**Algorithm for a novel SMS spam filtering approach**

1. Read in the SMS Message

2. Perform a Pre-processing operation

3. Perform a 1D-TP transformation

4. Perform a 1D-TP histogram

5. Perform a Simulated Annealing Optimization

6. Perform a classification operation

Breaking down each step of the algorithm as follows:

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

1. Remove every unwanted character from the input SMS message

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

3. Output the UTF-8 values of the SMS message

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

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

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

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

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

5. For each Clusters, Ck repeat step 5 to step 16 until all cluster is processed.

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

7. 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

8. 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

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

10. For every ith character of the list Pk, repeat step 11 to step 13 until i = size of list Q

11. Compute the Tenary Code(TP) by carrying out comparison based on the following condition:

If (Pc > Qi + B), then Set TPi = 1

else if (Pc <= Qi + B AND Pc >= Qi - B), then Set TPi = 0

else if (Pc < Qi - B), then Set TPi = -1

end if

12. 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

13. 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

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

15. 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(BRPi ).

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

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

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

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

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

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

**D. Perform a Simulated Annealing Optimization**

1. Set initial value for the maximum possible parameter called the "temperature", as Tmax
2. Set initial value for the minimum possible parameter called the "temperature", as Tmin
3. Set initial value for the maximum possible Iteration, as MaxIt
4. Set the initial value for the total number of utilized neighbours of characters, i.e. P as X1
5. Set the initial value for the threshold parameter, i.e. Beta(B) as X2
6. Start a global Iteration, at T = Tmax and alpha = random(0, 1)
7. Start a new Local Iteration, at i = 1
8. 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)
9. Generate a random neighboring solution of X1 and X2 as Next\_X1 and Next\_X2 respectively
10. 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)
11. Evaluate the change in cost as delta\_E = E\_Next – E
12. if (delta\_E < acceptance\_treshold\_value) then, move to the next solution by accepting the new values of X1 and X2 as, set X1 = Next\_X1 and set X2 = Next\_X2, goto step 14 otherwise goto step 13
13. if (Exp(delta\_E, T) > random(0,1)) then, move to the next solution by accepting the new values of X1 and X2 as, set X1 = Next\_X1 and set X2 = Next\_X2
14. Increment the value of i by 1
15. Check if the current iteration is the last iteration, i.e. is i <= MaxIt then goto step 8 otherwise continue on the next step
16. Update the value of T as, set T = alpha \* T
17. Check if the current T is out of the region(Tmax, Tmin), i.e. is T >= Tmin then goto step 7 otherwise continue on the next step
18. Output the best solution obtained for X1 and X2 as P and B respectively.

**E. Perform a Classification Operation**

1. Get result of 1D-TP Histogram as the input

2. Choose any of the classification methods (Naïve Bayes, Bayesian network, random forest (RF), k nearest neighbors, and artificial neural network)to classify the SMS message as ham or spam message.