O. M. Tkachenko, Cand. Sc. (Eng.), Assist. Prof.; O. D. Feferman

METHODS OF DATA CLASTERIZATION IN DIGITAL PROCESSING OF SPEECH SIGNALS

The paper considers the main methods of data clusterization, their advantages and disadvantages are analyzed, further research lines are formulated.

Key words: *speech signals, compression, clusterization.*

Introduction

In the parametric coding of speech signals in current digital communication systems instead of direct transmission of linear prediction coefficients or linear spectral frequencies the indices of these coefficients in the table, i.e. in the coding book, are transmitted. At the receiving side these indices are decoded using the same code book, and from the received parameters a speech signal is synthesized.

Not all of the possible parameter values should be kept in the code book because in this case we cannot receive a big degree of compression. Therefore, only the most representative values are kept in it – such values with which the whole spectrum of the possible parameter values can be encoded. Thus, after a certain parameters vector is obtained during the signal coding process, search for the closest vector (the most similar one) is being performed in the code book and its index is transmitted into the communication channel.

As it is necessary for the code book values to be representative ones, the code book is created in the following way. First, a representative phonetic material is selected, i.e. the material containing all possible sounds, then it is encoded and a parameter set for each frame is obtained. Using this set of values, the values for the code book are calculated. For calculation of the code book values on the basis of the available material different clusterization methods are used.

There could be scalar and vector code books. A scalar code book contains a set of values for each parameters vector element separately. A vector code book contains a set of parameter vectors. Scalar code books are simpler in calculations and require less memory for storage. Computational complexity and the required memory capacity are higher while using vector code books as compared to scalar code books, but they provide higher degree of compression.

Different clusterization methods are used for both scalar and vector code books. In order to pass to direct consideration of clusterization methods it is necessary to define the term clusterization first.

Clusterization task in the code books creation

Clusterization is the division of a definite set of objects into subsets (clusters) that do not intersect so that each cluster contains similar objects, and the objects of different clusters must be different [1].

Let X be a set of objects, X^m – a final input sampling of objects, $X^m = \{x_1, x_2, ..., x_m\} \subset X$.

A definite function of the distance between $d(x, x')$ objects is set. On the basis of the input sampling of objects clusterization determines a set of clusters $Y = \{y_1, y_2, ..., y_n\}$.

To each $x_i \in X^m$ a corresponding $y_j \in Y$ is set. Each cluster consists of the objects that are close according to metric d . Function $\alpha : X \to Y$, that to each $x \in X$ sets a corresponding $y \in Y$, is defined.

In the case of these objects being numbers or vectors of numbers, clusterization is referred to as quantization.

Scalar quantizer Q of N size is transformation of real number set $x \in R$ into finite set Y , containing N values that are referred to as code vectors or centroids: $Q: R \to Y$, where $(y_1, y_2, ..., y_n) \in Y$ [2].

Y is referred to as a quantizer code book. The quantization operation can also be written in the following way:

$$Q(x) = y_i, x \in R, i = 1,..., N . \quad (1)$$

The code region for a scalar quantizer is a part of the real number space such that

$$R_i = \{x \in R, Q(x) = y_i\} = Q^{-1}(y_i), i = 1,2,..., N . \quad (2)$$

It is also clear, that

$$i \neq j \Rightarrow R_i \cap R_j = \varnothing . \quad (3)$$

And also

$$\bigcup_i R_i = R . \quad (4)$$

Code regions with boundaries are referred to as bounded regions. Regions that do not have boundaries are referred to as unbounded regions.

In the sphere of speech signal processing the operation of quantization can be treated as a set of coding operations (E) and decoding operations (D):

$$Q(x) = y_i \Rightarrow E(x) = i, D(i) = y_i . \quad (5)$$

A quantizer is considered to be optimal if it minimizes distortion D:

$$D = \int_{-\infty}^{\infty} d(x, Q(x)) f_x(x) dx , \quad (6)$$

where d is a distance function defined in the real numbers set.

