are two distinct regimes. For large number of phoneme clusters, the recognition performance is roughly independent of it. For small number of phoneme clusters, however, the recognition error rate increases nonlinearly as it is decreased. From numerical calculation, it is found that this nonlinear regime might be modeled by a power law function. The result also shows that about 166 phoneme clusters would be the optimal number for recognition of 300 isolated words. This amounts to roughly 3 variations per phoneme.

키워드

speech recognition, number of phoneme clusters, fuzzy vector quantization, hidden Markov model
음성 인식, 음소 클러스터 수, 퍼지 벡터 양자화, 은닉 마코브 모델

Ⅰ. ntroduction

The state of the art in the field of speech recognition has now reached such a level of perf-ormance and robustness, even in noisy enviro-nment, that permits lots of daily applications. Therethrough, we are now living in a world of various devices which deploy the relevant technology[1-4]. As a method of communication between man and machine, speech recognition aff-

* Div. of Information Systems Engineering, Dongseo University(seewhy@dongseo.ac.kr)
접수일자 : 2014. 08. 11 심사(수정)일자 : 2014. 10. 20 게재확정일자 : 2014. 11. 10

ords a very effective interface. Speech input to a machine is about twice as fast as information entry by a skilled typist[5].

It is known in practical applications that the absolute level of performance is relatively unimportant so long as the recognition accuracy exceeds some level[6]. When it is above a certain threshold (e.g. 92%), the user tends to attribute the occasional error to an improper and/or uncooperative speaking mode of his (or her) own, rather than to an inadequacy in the speech recognizer. If the performance falls below a certain level, however, the perception of the user is that the system makes too many errors and is therefore unreliable. There are so many factors affecting the performance of the speech recognition system and lots of endeavors for enhancement have been made for several decades.

One of the main elements governing the system accuracy might be phrased in terms of the clustering procedure. As a method to expedite the processing and save several kinds of cost, vector quantization (VQ) of the feature vectors extracted from the speech signal is frequently used. In this procedure, we consider some number of representative vectors (clusters or classes) and use them (actually their indices) in the pattern classifier such as HMM or neural networks. Here, the following question naturally arises: how many exemplary feature vectors are optimal for the best performance of a specific speech recognizer?

The number of clusters should somehow reflect the number of the basic elements of speech, i.e., phonemes in a language. It is usual to consider about 50 phonemes for speech processing[7], even though there are minor differences from language to language. Therefore, if we choose to use, for example, 256 clusters for vector quantization, it means that five variations for each phoneme on average are being considered. By 'variations' we mean not only the person-to-person differences but the context-dependence in speech production.

It is not known a priori how many variations for each phoneme would yield the best performance in speech recognition. If the number of clusters is too small, then the mesh of discrimination in the feature vector space becomes so coarse that the resolving power becomes weak and distinction between dissimilar patterns would become difficult. If the number of clusters is too large, on the other hand, then the mesh is so refined that similar enough patterns might be classified as different. The best number of clusters should be determined in such a way that discrimination between dissimilar and identification of similar patterns be optimally balanced.

For the clustering of the feature vectors, the Linde-Buzo-Gray (LBG) algorithm has long been used. In this method, the number of clusters are successively doubled on each bifurcation starting from a single cluster. Therefore, the number of clusters can not be chosen arbitrarily in this scheme. Codebooks of orders 8~10 corresponding to 256~1024 clusters are commonly used on empirical grounds.

To examine the effect of the number of clusters on speech recognition in more detail than permitted by the LBG algorithm, we need to employ another tool that permits us to choose the number of clusters freely.

The organization of this paper is as follows. Section II describes experimental details including a slightly modified clustering procedure. After providing experimental results and mathematical analysis for them in section III, concluding remarks will be given in section IV.

II. Experiment

Our experiments were performed on a set of phone-balanced 300 Korean words. Twenty male

and female speakers each produced speech. Though the amount of training data was insufficient, speech utterances of these people were divided into three disjoint groups as shown in Table I.

