CHAPTER 3

Research Designs and Methods

This chapter describes the research designs and the proposed methods about string grammar fuzzy clustering methods. There are five sub-sections. Section 3.1.1 describes the string grammar fuzzy C-medians clustering (sgFCMed). Section 3.1.2 explains the string grammar fuzzy possibilistic C-Medians (sgFPCMed) clustering. The string grammar possibilistic fuzzy C-medians (sgPFCMed) is described in section 3.1.3. Section 3.1.4 describes the string grammar unsupervised possibilistic C-medians (sgUPCMed) and string grammar unsupervised possibilistic fuzzy C-medians (sgUPFCMed) is briefly review in Section 3.1.5.

3.1 String Grammar Fuzzy Clustering

Let $S = \{s_1, s_2, \ldots, s_N\}$ be a set of N strings. Each string (s_k) is a sequence of symbols (primitives). For example, $s_k = (x_1x_2\ldots x_l)$, a string with length l, where each x_i is a member of a set of defined symbols or primitives. Let $V = (sc_1, sc_2, \ldots, sc_c)$ represents a C-tuple of string prototypes each of which characterizes one of the C clusters. Then d_{ij} is computed from the Levenshtein distance between string s_j and string prototypes sc_i ($Lev(sc_i, s_j)$) (a smallest number of transformations needed to derive one string from the other) between input string j and cluster prototype i.

In this thesis, we proposed five string grammar fuzzy clustering methods, i.e., string grammar fuzzy C-medians clustering (sgFCMed), string grammar fuzzy possibilistic C-Medians (sgFPCMed), string grammar possibilistic fuzzy C-medians (sgPFCMed), string grammar unsupervised possibilistic C-medians (sgUPCMed) and string grammar unsupervised possibilistic fuzzy C-medians (sgUPFCMed) for eliminating the drawback of string grammar hard clustering method that is described in Chapter 1.

3.1.1 String Grammar Fuzzy C-medians

In the first algorithm, we modify fuzzy C-medians (FCMed) [11, 12] to cope with the syntactic data set which called the string grammar fuzzy C-Medians (sgFCMed) by using the Levenshtein distance instead of the Euclidean distance in the objective function and the sgFCMed aims at minimizing an objective function given by:

$$\begin{aligned} J_{m,\eta}(\mathbf{U},\mathbf{V};S) &= \sum_{k=1}^{N}\sum_{i=1}^{C}(u_{ik}^{m})Lev\left(s_k, sc_i\right) \\ s.t.: \sum_{i=1}^{C} u_{ik} &= 1,\ 0 \le u_{ik} \le 1 \ \text{ for } k = 1,...,N \end{aligned} . \tag{3.1}$$

Theorem 1 (sgFCMed): If $Lev(s_k, sc_i) > 0$ for all *i* and *k*, when *m, k*>1, and *S* contains *C*<*N* distinct string data, then J_m is minimized only if the update equation of u_{ik} is

$$u_{ik} = \frac{1}{\sum_{j=1}^{C}\left(\frac{Lev\left(sc_i, s_k\right)}{Lev\left(sc_j, s_k\right)}\right)^{1/(m-1)}}. \tag{3.2}$$

where u_{ik} is the membership value of string s_k belonging to cluster *i* , sc_i is the string prototype of cluster *i* and *m* is the fuzzifier.

Proof We proof equation 3.2 with the Lagrange multiplier theorem. Equation 3.2 is obtained by solving the reduced problem $\min_{\mathbf{U}\in M_{fcn}}\left\{J_m^k(\mathbf{U}) = \sum_{i=1}^{C} u_{ik}^{m} Lev\left(s_k, sc_i\right)\right\}$ with **V** fixed for the *k*-th column of **U**. Then, the function J_m^k is minimized over M_{fcn}. The Lagrangian of Equation 3.1 with constraints is as follows:

$$L_k(\mathbf{U},\lambda) = J_{m,\eta}^{k}(\mathbf{U},\mathbf{V}) - \lambda g(\mathbf{U}) \tag{3.3}$$

where λ is the Lagrange multiplier and $g(\mathbf{U}) = \sum_{i=1}^{C} u_{ik} - 1$.

Hence, we have

$$L_k(\mathbf{U},\lambda) = \sum_{i=1}^{C} u_{ik}^{m} Lev(s_k, sc_i) - \lambda\left(\sum_{i=1}^{C} u_{ik} - 1\right) \tag{3.4}$$

The Lagrangian's gradient is then set to zero, we obtain

$$\frac{\partial L_k(\mathbf{U},\lambda)}{\partial \lambda} = \sum_{i=1}^{C} u_{ik} - 1 = 0, \text{ this gives } \sum_{i=1}^{C} u_{ik} = 1, \tag{3.5}$$

and
$$\frac{\partial L_k(\mathbf{U},\lambda)}{\partial u_{jk}} = m(u_{jk})^{m-1} Lev(s_k, sc_j) + \lambda = 0. \tag{3.6}$$

Hence
$$u_{jk} = \left(-\lambda / m\right)^{1/(m-1)} \left(1 / Lev(s_k, sc_j)\right)^{1/(m-1)}. \tag{3.7}$$

Then u_{jk} is substituted into equation 3.5, and we have

$$\sum_{i=1}^{C} \left(-\lambda / m\right)^{1/(m-1)} \left(1 / Lev(s_k, sc_i)\right)^{1/(m-1)} = 1. \tag{3.8}$$

Hence the update equation of u_{ik} is

$$u_{ik} = 1 / \sum_{j=1}^{C} \left(\frac{Lev(sc_i, s_k)}{Lev(sc_j, s_k)} \right)^{1/(m-1)}. \tag{3.9}$$

Also, if there exists i such that $Lev(sc_i,s_k) = 0$, then it is naturally that the membership value of string k in those clusters will be non-zeros distributive, whereas those with $Lev(sc_i,s_k) > 0$ will be 0 subjected to $\sum_{i=1}^{C} u_{ik} = 1$. ■