The condition for the closest neighbor for quantizer optimality is formulated in the following way:

$$R_i = \{x : d(x, y_i) \leq d(x, y_j)\}, \forall j \neq i . \quad (7)$$

This means that for any vector falling into the code region the distance to a corresponding centroid must be smaller than the distance to any other centroid in the code book.

As $Q(x) = y_i$ only if $d(x, y_i) \leq d(x, y_j)$, we can say that

$$d(x, Q(x)) = \min_i (d(x, y_i)) . \quad (8)$$

Thus, the distance from a definite vector to its quantized value is the minimal value from all of the distances to each centroid in the code book.

The centroid optimality condition for code region $R_i, i = 1,2,..., N$:

$$\sum d(x, y_i) \to \min, x \in R_i . \quad (9)$$

This means that a centroid must be chosen so that it would minimize the distortion for each vector belonging to the given code region. This condition must be satisfied for all of the regions in the code book.

A vector quantizer is defined as follows. Vector quantizer Q of K dimensionality and N size is K-dimensional transformation of x vector from Euclidean space R^K into the final set Y, that contains N K-dimensional vectors referred to as code vectors or centroids $Q: R^K \to Y$, where $x = [x_1, x_2, ..., x_K]$, $(y_1, y_2, ..., y_N) \in Y$, $y_i = [y_i^1, y_i^2, ..., y_i^K]$, $i = 1,2,...N$.

Optimality condition for a vector quantizer is determined similar to that of a scalar quantizer.

Crisp and fuzzy clusterization. The method of k-means

In crisp clusterization the objects are divided into separate clusters so that each object can belong to one cluster only. In fuzzy clusterization an object may belong to several clusters. Its membership degree is specified for each of them and characterizes the degree of the object connection with a definite cluster.

An example of the crisp clusterization method is k-means algorithm. The number of clusters N is chosen first and initialization of $y_1, y_2, ..., y_N$ centroids is performed. Initialization could be performed both randomly and simply by even division as well as by the application of simpler clusterization algorithms. Then the input values classification is performed using the calculated centroids:

$$d(x, y_i) = \min_j (x, y_j) \Rightarrow Q(x) = y_i,\ j = 1,2,...,N\ . \tag{10}$$

At the next step each centroid is replaced by the mean value of the objects that belong to the corresponding cluster.

$$y_i = \frac{\sum_{k=1}^{K_i} x_k}{K_i}, Q(x_k) = y_i, i = 1,2,...,N\ , \tag{11}$$

where K_i is the number of vectors in the i-th cluster.

Given steps are repeated till at a certain step none of the centroids is changed.

An example of the fuzzy clusterization method is the method of fuzzy c-means [4]. The goal of this method consists in J_m index minimization:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m d^2(x_i, c_j), 1 \le m < \infty\ , \tag{12}$$

where C is the number of centroids; N – the number of input objects; x_i – i-th input object; c_j – j-th centroid; d – distance function; u_{ij} – the degree of x_i object membership in j-th cluster; m – the so called fuzzy parameter.

The algorithm operates in the following way. At the first step initialization of U vectors membership matrix is conducted. The initial value is assigned to the number of step k.

At each of the k-th step the centroids value is calculated:

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m}\ . \tag{13}$$

Then, after the centroids are renewed, renewal of the centroids membership matrix is performed:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{d(x_i - c_j)}{d(x_i - c_k)} \right)^{\frac{2}{m-1}}}\ . \tag{14}$$

After that the condition of the algorithm operation completion is calculated. If $d(U^{(k+1)}, U^k) < \varepsilon$, the algorithm operation is completed Otherwise the transition to step $k+1$ occurs. ε Is a positive number from 0 to 1.