Table 1. Division of the 40 people's speech production into three groups

Group ID	Number of People
I	32
II	4
III	4

The group I consisting of 32 people's speech was used for codebook generation and training of HMM parameters: π, the initial state probabilities, $A=\{a_{ij}\}$, the state transition probabilities, and $B=\{b_i(j)\}$, the event observation probabilities. These parameters are continually updated in the course of training iterations. For the choice of $\lambda=(\pi, A, B)$ to be used in actual recognition test, some test speeches are necessary. The parameters that yield the best performance on the group II were used for the group III to get the final performance of the recognition system. This prescription prevents the system from falling too deep into the local minimum driven by the training data of the group I and hence becoming less robust against the speaker-independence when applied to the group III[8].

Each speech utterance was sampled at 16 kHz and quantized by 16 bits. 512 data points corresponding to 32 ms of time duration were taken to be a frame. The next frame was obtained by shifting 170 data points, thereby overlapping the adjacent frames by ≒2/3 in order not to lose any information contents of coarticulation. To each frame, Hanning window was applied after pre-emphasis of spectral flattening. MFCC feature vectors of order 13 were then obtained[9].

Codebooks of various number of clusters were generated by k-means clustering algorithm. In order not to have empty clusters, which incurs critical problem in clustering, we slightly modify the centroid update according to the work of Pakhira[10] as follows. In usual k-means clustering, centroid update is performed by

$$\mathbf{c}_k = \frac{1}{n_k} \sum_{\mathbf{x}_j \in \mathbf{c}_k} \mathbf{x}_j \tag{1}$$

where n_k is the number of vectors belonging to the class $\mathbf{c}_k$. However, in the new scheme, the update is done by

$$\mathbf{c}_k = \frac{1}{n_k+1} \left[\sum_{\mathbf{x}_j \in \mathbf{c}_k} \mathbf{x}_j + \mathbf{c}_k^{\text{old}} \right] \tag{2}$$

In this prescription, a fictitious vector is added as if it belongs to the updated cluster. This has the effect of removing the possibility of having empty cluster, which sometimes happens due to unlucky initialization. Once the codebook is thus generated, the next procedures are applying FVQ and feeding the resultant vectors into the pattern recognizer.

As one of the popular recognizers, we employ HMM with non-ergodic left-right (or Bakis) machine. The number of states that is set separately for each class (word) was made proportional to the average number of frames of the training samples in that class[11]. Initial estimation of HMM parameters was obtained by k-means segmental clustering after the first training. This procedure makes convergence of the parameters so fast.

Backward state transitions were prohibited by suppressing the state transition probabilities a_{ij} with $i>j$ to a very small value but skipping of states was allowed. The last frame was restricted to end up with the final state associated with the word being scored within a tolerance of 3. Parameter reestimation was performed by Baum-

Welch reestimation formula with scaled multiple observation sequences to avoid machine errors caused by repetitive multiplications of small numbers. After each iteration, the parameters $b_i(j)$ were boosted above a small value[12].

Three features were monitored while training the HMM parameters: (1) the recognition error rate for the group II of Table 1, (2) the total probability likelihood of events summed over all the words of the training set according to the trained model, and (3) the event observation probabilities for the first state of the first word in the vocabulary list. Training was terminated when the convergences for these three features were thought to be enough. The parameter values of $\lambda=(\pi,A,B)$ that give the best result for the group II were stored and used in speech recognition test on the group III.

III. Results and Discussion

Fig. 1 shows the recognition error rate E vs. the number of phonemes clusters N of codebook. We see that the recognition error rate is minimum in the vicinity of N=200. This result implies that consideration of around 4 variations for each Korean phoneme is appropriate for speech recognition task.

