Report

Objective:

To develop notification system from SMS.

Complexity:

Computation capability: I had personal laptop with 4GB ram and no GPU.
Huge Dataset: 1 cr points, failed to generalize.
Different patterns, hence need more time to study and draw out patterns

Trails and failures:

Use of **dependency parse tree** to fetch important tags: To fetch relevant information out of the given corpus, first I tried to use POS tagging and dependency parse tree. The algorithms in most cases tried to separate out dates and values, which in most cases resulted in ambiguity and hence it failed to generalize at large scale.

Use of **sentence similarity**: Major strategy was to fetch out relevant SMS and then try to match the similarity of each SMS with the given one. Major obstruction was the computation power and memory need. For tf-idf to work we need to convert each SMS to vectorize format and that require either pre-trained word2vec model or very large RAM size. Clearly, this approach gave less accuracy and took more time, hence it can be deployed to mobile devices.

Use to **LSTM** to classify useful/not useful SMS. My approach was to train a model to classify as a favorable and unfavorable case. The pattern suggests that if sender name's last six places contains only numerical values, then it is a promotional SMS(reference). I exploited this nature to create training examples out of given dataset. But the given model also had similar time and computation constraints.

Final model:

Using regex and simple python to find dates and amount and will use the bag of words to classify categories. As we refine more on finding the correct regex patterns to find money and dates, model accuracy improves. Still in developing form but this approach is clearly scalable and easily deployed to mobile devices. This approach process over 1 cr SMS in less than 3 seconds.

Declaration: The whole work is solely done by me and there is no plagiarism in the project.