Fuzzy clusterization algorithms such as c-means are rarely used in the problems of speech signals compression. More often they are used in the problems of their recognition and digital image processing. k-means methods is widely used for speech signals processing. In particular, it is used for the construction of scalar code books for CELP vocoder. The main advantage of this method is its simplicity and low computational complexity. The disadvantage consists in the dependence of the result of its operation on the initial choice of centroids. In general case the method does not give optimal results and allows finding only suboptimal solutions.

Clusterization with a number of clusters being known. LBG method

Certain methods require preliminary setting of the number of clusters into which the training sequence is to be divided. The above-considered k-means method is of this type. First, the number of clusters is set and initial values are assigned to the centroids. After that the code book optimization is performed through modification of centroids and code regions. There are also other methods such as LBG (Linde, Buzo, Gray) [5]. In this method it is not necessary to set the clusters number and to assign initial values to the centroids. Several stages can be distinguished in the method operation.

The code book initialization is performed first. Mean value of all the vectors is calculated:

$$c_1^* = \frac{1}{M}\sum_{m=1}^{M} x_m \,, \tag{15}$$

where c_1^* is the initial code vector, M – the number of training vectors, x_m – training vector.

So, this value is the first code vector that the code book contains. The number of vectors in the code book is set $N = 1$. After this the value of mean-square distortion is calculated:

$$D_{ave}^* = \frac{1}{Mk}\sum_{m=1}^{M} D(x_m, c_1^*) \,, \tag{16}$$

where k is vectors dimensionality and D function is defined as:

$$D(x,c) = (x_1 - c_1)^2 + (x_2 - c_2)^2 + \ldots + (x_k - c_k)^2 \,. \tag{17}$$

The next step is division of the vectors: each vector of the code book is divided into 2 vectors in the following way:

For $i = 1,2,\ldots,N$:

$$c_i^{(0)} = (1+\varepsilon)c_i^* \,, \tag{18}$$

$$c_{N+i}^{(0)} = (1-\varepsilon)c_i^* \,. \tag{19}$$

Then a number of steps is performed until a desired number of vectors is obtained (LBG makes it possible to receive 2^n vectors). At each step updating of the code regions takes place first. For each training vector the closest to this vector code region is found, and the training vector is transferred to the corresponding code region, i.e. if $m = 1,2,\ldots M$ and $n = 1,2,\ldots,N$:

$$\min(D(x_m, c_n^{(i)})) = D(x_m, c_{n^*}^{(i)}) \,, \tag{20}$$

where i is iteration number, then:

$$Q(x_m) = c_{n^*}^{(i)} \,. \tag{21}$$

After that the code vector values are renewed. To each code vector the average of all the vectors, belonging to the corresponding region, is assigned:

$$c_n^{(i+1)} = \frac{\sum_{Q(x_m)=c_n^{(i)}} x_m}{\sum_{Q(x_m)=c_n^{(i)}} 1}, \text{ n = 1, 2, …, N.} \quad (22)$$

Then a mean square distortion is calculated:

$$D_{ave}^{(i)} = \frac{1}{Mk}\sum_{m=1}^{M} D(x_m, Q(x_m)). \quad (23)$$

After this it is determined how this value has changed compared to the previous one. If necessary, the operation of the code regions and centroids updating is repeated and only after that transition to the next step takes place.

LBG method is widely used in speech signal and image processing, particularly, for speech signal compression. The main advantages of the given method are low computational complexity, easy implementation, independence of the method application results from the initial data (as centroids are formed dynamically on the basis of training sequence and there is no need for assigning initial values to them. On the whole the method gives good results but it has certain shortcomings connected with specific features of the method, and namely, uneven location of the centroids in the code book relative to training sequence objects distribution. This shortcoming is caused by the value of each following centroid being calculated on the basis of the previous one and, therefore, the centroids position cannot be changed radically, if it is necessary, because it can be changed only within certain limits.