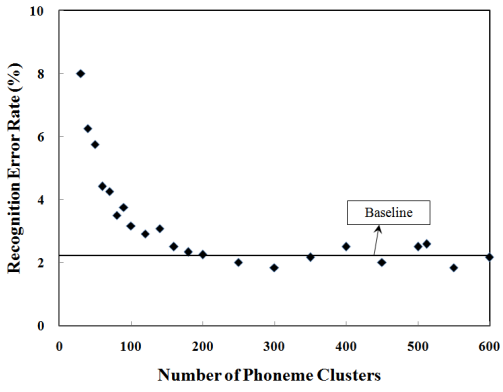

Fig. 1 The recognition error rate vs. the number of phonemes clusters of codebook

The horizontal solid line inserted in Fig. 1 denotes the ultimate value of the recognition error rate for the given system. It is noteworthy that further increase of the number of phoneme clusters above a certain threshold is of no use in enhancing the recognition performance. This implies that there exists an optimal value of the number of phoneme clusters that allows the best recognition performance with the least computational cost. We will pursue this issue soon.

Fig. 2 shows the result of Fisher discriminant analysis. It is shown that the discriminating power increases roughly in proportion to the number of clusters. This is as expected in view of the fact that the feature vector space becomes fine as the number of clusters is increased.

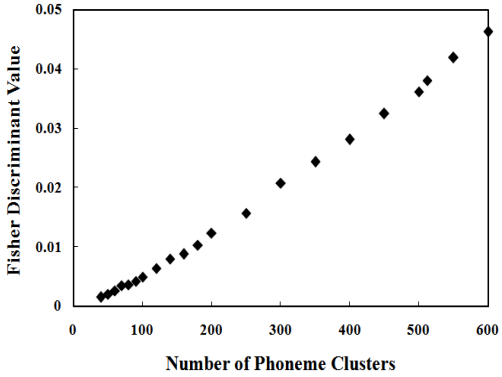

Fig. 2 The fisher discriminant value vs. the number of phoneme clusters

The data in Fig. 1 might be phrased in terms of two regimes. As the number of phonemes is decreased from large values, the recognition performance does not show significant change (regime I). However, below a certain threshold, recognition error rate begins to increase nonlinearly (regime II). For numerical analysis of the second regime, we first try by the exponential function

$$E = E_0 \exp(-\alpha N) \tag{3}$$

Here, E_0 and α are adjustable parameters which is determined for the best fit of the data. By taking the logarithm of both sides and applying the routine of the least square method, E_0 and α can be calculated. The result is given in Fig. 3.

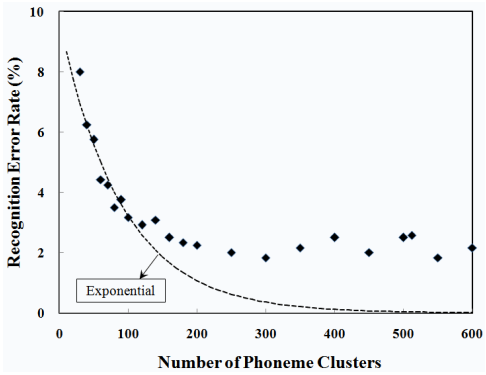

Fig. 3 Curve-fitting result of the recognition error rate vs. the number of clusters by exponential model as given by Eq. (3)

We found that

$$E = 9.7 \exp(-0.01N)$$

is the best fit. However, Fig. 3 shows that the result is not satisfactory. In a sense, this result is obvious since the exponential fit implies that the recognition does not diverge but converges to a finite value as the number of phoneme clusters approaches zero.

For this reason, we employ a power law model

$$E = ax^b \tag{4}$$

instead with a and b adjustable parameters. From a similar calculation, the best fit was found to be

$$E = 97\, x^{-0.74}$$

which is shown in Fig. 4 by a curved solid line. We see that the result is satisfactory this time.

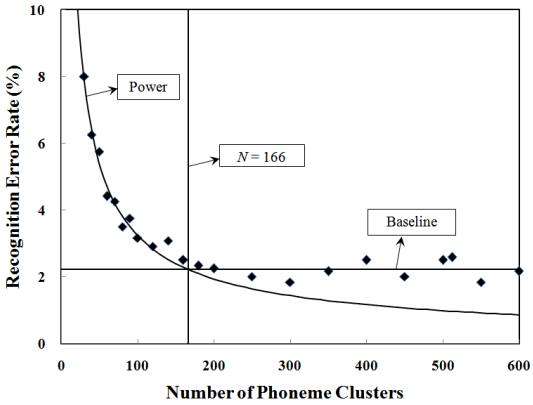

Fig. 4 Curve-fitting result of the recognition error rate vs. the number of clusters by power law model

In Fig. 4, it is also shown that the optimal value of the number of phoneme clusters is $N=166$, which corresponds to about 3 variations for each Korean phoneme. By "optimal", we mean that it requires the minimum computational cost without degradation of the system performance.

