APPENDIX (A)

FFT_Algorithm: Forming the distance Matrix based on FFT

Sets input includes $m \times n$ instances, where m represents the number of websites and n denotes the number of instances each website

Sets Input: $T = \{t_i\}$ that $t_i = p_{i_1}, p_{i_2}, \cdots p_{i_l}$ $\quad 1 \le i \le m \times n$ and $34 \le n \le 40$

Output: $PDM_{m.n \times m.n}$ *and* $DM_{m.n \times m}$

$PDM = \{sd_{ij}\}$ $1 \le i, j \le m \times n$ *where* sd_{ij} *is calculated by relation*(3)

$DM = \{sd'_{ij}\}$ $1 \le i \le m \times n$ *and* $1 \le j \le m$ *where* sd'_{ij} *is calculated by relation*(6).

If $r = [1: m \times n]$ *so* t_r *denotes the* r^{th} *instance that its actual length is* l_r .

1. *Actual length of each* t_r *fixed to Length L, by padding (0) or truncating the instance.*
2. *Map each instance to new domain by apply real Function on the FFT coefficients.*

$t_r^F = real(FFT(t_r, L))$.

3. *Create the primary Distance Map* ($DM_{m.n \times m.n}$)

$for\ i = 1 : m \times n$

$for\ j = 1 : m \times n$

$PDM(i,j) = sd\left(t_i^F, t_j^F\right)$; *% using the relation (3)*

4. *Create the Distance Map* ($DM_{m.n \times m}$) *as follows:*

$for\ i = 1 : m \times n$

$for\ j = 1 : m$

$DM(i,j) = sd\left(t_i^F, symbol_j\right)$ % $symbol_j$ *is calculated by relation*(6)

Return (PDM, DM)