Split and multistage quantization

While using vector code books, vector parameters of the same dimensionality as a training sequence vectors cannot be always kept in the code book. Representation of all the required variants of vectors in the code book would require much storage space, which complicates such systems implementation in personal computers and would make their hardware realization almost impossible.

Therefore, there exist two different approaches to the application of vector code books – split and multistage quantization.

In split quantization the parameter vector is divided into subvectors and each subvector is encoded using its own vector code book. The vector code book is built for each subvector independently using one of the clusterization algorithms. In this case it is necessary to determine the number of vectors that the initial vectors must be divided into, the number of elements in each subvector and the number of values to be kept in the code book for each subvector.

Multistage quantization is based on the absolutely different idea. The main difference consists in that the dimensionality of vectors, kept in the code book, coincides with that of the code sequence. Also, this method creates several code books. However, while a certain vector is quantized, it is replaced not by a vector from one of these code books, but by a combination of vectors from different code books formed using this method.

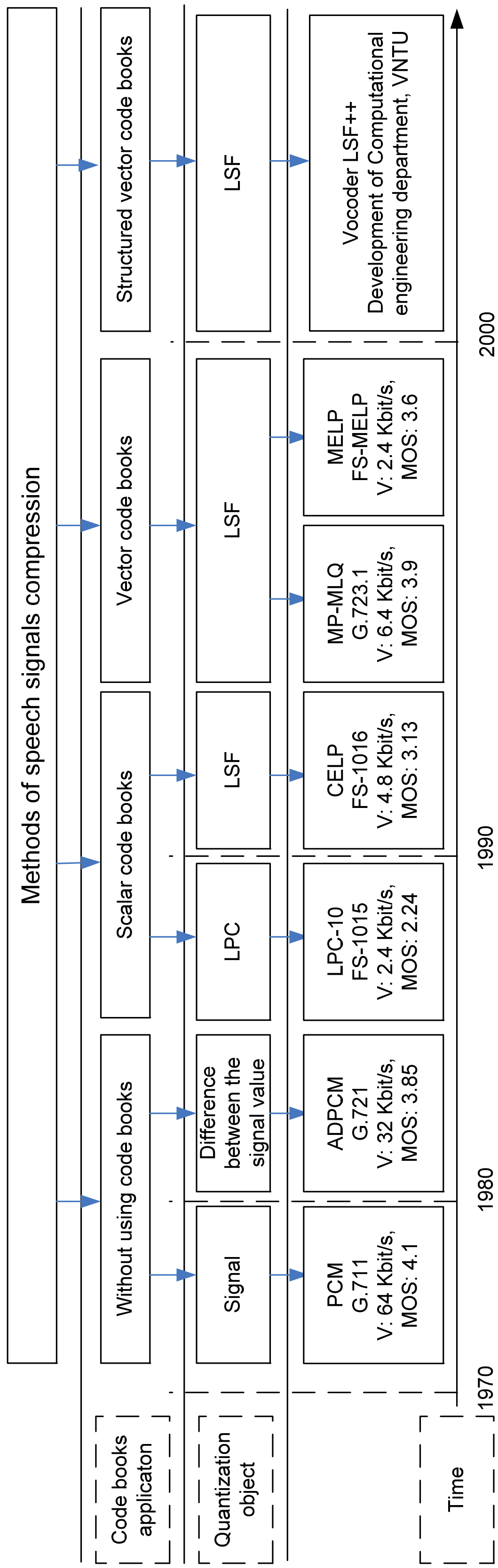
In the coder the input vector x is compared with the vector $\hat{x}$:

$$\hat{x} = y_{i1}^{(1)} + y_{i2}^{(2)} + \ldots + y_{iK}^{(K)}, \quad (24)$$

where $y_i^{(l)}$ – the i-th code vector from the code book obtained at the l-th stage. All the code books have the same dimensionality as the input vector.