Although the sgFCMed utilizes the Levenshtein distance not the Euclidean distance as in FCMed [11, 12], the effects of m on the sgFCMed are similar to those on FCMed. If m approaches to infinity or 1, the membership value of string k in cluster i will be

$$\lim_{m\to\infty}\{u_{ik}\} = \lim_{m\to\infty}\left[1 / \sum_{j=1}^{C} \left(\frac{Lev(sc_i, s_k)}{Lev(sc_j, s_k)} \right)^{1/(m-1)} \right] = 1/\mathrm{C} \quad \forall i,k, \tag{3.10}$$

and
$$\lim_{m\to 1^+}\{u_{ik}\} = \left\{ \begin{array}{l} 1; \text{if } Lev(sc_i, s_k) < Lev(sc_j, s_k), \forall j \neq i \\ 0; \text{ otherwise} \end{array} \right\} \forall i,k \quad \text{,respectively.} \tag{3.11}$$

To compute each string prototype, we cannot follow the fuzzy median using equation 3.12 as described in [23] because it is difficult to compute this fuzzy median to string in the sgFCMed. Normally, the root of equation 3.12 is fuzzy median of numeric vectors [11, 12].

$$\Psi_{fuzzy}(med_{il}) = \sum_{k=1}^{N} u_{ik}^{m} \operatorname{sgn}(x_{kl} - med_{il}) \quad \text{for } l = 1, ..., p \tag{3.12}$$

However, in [23] proposed a median string in a set of strings S of cluster *i* can be calculated as

$$sc_i = \arg\min_{j \in S_i} \sum_{k=1}^{N_i} Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C \tag{3.13}$$

where N_i is the number of strings in cluster *i*. Hence, we can modify equation 3.13 to incorporate the idea of fuzzy median by assuming that each string can be a prototype for a particular cluster. We then find the string that gives the minimum value of summation of Levenshtein distances between that string and other strings in the set with membership value of strings in that cluster. Therefore, the equation to find a fuzzy median string of cluster *i* is as follows:

$$sc_i = \arg\min_{j \in S} \sum_{k=1}^{N} u_{ik}^{m} Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C \tag{3.14}$$

We can use sc_i as a cluster center *i*. However, in [28], the modified median string has proved that it offers a better classification rate than the regular median string. We then modified the method in [23] to calculate our fuzzy median string. Let $\sum^*$ be the free monoid over the alphabet set $\sum$ and a set of strings $S \subseteq \sum^*$. This process is an approximation of fuzzy median finding by edition operations (insertion, deletion, and substitution) over each symbol of the string. The selected string (sc_i) will be the one that gives the minimum value, will be

$$sc_i = \arg\min_{j \in \Sigma^*} \sum_{k=1}^{N} u_{ik}^{m} Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C\,. \tag{3.15}$$

The algorithm of the modified fuzzy median string for sgFCMed is similar to the modified median string in Chapter 2, but we use equations 3.16 to 3.18 instead of equations 2.40 to 2.42

$$\text{If } \sum_{k=1}^{N}\left(u_{ik}^{m}\right) Lev\left(z', s_k\right) < \sum_{k=1}^{N}\left(u_{ik}^{m}\right) Lev\left(z, s_k\right) \tag{3.16}$$

$$\text{If } \sum_{k=1}^{N}\left(u_{ik}^{m}\right) Lev\left(x', s_k\right) < \sum_{k=1}^{N}\left(u_{ik}^{m}\right) Lev\left(x, s_k\right) \tag{3.17}$$

$$s' = \arg\min_{G \in \{s,x,y,z\}} \sum_{k=1}^{N}\left(u_{ik}^{m}\right) Lev\left(G, s_k\right). \tag{3.18}$$

The following is the summarization of the sgFCMed algorithm

Store N unlabeled finite strings $S = \{s_k; k = 1,..., N\}$

Initialize string prototypes for all C classes by using equation (3.12) for single prototype clustering and randomly choose string prototypes as the initial cluster centers for multi-prototypes clustering.

Set m

Do {

Compute Levenshtein distance between input string j and cluster prototype i ($Lev(s_j, sc_i)$)

Update membership value using equation (3.2)

Update center string of each cluster i (sc_i) using equation (3.14) and (3.15)

} **Until** (Maximum number of iterations or Levenshtein distance between cluster center of previous iteration and current iteration less than stopping criteria)

The time complexity of the modified fuzzy median for sgFCMed is approximately $O(l^3 \cdot c \cdot |\sum|)$ where l is the maximum length of the strings in S and c is number of cluster center.

Therefore, the computational complexity of the whole algorithm is $O((l^2 \cdot N^2)+(l^3 \cdot c \cdot |\sum|))$.

3.1.2 String Grammar Fuzzy Possibilistic C-medians

Previously, we describe sgFCMed, the sgFCMed is modified from fuzzy C-medians in which a fuzzy median approach is applied for finding fuzzy median string as the center of string data and then we improved a method to compute fuzzy median string

with the edition operations (insertion, deletion, and substitution) over each symbol of the string. However, the results of classification are not good for some applications with noise and overlapping data.

In order to improve performance of the string grammar clustering algorithm, we propose a string grammar fuzzy-possibilistic C-medians (sgFPCMed). In particular, an extension of fuzzy median is presented and applied to the FPCM [13] for string. This algorithm combines the properties of both string grammar fuzzy C-median and possibilistic theory. Membership and typicality are both important to improve the clustering result. The typicality will consider the clustering problem with respect to all N data, but not respect to all C cluster. It is an important term for reducing the effects of outliers. The sgFPCMed utilizes the Levenshtein distance [2] as a dissimilarity measure and the fuzzy median [11, 12] is utilized to calculate a cluster prototype almost similar to sgFCMed. Then the objective function of sgFPCMed depending on both membership and typicality can be shown as:

$$\begin{aligned} & J_{m,\eta}(\mathbf{U},\mathbf{V},\mathbf{T};S)=\sum_{k=1}^{N}\sum_{i=1}^{C}(u_{ik}^{m}+t_{ik}^{\eta})Lev\left(s_k,sc_i\right) \\ & s.t.:\sum_{i=1}^{C}u_{ik}=1,\ 0\le u_{ik}\le 1 \text{ for } k=1,...,N\ , \\ & \sum_{k=1}^{N}t_{ik}=1 \text{ and } \ 0\le t_{ik}\le 1 \text{ for } i=1,...,C \end{aligned} \tag{3.19}$$