It should be remarked that the estimated parameter values in our work is system dependent. Our experiment was performed on a small-size environment due to various limitations. For large vocabulary and large speech tokens, it might be inferred that the optimal number of phonemes be larger than our result obtained in this paper. All in all, it might be phrased that it is desirable to search for the optimal number of phoneme clusters that requires minimum computational cost without deteriorating the recognition capability.

IV. Conclusion

In order to find the optimal number of clusters for speaker-independent speech recognition, we varied the number of clusters in codebook generation and examined the resultant effect on the speech recognition performance. As we decrease the cluster numbers from large values, the recognition

error rate does not show significant change until it reaches a certain threshold value. After that it begins to increase nonlinearly.

We modeled the nonlinear regime in two ways. Numerical estimation showed that power fit was better than exponential one. The result yielded that 166 cluster size for codebook was optimal in the sense that it requires the least computational cost without degrading the recognition performance.

This result suggests that about 3 variations per phoneme might be desirable at least in the case of our study. It should be kept in mind, however, that the obtained result be system dependent.

References

[1] Y. Chang, S. Hung, N. Wang, and B. Lin, "CSR: A Cloud-assisted speech recognition service for personal mobile device," *Int. Conf. on Parallel Processing*, Taipei, Taiwan, Sept. 2011, pp. 305-314.

[2] M. Kang, "A Study on the Design of Multimedia Service Platform on Wireless Intelligent Technology," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 4, no. 1, 2009, pp. 24-30.

[3] J. Yoo, H. Park, H. Shin, and Y. Shin, "A Study of the Communication Infrastructure Construction for u-City in Korea," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 1, no. 2, 2006, pp. 127-135.

[4] B. Kim, "Service Quality Criteria for Voice Services over a WiBro Network," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 6, no. 6, 2011, pp. 823-829.

[5] G. Kaplan, "Words Into Action I," *IEEE Spectrum*, vol. 17, 1980, pp. 22-26.

[6] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition.* Englewood Cliffs, New Jersey : Prentice Hall, 1993.

[7] J. Deller, J. Proakis, and J. Hansen, *Discrete Time Processing of Speech Signals.* New York : Macmillan, 1993, pp. 115-119.

[8] L. Fausett, *Fundamentals of Neural Networks.* Englewood Cliffs, New Jersey : Prentice Hall, 1994.

[9] J.-C. Wang, J.-F. Wang, and Y. Weng, "Chip design of MFCC extraction for speech recognition," *The VLSI J.*, vol. 32, 2002, pp. 111-131.

[10] M. K. Pakhira, "A Modified k-means Algorithm to Avoid Empty Clusters," *Int. J. of Recent Trends in Engineering*, vol. 1, no. 1, 2009, pp. 220-226.

[11] M. Dehghan, K. Faez, M. Ahmadi, and M. Shridhar, "Unconstrained Farsi Handwritten Word Recognition Using Fuzzy Vector Quantization and Hidden Markov Models," *Pattern Recognition Letters*, vol. 22, 2001, pp. 209-214.

[12] S. E. Levinson, L. R. Rabiner, and M. Sondhi, "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition," *Bell Systems Tech. J.*, vol. 62, no. 4, 1983, pp. 1035-1074.

저자 소개

이창영(Chang-Young Lee)

1982년 2월 서울대학교 물리교육학과 졸업(이학사)
1984년 2월 한국과학기술원 물리학과 졸업(이학석사)
1992년 8월 뉴욕주립대학교 (버펄로) 물리학과 졸업(이학박사)
1993년~현재 동서대학교 시스템경영공학과 교수
※ 관심분야 : 패턴인식, 신호처리