APPENDIX B: THE PSEUDO-CODES OF THE PROPOSED SOLUTION

This appendix contains three pseudo-code snippets outlining the proposed solution's approach. Due to page limitations, these snippets have been included here for reference. These pseudo-code snippets provide a concise overview of the algorithmic processes involved in the proposed solution. Further details and context can be found in the main body of the paper.

Algorithm 1. TFEM Pseudocode

```
$\mathcal{I}nseq(\mathcal{S})$ ← Vector of sequence family
$\mathcal{K}mer$ ← Desired motif length
$\mathcal{M}ax(n)$ ← Maximum desired number of motif candidates
$\mathcal{M}in(\mathcal{S}im\_thr)$ ← Minimum desired motif similarity
$\mathcal{A}minoQ$ ← Primary_Candidates$(\mathcal{I}nseq(\mathcal{S}),\mathcal{M}in(\mathcal{S}im\_thr))$
$i \leftarrow 1$
while $\mathcal{A}minoQ_i \neq NULL$ do
    if $|\mathcal{A}minoQ_i| < kmer$ then $SubCandidates \leftarrow$
        $Extend\_Motif(\mathcal{I}nseq(\mathcal{S}),\mathcal{A}minoQ_i,\mathcal{M}in(\mathcal{S}im\_thr))$
        for each $\gamma$ in $SubCandidates$ do
            $\mathcal{P} \leftarrow$ rear$(\mathcal{A}minoQ)$, $\mu \leftarrow \gamma$
            while $TRUE$ do
                if $\mathcal{A}minoQ_{\mathcal{P}} = \mu$ then $break$
                else if $\mathcal{S}im(\mathcal{P}) > \mathcal{S}im(\mu)$ and $|\mathcal{P}| = |\mu|$
                        then $\mathcal{P} \longleftarrow$ back$(\mathcal{P})$
                else
                        next$(\mu) \longleftarrow$ next$(\mathcal{P})$
                        back$(\mu) \longleftarrow \mathcal{P}$
                        next$(\mathcal{P}) \longleftarrow \mu$
                        if next$(\mu) \neq NULL$
                            then back(next$(\mu)$) $\longleftarrow \mu$
                        break
            end while
        end for
    $i \leftarrow i + 1$
end while
$\mathcal{P} \leftarrow$ rear$(\mathcal{A}minoQ)$
$i \leftarrow 1$
while $(i \leq \mathcal{M}ax(n)$ AND $|\mathcal{P}| = \mathcal{K}mer)$ do
    Print $\mathcal{P}$
    $\mathcal{P} \longleftarrow$ back$(\mathcal{P})$
    $i \leftarrow i + 1$
end while
```

Algorithm 2. Primary_Candidates Function

```
$\mathcal{A}$ ←the ordered list of alphabetic characters
$\mathcal{S}um$ ← a zero initialized vector with the size of $|\mathcal{A}|$
for i = 1 to $|\mathcal{I}nseq(\mathcal{S})|$ do
    Presence ⟵a zero vector with a size of $|\mathcal{A}|$
    j , $\mathcal{C} \longleftarrow 0$
    while j < $|\mathcal{S}_i|$ and $\mathcal{C} \leq |\mathcal{A}|$
        ω ⟵ getalphabet_index$(\mathcal{S}_{[i][j]})$
        if Presence$_\omega$= 0 then
            $\mathcal{C} \longleftarrow \mathcal{C} + 1$
            Presence$_\omega$= 1
        end if
        j ⟵ j+1
    end while
    for j=1 to $|\mathcal{A}|$ do
        Sum$_j$ ⟵ Sum$_j$ + Presence$_j$
    end for
end for
for $i$= 1 to $|\mathcal{A}|$ do
    if Sum$_i$/$|\mathcal{I}nseq(\mathcal{S})| \geq \mathcal{M}in(\mathcal{S}im\_thr)$
        then insert $\mathcal{A}_i$ into $\mathcal{A}minoQ$
end for
return $\mathcal{A}minoQ$
```

Algorithm 3. Extend_Motif Function

```
$\mu$ ⟵ An $\mathcal{A}minoQ$ passed from Algorithm 1
$SubCandidates$ ← An empty vector of the motifs
$\mathcal{A}$ ⟵The ordered list of alphabetic characters
$\mathcal{L}$presence ⟵ Zero initialized matrix by the size of
$|\mathcal{I}nseq(\mathcal{S})| \times |\mathcal{A}|$
$\mathcal{R}$presence ⟵ Zero initialized matrix by the size of
$|\mathcal{I}nseq(\mathcal{S})| \times |\mathcal{A}|$
for $i = 1$ to $|\mathcal{I}nseq(\mathcal{S})|$ do
    $\upsilon$ ⟵ All indexes of $\mu$ within $\mathcal{S}_i$
    for j = 1 to $|\upsilon|$ do
        if $\upsilon_j - 1 > 0$ then
            $k$ ⟵ The position of $\mathcal{S}_{\lfloor i \rfloor \lfloor \upsilon_j - 1 \rfloor}$ in $\mathcal{A}$
            $\mathcal{L}$presence$_{[i][k]}$ ⟵ 1
        end if
        if $\upsilon_j + |\mu| \leq |\mathcal{S}_i|$ then
            $k$ ⟵ find the position of $\mathcal{S}_{[i][\upsilon_j + |\mu|]}$ in $\mathcal{A}$
            $\mathcal{R}$presence$_{[i][k]}$ ⟵ 1
        end if
    end for
end for
for i=1 to $|\mathcal{A}|$ do
    $\mathcal{L}$Sum, $\mathcal{R}$Sum ⟵ 0
    for j=1 to $|\mathcal{I}nseq(\mathcal{S})|$ do
        $\mathcal{L}$Sum ⟵ $\mathcal{L}$Sum + $\mathcal{L}$presence$_{[j][i]}$
        $\mathcal{R}Sum$ ⟵ $\mathcal{R}Sum + \mathcal{R}presence_{[j][i]}$
    end for
    $\mathcal{L}Sim$ ⟵ $\mathcal{L}$Sum/$|\mathcal{I}nseq(\mathcal{S})|$
    $\mathcal{R}Sim$ ⟵ $\mathcal{R}$Sum/$|\mathcal{I}nseq(\mathcal{S})|$
    if $\mathcal{L}Sim \geq \mathcal{M}in(\mathcal{S}im\_thr)$ then
        Insert candidate $(\mathcal{A}_i\mu)$ into $SubCandidates$
    if $\mathcal{R}Sim \geq \mathcal{M}in(\mathcal{S}im\_thr)$ then
        Insert candidate $(\mu\mathcal{A}_i)$ into $SubCandidates$
end for
return $SubCandidates$
```