Theorem 2 (sgFPCMed) Suppose $Lev(s_k,sc_i)>0$ for all i and k, when m>1, and S contains C<N distinct string data, then $(\mathbf{U},\mathbf{V},\mathbf{T})\in M_{fcn}\times M_{tcn}\times\Re^{P}$ may minimize $J_{m,\eta}$ only if the update equation of u_{ik} is

$$u_{ik}=\frac{1}{\sum_{j=1}^{C}\left(\frac{Lev\left(sc_i,s_k\right)}{Lev\left(sc_j,s_k\right)}\right)^{1/(m-1)}} \tag{3.20}$$

and the update equation of t_{ik} is

$$t_{ik} = \frac{1}{\sum_{j=1}^{N}\left(\frac{Lev(sc_i, s_k)}{Lev(sc_i, s_j)}\right)^{1/(m-1)}} \tag{3.21}$$

Proof The equations 3.20 and 3.21 are extension of the equation 2.14 and 2.15. These two equations follow immediately with the Lagrange multiplier theorem obtained by solving the reduced problem $\min_{\mathbf{U}\in M_{fcn}}\left\{J^k_{m,\eta}(\mathbf{U}) = \sum_{i=1}^{C}\left(u_{ik}^m + t_{ik}^\eta\right)Lev(s_k, sc_i)\right\}$ with **T** and **V** fixed for the *k*-th column of **U**. The proof of the equation 3.19 is similar to that in Theorem 1.

Similarly, The equation 3.21 can be proof as the first equation by solving the reduced problem $\min_{\mathbf{T}\in M_{tcn}}\left\{J^i_{m,\eta}(\mathbf{T}) = \sum_{k=1}^{N}\left(u_{ik}^m + t_{ik}^\eta\right)Lev(s_k, sc_i)\right\}$ with **U** and **V** fixed for the *k*-th row of **T**. The function $J^i_{m,\eta}$ is minimized over M_{tcn}. The Lagrangian of 3.19 with constraints is as follows:

$$L_i(\mathbf{T}, \lambda) = J^i_{m,\eta}(\mathbf{U}, \mathbf{T}, \mathbf{V}) - \lambda g(\mathbf{T}) \tag{3.22}$$

where λ is the Lagrange multiplier and $g(\mathbf{T}) = \sum_{k=1}^{N} t_{ik} - 1$. Hence, we have

$$L_i(\mathbf{T}, \lambda) = \sum_{k=1}^{N}(u_{ik}^m + t_{ik}^\eta)Lev(s_k, sc_i) - \lambda[\sum_{i=1}^{N} t_{ik} - 1] \tag{3.23}$$

Again, the Lagrangian's gradient is set to zero, we obtain

$$\frac{\partial L_i(\mathbf{T}, \lambda)}{\partial \lambda} = \sum_{k=1}^{N} t_{ik} - 1 = 0, \text{ this gives } \sum_{i=1}^{N} t_{ik} = 1, \tag{3.24}$$

and $$\frac{\partial L_i(\mathbf{T},\lambda)}{\partial t_{ij}} = \eta\left(t_{ij}\right)^{\eta-1} Lev\left(s_j, sc_i\right) + \lambda = 0\,. \tag{3.25}$$

Hence $$t_{ij} = \left(-\frac{\lambda}{\eta}\right)^{1/\eta-1}\left(\frac{1}{Lev\left(s_j, sc_i\right)}\right)^{1/\eta-1}. \tag{3.26}$$

We then substitute t_{ij} in equation 3.21, we get

$$\sum_{k=1}^{N}\left(-\frac{\lambda}{\eta}\right)^{1/\eta-1}\left(\frac{1}{Lev\left(s_k, sc_i\right)}\right)^{1/\eta-1} = 1\,. \tag{3.27}$$

Hence, the update equation of t_{ik} will be

$$t_{ik} = \frac{1}{\sum_{j=1}^{N}\left(\left(\frac{Lev\left(sc_i, s_k\right)}{Lev\left(sc_i, s_j\right)}\right)\right)^{1/(\eta-1)}}. \tag{3.28}$$

Again, if there is some object (k) such that $Lev(sc_i,s_k) = 0$, then the typicality value of string k in those clusters will be distributively non-zeros and those with $Lev(sc_i,s_k) > 0$ subjected to $\sum_{k=1}^{N} t_{ik} = 1$. ■

The effects of m and η on the sgFPCMed are similar to those on FPCM [13]. If m approaches to infinity or 1, the membership value of string k in cluster i will be similar to equations 3.10 and 3.11. Also, if η approaches to infinity or 1, the typicality value of string k in cluster i will be

$$\lim_{\eta\to\infty}\{t_{ik}\} = \lim_{\eta\to\infty}\left[\frac{1}{\sum_{j=1}^{N}\left(\frac{Lev\left(sc_i, s_k\right)}{Lev\left(sc_i, s_j\right)}\right)^{1/(\eta-1)}}\right] = \frac{1}{N} \quad \forall i,k \tag{3.29}$$

and $\lim_{\eta \to 1^+}\{t_{ik}\} = \left\{ \begin{matrix} 1; \text{if } Lev(sc_i, s_k) < Lev(sc_i, s_j), \forall j \neq k \\ 0; \text{ otherwise} \end{matrix} \right\} \forall i,k$, respectively. (3.30)

To compute each string prototype, because of the method in equation 3.14 involve only the membership values. Hence, we modified fuzzy median string method in equation 3.14 by incorporating a typicality value to find fuzzy median. Then the modified fuzzy median will be

$$sc_i = \arg\min_{j \in S} \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C . \tag{3.31}$$

Similarly, we then modify the equation 3.15 by incorporating a typicality value to the modified fuzzy median over the alphabet set $\sum$ as follows:

$$sc_i = \arg\min_{j \in \Sigma^*} \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C . \tag{3.32}$$

The algorithm of the modified fuzzy median string for sgFPCMed is also similar to the modified median string in Chapter 2, but we use equations 3.33 to 3.35 instead of equations 2.40 to 2.42

$$\text{If } \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(z', s_k\right) < \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(z, s_k\right) \tag{3.33}$$

$$\text{If } \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(x', s_k\right) < \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(x, s_k\right) \tag{3.34}$$

$$s' = \arg\min_{G \in \{s,x,y,z\}} \sum_{k=1}^{N} \left(u_{ik}^m + t_{ik}^\eta\right) Lev\left(G, s_k\right). \tag{3.35}$$

Hence, the summarized sgFPCMed algorithm is as follows:

Store N unlabeled finite strings $S = \{s_k; k = 1,..., N\}$

Initialize string prototypes for all C classes by using equation (3.12) for single prototype clustering and randomly choose string prototypes as the initial cluster centers for multi-prototypes clustering.