By choosing different indices the coder tries to minimize the distance between x and $\hat{x}$. After such $\{i_1, i_2, \ldots, i_K\}$ set, which minimizes the distance, is found, the indices are transmitted to the receiving side where the decoder restores the signal using the same code books.

One of the main advantages of multistage quantization, compared to split quantization, is



* V – carrying capacity, required for a single-side speech channel creation .
** MOS (Mean opinion score) – Mean subjective evaluation of sound quality .

Fig. 1. Classification of the speech signals compression methods

the reduction of the memory space required for storing the code books. In terms of computational complexity, multistage quantization requires more resources than split quantization.

The main disadvantage of split quantization is the fact that in the process of quantization the parameter system can become less stable. There exist certain relationships between the elements of the quantized parameter vector that could be broken during the quantization process. This causes the corresponding distortion of the multiplexed signal. In multistage quantization this disadvantage is not observed because vectors, stored in the code books, have the same dimensionality as training vectors and the above-mentioned relationships are not broken in them. Multistage quantization is used for vector code books construction using MELP algorithm.

Classification of speech signals compression methods

As a rule, vocoders are classified according to a single parameter only – speech signal transmission in a communication channel. In this way they are divided into low-speed, average-speed and high-speed signals [6]. However, such classification does not allow evaluating the regularities in the development of speech signals compression methods.

Different clusterization methods, which could be used for speech signal compression, have been considered above. As in each of the speech signals compression methods quantization is used in a certain form, it can be taken as one of its classification criterion. Fig. 1 presents the proposed classification of speech signal compression methods.

First, the methods are classified according to the type of the code books that are used: those which use no code books and simply quantize the value according to a definite law, those using scalar code books and the ones using vector code books. The methods are further divided according to the quantization object (according to what is stored in the code book): the methods directly quantizing a signal, signal derivatives (e.g. difference between the previous and the next signal values), linear prediction coefficients (LPC) and the methods which quantize linear spectral frequencies (LSF).

Further research lines

The above classification makes it possible to trace the regularities in the development of speech compression methods and to predict the trends of their further development. As in this classification time sequence in the development of speech signals compression methods can be clearly observed, conclusions can be drawn about possible future trends.

1. Research and development of general principles for the construction of structured code books that must replace unstructured code books.
2. Development of the criteria to evaluate the quality of clusterization for structured code books.
3. Elaboration of the methods for ordering and search of the vectors in structured code books.
4. Linear spectral frequencies may still remain a quantization object in vector code books but investigation of alternative models for representation of the spectral information about a signal is considered to be a prospective trend.

REFERENCES

1. A. K. Jain. Data clustering: a review / A. K. Jain, M. N. Murty, P. J. Flynn.– ACM Comput. Surv.–1999. – №31. – 60 c.

2. J. Makhoul. Vector quantization in speech coding / J. Makhoul, S. Houcos, H. Gish . – Proc. IEEE. – 1985 – №73. – pp. 1551 – 1558.

3. P. Hedelin. Vector Quantization for Speech Transmission / P. Hedelin, P. Knagenhjelm, M. Skoglund. – Speech Coding and Synthesis. – 1995. – pp. 311 – 346.

4. J. C. Dunn A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, Well Separated Clusters / J. C. Dunn. – Journal of Cybernetics. – 1973. – №3. – pp. 32 – 57.

5. Linde Y. An Algorithm for Vector Quantizer Design / Linde, Y., Buzo, A., Gray, R. – IEEE Transactions on

Communications. – 1980. – №28., pp. 84 – 94.
6. Рабинер Л. Р. Цифровая обработка речевых сигналов: Пер. с англ. / Рабинер Л. Р., Шафер Р. В. – М.: Радио и связь, 1981. – 496 с.

Alexander Tkachenko – Cand. Sc. (Eng), Assoc. Prof. of the Computational engineering department.

Oleg Feferman – Post-graduate student of the Computational engineering department.
Vinnytsia National Technical University.