**Flowchart for a novel SMS filtering approach**

**A. Perform a Pre-processing operation**

Start

Read in the SMS Message

Remove every unwanted character from the input SMS message

Convert the message to UTF-8 values of the characters in the text

Output the UTF-8 values of the SMS message

B

**B. Perform a 1D-TP transformation**

B

Get the UTF-8 values of the SMS message as an input

Set the initial value for the total number of utilized neighbours of characters, i.e. P

Set the initial value for the threshold parameter, i.e. Beta(B)

Arrange the input UTF-8 values of the SMS message into a cluster of P+1 characters, i.e. C

For each Clusters, Ck

Get the center or middle character of the Cluster, i.e. Pc

Get the list of Left neighbour characters of Pc by extract all the P/2 characters that comes before Pc in the given cluster as Pl

Get the list of Right neighbour characters of Pc by extract all the P/2 characters that comes after Pc in the given cluster as Pr

Merge the two lists (Pl and Pr) together into a new list as Q

For every ith character of the list Pk

Is

(Pc <= Qi + B AND Pc >= Qi - B)?

Is

(Pc > Qi + B)?

Set TPi = 1

Set TPi = 0

Set TPi = -1

Y

Y

N

N

C

D

E

C

Generate a Bit value (1) for every positive value of TP and a Bit value (0) for every negative value of TP, then add this to the left list BLPi

Generate a Bit value (0) for every positive value of TP and a Bit value (1) for every negative value of TP, then add this to the right list BRPi

Is

i = size of list Q?

Y

N

D

Generate the decimal value of the string of Bit values generated in the left list BLPk , and set this as the Upper feature value as UP = decimal(BLPk )

Generate the decimal value of the string of Bit values generated in the right list BRPi , and set this as the Lower feature value as LP = decimal(BRPk )

Output the two generated values (UP and LP) as the 1D-TP transformation of the input SMS message.

Is

All Cluster processed?

Y

N

E

F

**C. Perform a 1D-TP Histogram**

Generate a stream of bit values of size P (total number of utilized neighbours of characters)

F

Get the result of 1D-TP transformation as the input

For each of the 2 list UP and LP, generate a list of frequently occurred items into another 2 list HU and HL.

Generate the Histogram value (i.e. how often each of the patterns appear in its corresponding signal) of the new lists HU and HL

G

**D. Perform an Simulated Annealing Optimization**

G

Set initial value for the maximum and minimum possible parameter as Tmax and Tmin

Set initial value for initial value for the maximum possible Iteration, as MaxIt

Set the initial value for the total number of utilized neighbours of characters, i.e. P as X1

Set the initial value for the threshold parameter, i.e. Beta(B) as X2

Start a global Iteration, at T = Tmax and alpha = random(0, 1)

Start a new Local Iteration, at i = 1

Compute the cost function (1D-TP transformation of a given SMS message with the current values of parameters X1 and X2) as, E = Cost(X1, X2)

Generate a random neighboring solution of X1 and X2 as Next\_X1 and Next\_X2 respectively

Compute the cost function (1D-TP transformation of a given SMS message with the next values of parameters Next\_X1 and Next\_X2) as, E\_Next = Cost(Next\_X1, Next\_X2)

Evaluate the change in cost as delta\_E = E\_Next – E

set X1 = Next\_X1 and set X2 = Next\_X2

Y

N

H

J

Is

delta\_E < acceptance\_treshold\_value?

Is

Exp(delta\_E, T) > random(0,1)?

Y

L

Increment the value of i by 1

H

Update the value of T as, set T = alpha \* T

Is

I <= MaxIt?

Y

N

K

Output the best solution obtained for X1 and X2 as P and B respectively

J

Is

T >= Tmin?

N

L

Y

**E. Perform a Classification Operation**

K

Get the best solution obtained from K above (i.e. the optimal value of 1D-TP Histogram) as the input

Choose any of the classification methods (Multinomial Naïve Bayes, Bayesian network, random forest (RF), k nearest neighbors, radial basis feed forward neural network, relevance vector machine and artificial neural network)to classify the SMS message as ham or spam message.

Get the report of each classifier

Stop