Set m,η

Do {

Compute Levenshtein distance between input string j and cluster prototype i ($Lev(s_j,sc_i)$)

Update membership value using equation (3.20)

Update typicality value using equation (3.21)

Update center string of each cluster i (sc_i) using equations (3.31) and (3.32)

} **Until** (Maximum number of iterations or Levenshtein distance between cluster center of previous iteration and current iteration less than stopping criteria)

The time complexity of the modified fuzzy median is approximately $O(l^3 \cdot c \cdot |\sum|)$ for each global iteration, where l is the maximum length of the strings in S and c is number of cluster center. Therefore, the computational complexity of the whole algorithm is $O((l^2 \cdot N^2)+(l^3 \cdot c \cdot |\sum)|)$.

3.1.3 String Grammar Possibilistic Fuzzy C-medians

The method of PFCM [14] is commonly used for clustering of numeric feature vector. However, it can be applied to string clustering and we call this algorithm as string grammar possibilistic fuzzy C-medians (sgPFCMed). This lead to the following optimization problem of sgPFCMed:

$$J_{m,\eta}(\mathbf{U},\mathbf{V},\mathbf{T};S) = \sum_{k=1}^{N}\sum_{i=1}^{C}(au_{ik}^{m} + bt_{ik}^{\eta})Lev\left(s_k, sc_i\right) + \sum_{i=1}^{C}\gamma_i\sum_{k=1}^{N}(1-t_{ik})^{\eta} \quad (3.36)$$

subject to the constraint: $m>1, \eta>1, a>0, b>0, \gamma>0, \sum_{i=1}^{C} u_{ik}=1$ for k = 1 to N, and $0 \le u_{ik}, t_k \le 1$.

Theorem 3 (sgPFCMed) If $Lev(s_k, sc_i) > 0$ for all i and k, when m, η, k>1, and S contains C<N distinct string data, then $(\mathbf{U}, \mathbf{V}, \mathbf{T}) \in \mathbf{M_{fcn}} \times \mathbf{M_{tcn}} \times \Re^P$ may minimize $J_{m,\eta}$ only if the update equation of u_{ik} is

$$u_{ik} = \left\{ \frac{1}{\sum_{j=1}^{C} \left(\frac{Lev(sc_i, s_k)}{Lev(sc_j, s_k)} \right)^{1/(m-1)}} \right\} \tag{3.37}$$

and then the update equation of t_{ik} is

$$t_{ik} = \frac{1}{1 + \left(\frac{b}{\gamma_i} Lev(sc_i, s_k) \right)^{\frac{1}{\eta-1}}}. \tag{3.38}$$

Proof Since equations 3.37 and 3.38 are the extension of equations 2.18 and 2.19. They follow immediately with the Lagrange multiplier theorem. Equation 3.33 is obtained by solving the reduced problem $\min_{\mathbf{U} \in M_{fcn}} \left\{ J^k_{m,\eta}(\mathbf{U}) = \sum_{i=1}^{C} \left(au^m_{ik} + bt^\eta_{ik} \right) Lev(s_k, sc_i) \right\}$ with **T** and **V** fixed for the k-th column of **U**. The proof of the equation 3.37 is similar to that in Theorem 1.Similarly, **U** and **V** are fixed for the i-th row **T**, equation 3.38 is proved by solving the problem $\min_{\mathbf{T} \in M_{tcn}} \left\{ J^{ik}_{m,\eta}(\mathbf{T}) = \left(au^m_{ik} + bt^\eta_{ik} \right) Lev(s_k, sc_i) + \gamma_i (1 - t_{ik}) \right\}$. The function $J^{ik}_{m,\eta}$ is minimized over M_{tcn}. The Differential of 3.32 with respect to t_{ik} and setting it to zero leads to:

$$\frac{\partial L_i(\mathbf{T},\lambda)}{\partial t_{ik}} = b\eta\left(t_{ik}\right)^{\eta-1} Lev\left(s_k, sc_i\right) - \gamma = 0\,. \tag{3.39}$$

Hence, the update equation of t_{ik} will be

$$t_{ik} = \frac{1}{1+\left(\frac{b}{\gamma_i} Lev\left(sc_i, s_k\right)\right)^{\frac{1}{\eta-1}}}\,. \tag{3.40}$$

■

The effects of m and η on the sgPFCMed are similar to those on PFCM. If m approaches to infinity or 1, the properties of membership value of string k in cluster i will be similar to equations 3.10 and 3.11. Also, if η approaches to infinity or 1, the typicality value of string k in cluster i will be

$$\lim_{\eta\to\infty}\{t_{ik}\} = \lim_{\eta\to\infty}\left[\frac{1}{1+\left(\frac{b}{\gamma_i} Lev\left(sc_i, s_k\right)\right)^{\frac{1}{\eta-1}}}\right] = 0.5 \quad \forall i,k \tag{3.41}$$

and

$$\lim_{\eta\to 1^+}\{t_{ik}\} = \left\{\begin{array}{l} 1; \text{if } bLev\left(sc_i, s_k\right) < \gamma_i \\ 0.5; \text{if } bLev\left(sc_i, s_k\right) = \gamma_i \\ 0; \text{ if } bLev\left(sc_i, s_k\right) > \gamma_i \end{array}\right\} \forall i,k\,, \tag{3.42}$$

respectively.

To compute each string prototype, we modified fuzzy median string method in equation 3.31 by adding parameters a and b to find fuzzy median. Then the modified fuzzy median for sgPFCMed will be

$$sc = \arg\min_{j\in S} \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta\right) Lev\left(s_j, s_k\right) \quad \text{for } 1 \le i \le C\,. \tag{3.43}$$

Similarly, we then modify the equation 3.32 by incorporating a typicality value of sgPFCMed to the modified fuzzy median over the alphabet set ∑ as follows:

$$sc_i = \arg\min_{j \in \Sigma^*} \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(s_j, s_k \right) \quad \text{for } 1 \le i \le C . \tag{3.44}$$

The algorithm of the modified fuzzy median string for sgFCMed is similar to the modified median string in Chapter 2, but we use equations 3.45 to 3.47 instead of equations 2.40 to 2.42

$$\text{If } \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(z', s_k \right) < \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(z, s_k \right) \tag{3.45}$$

$$\text{If } \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(x', s_k \right) < \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(x, s_k \right) \tag{3.46}$$

$$s' = \arg\min_{G \in \{s,x,y,z\}} \sum_{k=1}^{N} \left(au_{ik}^m + bt_{ik}^\eta \right) Lev\left(G, s_k \right) . \tag{3.47}$$

Hence, the summarized sgPFCMed algorithm is as follows:

Store N unlabeled finite strings $S = \{s_k; k = 1,..., N\}$

Initialize string prototypes for all C classes by using equation (3.12) for single prototype clustering and randomly choose string prototypes as the initial cluster centers for multi-prototypes clustering.

Set m,η,γ

Do {

Compute Levenshtein distance between input string j and cluster prototype i ($Lev(s_j,sc_i)$)

Update membership value using equation (3.37)

Update typicality value using equation (3.38)

Update center string of each cluster i (sc_i) using equation (3.43) and (3.44)

} **Until** (Maximum number of iterations or Levenshtein distance between cluster center of previous iteration and current iteration less than stopping criteria)

The time complexity of the modified fuzzy median is approximately $O(l^3 \cdot c \cdot |\sum|)$ for each global iteration, where l is the maximum length of the strings in S and c is number of cluster center. Again, the computational complexity of the whole algorithm is $O((l^2 \cdot N^2)+(l^3 \cdot c \cdot |\sum|))$.

3.1.4 String Grammar Unsupervised Possibilistic C-medians

The unsupervised possibilistic C-means (UPCM) [15] is another posibilistic clustering algorithm which is based on the FCM objective function, the partition coefficient (PC) and partition entropy (PE) validity indexs. In order to classify string objects. On the basis of UPCM algorithm, we are going to introduce a new possibilistic clustering techniques for string which is adapted from UPCM. This technique is called string grammar unsupervised possibilistic C-medians (sgUPCMed). Then the objective function of sgUPCMed is

$$\min \sum_{k=1}^{N}\sum_{i=1}^{C} u_{ik}^{m} Lev(s_k, sc_i) + \frac{\beta}{m^2\sqrt{c}} \sum_{i=1}^{C}\sum_{k=1}^{N} (u_{ik}^{m} \log u_{ik}^{m} - u_{ik}^{m}) \quad \text{for } k = 1,\ldots,N \text{ and } 0 \le u_{ik} \le 1. \tag{3.48}$$

subject to the constraint: $m > 1$ for $k = 1$ to N where m is weighting exponents ; $0 \le u_{ik} \le 1$ where u_{ik} is the fuzzy membership values of s_k in class i.

Theorem 4 (sgUPCMed): If $Lev(s_k, sc_i) > 0$ for all i and k, when $m, \eta, k>1$, and S contains $C<N$ distinct string data, then $J_{m,\eta}$ is minimized only if the update equation of u_{ik} is

$$u_{ik} = \exp\left(-\frac{m\sqrt{c}\,Lev(sc_i, s_k)}{\beta}\right). \tag{3.49}$$

where β is a positive parameter, n is number of all strings, Yang and Wu defined β as the sample co-variance [15], the calculation of β value is based on sample co-variance [15] which differs from the β of UPCM by using the Levenshtein distance instead of Euclidean distance. Our β can be calculated as following:

$$\beta = \frac{\sum_{k=1}^{n} Lev(Med, s_k)}{n} \text{ with } Med = \arg\min_{j \in S} \sum_{k=1}^{N} Lev(s_j, s_k) \quad \text{for } 1 \le i \le C \qquad (3.50)$$

Proof we use the Lagrange multiplier theorem for proof the equation 3.43. It is obtained by solving the reduced problem

$$\min \left\{ L_i(\mathbf{U}, \lambda) = J_m^{ik}(\mathbf{U}) = u_{ik}^m Lev(s_k, sc_i) + \frac{\beta}{m^2 \sqrt{c}} \sum_{i=1}^{C} \sum_{k=1}^{N} (u_{ik}^m \log u_{ik}^m - u_{ik}^m) \right\} \text{ with } \mathbf{V} \text{ fixed for}$$

the k-th column of $\mathbf{U}$. The derivative of $L_i(\mathbf{U},\lambda)$ with respect to u_{ik} and setting it to zero leads to:

$$\frac{\partial L_i(\mathbf{U}, \lambda)}{\partial u_{ik}} = m(u_{ik})^{m-1} Lev(s_k, sc_i) + \frac{\beta}{m^2 \sqrt{c}} (m u_{ik}^{m-1} m \ln u_{ik} - m u_{ik}^{m-1}) = 0$$

$$\frac{m\sqrt{c} Lev(s_k, sc_i) + \beta u_{ik}^{m-1} \ln u_{ik}}{\sqrt{c}} = 0 \qquad (3.51)$$

$$u_{ik} = \exp\left(-\frac{m\sqrt{c} Lev(sc_i, s_k)}{\beta} \right)$$

Hence, the update equation of u_{ik} will be (3.49). ■

The algorithm of the modified fuzzy median string for sgUPCMed is similar to sgFCMed. To compute each string prototype, we use modified fuzzy median string method same as in equations 3.14 and 3.15 of sgFCMed.

Hence, the summarized sgUPCMed algorithm is as follows:

Store N unlabeled finite strings $S = \{s_k; k = 1,..., N\}$

Initialize string prototypes for all C classes by using equation (3.12) for single prototype clustering and randomly choose string prototypes as the initial cluster centers for multi-prototypes clustering.

Set m

Compute β using equation (3.50)

Do {

Compute Levenshtein distance between input string j and cluster prototype i ($Lev(s_j, sc_i)$)

Update membership value using equation (3.49)

Update center string of each cluster i (sc_i) using equation (3.14) and (3.15)

} **Until** (Maximum number of iterations or Levenshtein distance between cluster center of previous iteration and current iteration less than stopping criteria)

The overall computational time of the sgUPCMed algorithm is $O((l^2 \cdot N^2)+(l^2 \cdot N^2)+(l^3 \cdot c \cdot |\sum|))$ for each global iteration.

3.1.5 String Grammar Unsupervised Possibilistic Fuzzy C-Medians

Now, we are ready to modified UPFCM [16] to cope with the syntactic data set which called the string grammar unsupervised possibilistic-fuzzy C-Medians (sgUPFCMed). This leads to the following optimization problem of sgUPFCMed:

$$J_{m,\eta}(\mathbf{U},\mathbf{T},\mathbf{V};S) = \sum_{k=1}^{N}\sum_{i=1}^{C}(au_{ik}^{m} + bt_{ik}^{\eta})Lev(s_k, sc_i) + \frac{\beta}{\eta^2\sqrt{c}}\sum_{i=1}^{C}\sum_{k=1}^{N}(t_{ik}^{\eta}\log t_{ik}^{\eta} - t_{ik}^{\eta}), \quad (3.52)$$

subject to the constraint: $m > 1, \eta > 1$ for $k = 1$ to N where m and η is weighting exponents ; $0 \le u_{ik}, t_{ik} \le 1$ and $\sum_{i=1}^{C} u_{ik} = 1$ where u_{ik} is the fuzzy membership values of s_k in class i and t_{ik} is the possibilistic values of s_k in class i; the constraint $a > 0, b > 0$ define the relative importance of fuzzy membership and possibilistic values in the objective function.

Theorem 5 (sgUPFCMed): If $Lev(s_k, sc_i) > 0$ for all i and k, when $m, \eta, k>1$, and S contains $C<N$ distinct string data, then $J_{m,\eta}$ is minimized only if the update equation of u_{ik} is

$$u_{ik} = \frac{1}{\sum_{j=1}^{C}\left(\frac{Lev(sc_i, s_k)}{Lev(sc_j, s_k)}\right)^{1/(m-1)}} \Bigg\} \tag{3.53}$$

and then the update equation of t_{ik} is

$$t_{ik} = \exp\left(-\frac{b\eta\sqrt{c}Lev(sc_i, s_k)}{\beta}\right). \tag{3.54}$$

where β is a positive parameter that can be calculated using equation 3.50.

Proof We use the Lagrange multiplier theorem. Equation 3.53 and 3.54 is obtained by solving the reduced problem $\min_{\mathbf{U}\in M_{fcn}}\left\{J^k_{m,\eta}(\mathbf{U}) = \sum_{i=1}^{C}\left(au^m_{ik} + bt^\eta_{ik}\right)Lev(s_k, sc_i)\right\}$ with **T** and **V** fixed for the k-th column of **U**. The proof of the equation 3.53 is similar to that in theorem 3 of sgPFCMed.

Similarly, **U** and **V** are fixed for the i-th row **T**, equation 3.54 is proved by solving the problem

$$\min\left\{L_i(\mathbf{T},\lambda) = J^{ik}_{m,\eta}(\mathbf{T}) = \left(au^m_{ik} + bt^\eta_{ik}\right)Lev(s_k, sc_i) + \frac{\beta}{\eta^2\sqrt{c}}\sum_{i=1}^{C}\sum_{k=1}^{N}(t^\eta_{ik}\log t^\eta_{ik} - t^\eta_{ik})\right\}.$$

The derivative of $L_i(\mathbf{T},\lambda)$ with respect to t_{ik} and setting it to zero leads to:

$$\frac{\partial L_i(\mathbf{T},\lambda)}{\partial t_{ik}} = b\eta\left(t_{ik}\right)^{\eta-1}Lev(s_k, sc_i) + \frac{\beta}{\eta^2\sqrt{c}}(\eta t^{\eta-1}_{ik}\eta\ln t_{ik} - \eta t^{\eta-1}_{ik}) = 0$$

$$\frac{\eta\sqrt{c}Lev(s_k, sc_i) + \beta {t_{ik}}^{\eta-1}\ln t_{ik}}{\sqrt{c}} = 0 \quad . \tag{3.55}$$

$$t_{ik} = \exp\left(-\frac{\eta\sqrt{c}Lev(sc_i, s_k)}{\beta}\right)$$

Hence, the update equation of t_{ik} will be

$$t_{ik} = \exp\left(-\frac{b\eta\sqrt{c}Lev\left(sc_i, s_k\right)}{\beta} \right). \tag{3.56}$$

■

The algorithm of the modified fuzzy median string for sgUPFCMed is similar to sgPFCMed. To compute each string prototype, we use modified fuzzy median string method same as in equations 3.43 and 3.44 of sgPFCMed.

Hence, the summarized sgUPFCMed algorithm is as follows:

Store N unlabeled finite strings $S = \{s_k; k = 1,..., N\}$

Initialize string prototypes for all C classes by using equation (3.12) for single prototype clustering and randomly choose string prototypes as the initial cluster centers for multi-prototypes clustering.
Set m, η, a, b
Compute β using equation (3.50)

Do {
Compute Levenshtein distance between input string j and cluster prototype i ($Lev(s_j,sc_i)$)
Update membership value using equation (3.53)
Update typicality value using equation (3.54)
Update center string of each cluster i (sc_i) using equation (3.43) and (3.44)
} **Until** (Maximum number of iterations or Levenshtein distance between cluster center of previous iteration and current iteration less than stopping criteria)

The overall computational time of the sgUPFCMed algorithm is $O((l^2 \cdot N^2)+(l^2 \cdot N^2)+(l^3 \cdot c \cdot |\sum|))$ for each global iteration.

3.2 Illustration of String Grammar Fuzzy Clustering

We illustrate our five string grammar fuzzy clustering algorithms with 2-classes sample of Thai printed numeric data set with 10 samples in each class and three outliers shown in Figure 3.1. The parameters setting is shown in Table 3.1

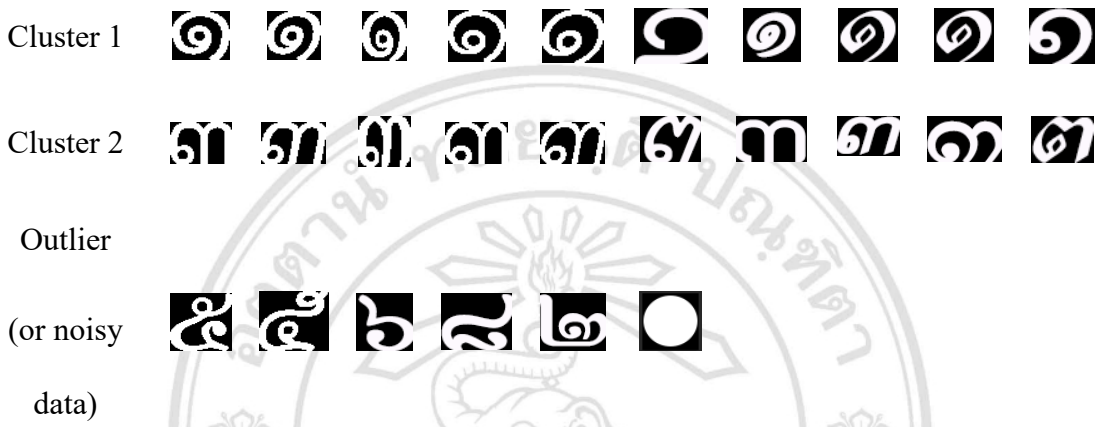


Figure 3.1 Examples of 2-class Thai printed numeric data set.

Table 3.1 Parameter setting of our algorithms for Thai printed numeric data set.

parameter	sgFCMed	sgFPCMed	sgPFCMed	sgUPCMed	sgUPFCMed
m	2	2	2	2	2
η	-	2	2	-	2
γ	-	-	2	-	-
a	-	-	1	-	1
b	-	-	1	-	1
stopping criteria	0.1	0.1	0.1	0.1	0.1
maximum number of iterations	50	50	50	50	50

To create a string from each image, we first cropped each image to only cover the object and then resize it to 50 pixels in height while the width is scaled according to the original aspect ratio. The boundary or the contour of the image was extracted using

the Moore-neighbor tracing algorithm [50]. After that, the boundary image was encoded using the 8-directional chain code [51]. Finally, the differential chain code [51] was used as the sequence of the image string.

In all experiments, we initial prototype of each class using its median string. After, our algorithms are converged, the final membership values and typicality values of each string are shown in Table 3.2. It shows that sgFCMed, sgFPCMed, sgPFCMed, sgUPCMed and sgUPFCMed results in the same max-membership hard partition. The relative ordering of string (in terms of membership values) also remains the same. This is because the cluster centers from all algorithms are almost located at the same location.

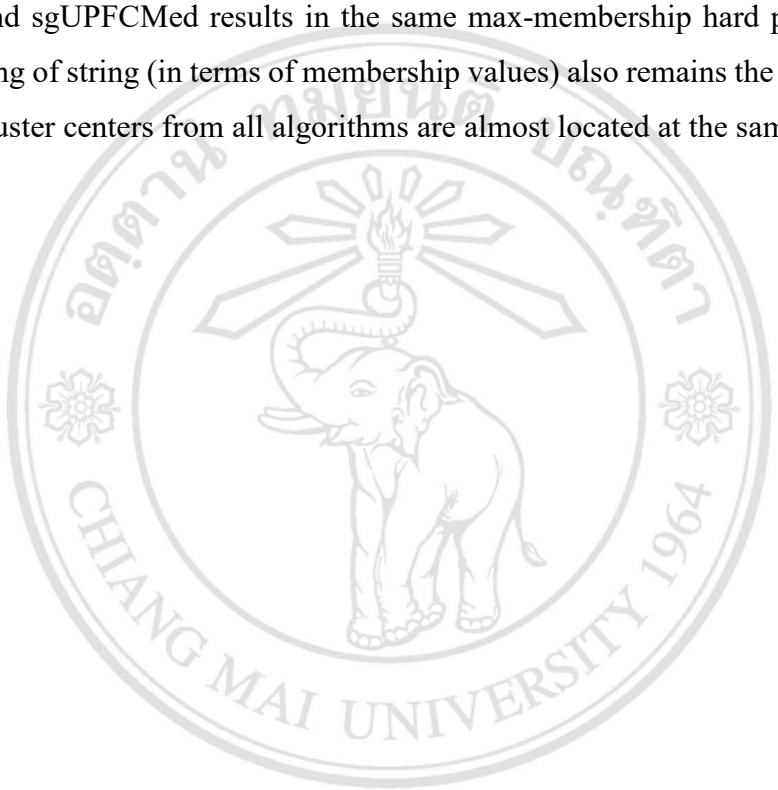


Table 3.2 The final membership from our algorithms for Thai printed numeric data set.

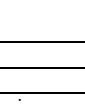
		sgFCMed		sgFPCMed				sgPFCMed				sgUPCMed		sgUPFCMed			
		Membership values		Membership values		Possibility values		Membership values		Possibility values		Possibility values		Membership values		Possibility values	
No.	image	u_{1k}	u_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}	t_{1k}	t_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}
1		0.9912	0.0088	0.9912	0.0088	0.6038	0.0044	0.9912	0.0088	0.9534	0.1124	0.9778	0.00062	0.9971	0.0029	0.9985	0.00069
2		0.8436	0.1564	0.8436	0.1564	0.1536	0.0043	0.8436	0.1564	0.8560	0.4534	0.8798	0.00029	0.7856	0.2144	0.8976	0.00027
3		0.8454	0.1546	0.8454	0.1546	0.0345	0.0046	0.8454	0.1546	0.8124	0.4779	0.8433	0.00028	0.7436	0.2564	0.8543	0.00026
4		0.7238	0.2762	0.7238	0.2762	0.0234	0.0044	0.7238	0.2762	0.8798	0.3673	0.8894	0.00027	0.7842	0.2158	0.8945	0.00025
5		0.7953	0.2047	0.7953	0.2047	0.0114	0.0042	0.7953	0.2047	0.8244	0.4689	0.8139	0.00033	0.7578	0.2422	0.8543	0.00031
6		0.6135	0.3865	0.6135	0.3865	0.0105	0.0032	0.6135	0.3865	0.7053	0.4252	0.7934	0.00022	0.6154	0.3846	0.8145	0.00019
7		0.7452	0.2548	0.7452	0.2548	0.0342	0.0036	0.7452	0.2548	0.7683	0.3052	0.8148	0.00025	0.7424	0.2576	0.8204	0.00023
8		0.7344	0.2656	0.7344	0.2656	0.045	0.0034	0.7344	0.2656	0.7974	0.3125	0.8425	0.00029	0.7364	0.2636	0.8498	0.00025
9		0.7043	0.2957	0.7043	0.2957	0.0332	0.0038	0.7043	0.2957	0.7453	0.3476	0.8198	0.00026	0.7078	0.2922	0.8236	0.00022
10		0.6899	0.3101	0.6899	0.3101	0.0241	0.0041	0.6899	0.3101	0.7367	0.4074	0.7969	0.00021	0.6949	0.3051	0.8035	0.00018
11		0.1547	0.8453	0.1547	0.8453	0.0031	0.2125	0.1547	0.8453	0.1670	0.7983	0.00061	0.8432	0.2146	0.7854	0.00060	0.8532
12		0.2654	0.7346	0.2654	0.7346	0.0029	0.0118	0.2654	0.7346	0.1879	0.7233	0.00060	0.7854	0.2864	0.7136	0.00050	0.7956
13		0.2166	0.7834	0.2166	0.7834	0.0034	0.0125	0.2166	0.7834	0.1806	0.7809	0.00064	0.8235	0.2567	0.7433	0.00058	0.8644
14		0.0136	0.9864	0.0136	0.9864	0.0032	0.6328	0.0136	0.9864	0.0112	0.9487	0.00061	0.9643	0.0037	0.9963	0.00056	0.9896
X15		0.2745	0.7255	0.2745	0.7255	0.0029	0.0143	0.2745	0.7255	0.1699	0.8751	0.00068	0.9056	0.2532	0.7468	0.00063	0.9145
16		0.3246	0.6754	0.3246	0.6754	0.0018	0.0115	0.3246	0.6754	0.4678	0.7235	0.00060	0.7864	0.3236	0.6764	0.00058	0.7889
17		0.3759	0.6241	0.3759	0.6241	0.0009	0.0131	0.3759	0.6241	0.4984	0.6973	0.00053	0.7043	0.3742	0.6258	0.00040	0.7088

Table 3.2 The final membership from our algorithms for Thai printed numeric data set. (continue)

		sgFCMed		sgFPCMed				sgPFCMed				sgUPCMed		sgUPFCMed			
		Membership values		Membership values		Possibility values		Membership values		Possibility values		Possibility values		Membership values		Possibility values	
No.	image	u_{1k}	u_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}	t_{1k}	t_{2k}	u_{1k}	u_{2k}	t_{1k}	t_{2k}
18		0.3179	0.6821	0.3179	0.6821	0.0011	0.0113	0.3179	0.6821	0.4235	0.7124	0.00061	0.7345	0.3183	0.6817	0.00058	0.7532
19		0.3779	0.6221	0.3779	0.6221	0.0007	0.0126	0.3779	0.6221	0.4976	0.7011	0.00063	0.7285	0.3781	0.6219	0.00060	0.7467
20		0.3858	0.6142	0.3858	0.6142	0.0011	0.0167	0.3858	0.6142	0.5832	0.6996	0.00059	0.7087	0.3864	0.6136	0.00055	0.7356
21		0.5703	0.4297	0.5703	0.4297	0.0007	0.0036	0.5703	0.4297	0.0892	0.0756	0.00014	0.00012	0.5805	0.4195	0.00009	0.00012
22		0.5325	0.4675	0.5325	0.4675	0.0009	0.0026	0.5325	0.4675	0.0766	0.0795	0.00016	0.00022	0.5084	0.4916	0.00006	0.00020
23		0.5988	0.4012	0.5988	0.4012	0.0012	0.0010	0.5988	0.4012	0.0812	0.0734	0.00013	0.00011	0.5991	0.4009	0.00010	0.00009
24		0.5443	0.4557	0.5443	0.4557	0.0010	0.0015	0.5443	0.4557	0.0763	0.0780	0.00012	0.00013	0.5444	0.4556	0.00009	0.00011
25		0.4563	0.5437	0.4563	0.5437	0.0011	0.0021	0.4563	0.5437	0.0697	0.0765	0.00012	0.00011	0.4523	0.5477	0.00010	0.00011
26		0.5352	0.4648	0.5320	0.4680	0.0003	0.0001	0.5320	0.4680	0.0045	0.0032	0.00009	0.00007	0.5310	0.4690	0.00006	0.00004

When we consider the possibity value or typical values of sgFPCMed, sgPFCMed, sgUPCMed and sgUPFCMed. The possibility values of samples for all four algorithms in cluster 1 $(t_{1k})_{1\le k\le 5}$ are far larger than possibility values of samples in cluster 2 $(t_{2k})_{1\le k\le 5}$. The sample numbers 1 to 10 are more possibility to cluster 1 than cluster 2. While those in cluster 1 $(t_{1k})_{6\le k\le 10}$ are smaller than $(t_{2k})_{6\le k\le 10}$ notably. Again, sample numbers 11 to 20 are more possibility to cluster 2 than cluster 1. When we consider noisy data with outlier, the membership values of the sgFCMed cannot detect outliers 21, 22, 23, 24, 25 and 26. Whereas the possibility values (or typical value) from sgFPCMed, sgPFCMed, sgUPCMed and sgUPFCMed provides a more informative description of the data than sgFCMed, since it provides roughly the same membership information but also shows, via the possibilities, for example, that number 21, 22, 23, 24, 25 and 26 are small possibility than others for either cluster. This indicates that six samples are not in the 2 clusters mentioned above. Moreover, the possibility value also gives more detail about outlier, that number 26 is far from prototype than numbers 21, 22, 23, 24 and 25, but not in the sgFCMed. Since sgFCMed model consider only membership value. Hence the sgFCMed has difficulty in handling outlier. The sgFPCMed, sgPFCMed, sgUPCMed and sgUPFCMed can solve the problem of sgFCMed, these algorithms consider the clustering analysis from the viewpoint of possibility theory. The possibility or typicality values will be low for outliers. Although sgFPCMed is created for solving noisy data, it seems to be worse than sgPFCMed in large dataset because of the constraint $\sum t_{ik} = 1$.

We need to have this constraint in sgPFCMed because the columns and rows of the typicality matrix are independent of each other, If the initialization of each row is not sufficiently distinct, coincident clusters may result. The sgPFCMed can improve the result of clustering in domain of string by relaxing the row of sum constraint of sgFPCMed (sum of the typicalities over all data points to a particular cluster is 1). For improving the classification accuracy result, sgUPCMed and sgUPFCMed algorithms use the exponential functions to describe the degree of belonging based on the validity indexes PC and PE. The possibility value is very small and tends to be zero for the samples that far away from each other such as sample number 26. Hence, both algorithms can detect the outliers of dataset as well.