

## **Declaration**

We , Samyak, Priyam and Atharv B.tech students of **CSE (Evening) of 7<sup>th</sup> Semester** of the Maharaja Surajmal Institute Of Technology, New Delhi declare that the Minor project Report entitled “**Text Classification using Active Learning**” is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other institute for the award of any other degree.

We hereby certify that no part of this assignment/product has been copied from any other student's work or from any other source except where due acknowledgement is made in the assignment.

No part of this assignment/product has been written/produced for us by another person except where such collaboration has been authorised by the subject lecturer/tutor concerned.

We are aware that this work may be reproduced and submitted to plagiarism detection software programs for the purpose of detecting possible plagiarism. We hold a copy of this assignment that we can produce if the original is lost or damaged.

Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and given their details in the references.

**Samyak Jain - 41896392717**

**Priyam Gupta - 42296302717**

**Atharv Mittal - 41796302717**

**Computer Science & Engineering (Eve)**

**7<sup>th</sup> Semester**

**Place - Maharaja Surajmal Institute of Technology**

**December - 2020**

## **Acknowledgement**

In the present world of Competition there is a race of existence in which those who are having will to come forward succeed. Project is like a bridge between theoretical and practical work . With this Willing we joined this particular project.

This project has been a great learning experience for us and We would like to express our sincere gratitude to all the people to guide us throughout the project and without the valuable guidance and suggestions this project would not be completed successfully. We would like to extend my sincere thanks to all of them.

We also take this opportunity to express a deep sense of gratitude to our mentor **Dr. Adeel Hashmi** (Head of Department , Department of Computer Science & Engineering ) & **Ms. Neeti Sangwan** (Proctor, Department of Computer Science & Engineering 7<sup>th</sup> Semester Evening Shift) for their cordial support, valuable information and guidance which helped us in completing the task through various stages.

We would like to thank all the respondents whom we interacted during my project for their help and suggestions.

Last, but not the least, An honorable mention goes to our family and friends for their understanding , support and suggestions is completing this project

# **Table of Contents**

Declaration.....	II
Certificate.....	III
Acknowledgement.....	IV
Table of Contents.....	V
List of Figures.....	VII
Abstract.....	IX
Chapter - 1 : Introduction.....	1
1.1 Introduction.....	2
1.2 Text Classification.....	3
1.3 Applications of Text Classification.....	4
1.3.1 Applications and use cases.....	5
1.4 Types of Learning.....	7
1.4.1 Unsupervised V/S Supervised Learning.....	7
1.4.2 Active vs. Passive Learning.....	8
1.5 Active Learning.....	9
1.5.1 Varieties of Active Learning.....	10
Chapter - 2 : Literature Surveys.....	12
2.1 Major Research on Active Learning.....	13
2.2 Major Approaches.....	13
2.3 Characteristics of Active Learning Algorithms.....	13
2.3.1 Ranker consideration.....	14
2.3.2 Computational complexity.....	16
2.3.3 Density.....	17
2.3.4 Diversity.....	18
2.3.5 Close to boundary.....	19
2.3.6 Far from boundary.....	20
2.3.7 Probabilistic or uncertainty of ranker.....	21
2.3.8 Myopic.....	21
2.4 Conclusion.....	23
Chapter - 3 : Proposed Solution.....	24
3.1 Problem Statement.....	25
3.2 Methodology.....	26
3.2.1 Flowcharts and diagrams.....	26
3.2.2 Explanation.....	28
a) Scenarios of Active Learning.....	28

b) Query Strategies.....	30
3.2.3 Approach.....	32
3.2.4 Text Preprocessing.....	35
Chapter - 4 : Implementation.....	36
4.1 Technologies used.....	37
4.2 Introduction to Python.....	37
4.2.1 Features of Python.....	38
4.2.2 Python Libraries.....	40
4.3 Introduction to Jupyter Notebook.....	43
4.3.1 Launching Jupyter Notebook.....	43
4.4 Deep Learning.....	47
4.4.1 Introduction to Neural Networks.....	47
4.4.2 How Neural Network works.....	48
4.4.3 Multilayer Neural Network.....	48
4.4.4 How Neural Network Learns.....	49
4.5 Active Learning.....	50
4.5.1 Introduction.....	50
4.5.2 Applications and Modern Research into Active Learning.....	52
4.5.3 Models and Algorithms.....	53
a) Random Forest Algorithm.....	53
b) Multinomial Naive Bayes.....	55
c) Support Vector Machine.....	56
d) Logistic Regression .....	58
4.6 Datasets.....	59
4.7 Screenshots.....	61
4.7.1 Code Implementation.....	61
4.7.2 Outputs.....	66
Chapter - 5 : Results.....	69
5.1 Classification Matrix.....	71
5.2 Comparative Results.....	71
5.3 Confusion Matrix.....	72
Chapter - 6 : Conclusion and Future Scope.....	73
6.1 Conclusion.....	74
6.2 Future Scope.....	75
6.3 Impact On Society.....	76
References.....	77

## List of Figures

Figure Number	Name Of Figure	Page Number
1.1	Text Classification Types	03
1.2	Content Tagging	05
1.3	CRM Tasks	05
1.4	Content SEO	06
1.5	Emergency Response System	06
1.6	Product Marketing	07
1.7	Different Organizations	07
1.8	Active Learner	10
1.9	Active Learning Process	11
2.1	Summary of characteristics of Active learning algorithms [1]	19
2.2	Summary of characteristics of active learning algorithms [2]	22
2.3	Different Algorithms	23
3.1	Active Learning Loop	26
3.2	Data Flow Diagram	26
3.3	Flow Chart for AL	27
3.4	AL Process	28
3.5	Pool-based Scenario	29
3.6	Stream-based Scenario	29
3.7	Member Query Synthesis	29
3.8	Query Strategies for AL	31
3.9	TF-IDF Vectorizer	35
4.1	Python Logo	39
4.2	Scikit-learn Logo	40
4.3	Keras Logo	41
4.4	Pandas Logo	42
4.5	Launching Notebook	43

4.6	Interface of Jupyter Notebook	44
4.7	Creating New Notebook	44
4.8	Running Notebooks	45
4.9	Cell in Notebook	45
4.10	Running Cell	45
4.11	MarkDowns	46
4.12	Toolbar	46
4.13	File Menu	46
4.14	Neural N/w with 2 hidden layers	47
4.15	Logistic Regression with 1 feature	48
4.16	Sigmoid output of Neuron	48
4.17	Multilayer Neural Network	48
4.18	Matrix representation of Neural Networks	49
4.19	Training Process	49
4.20	Minimizing Loss	50
4.21	Graph plot examples	51
4.22	Diff. b/w Active and Reinforcement Learning	52
4.23	Feature Trees	53
4.24	Features and Hyperplanes	56
4.25	Hyperplanes	57
4.26	Support Vectors	57
4.27	Logistic Regression Model	58
4.28	Dataset(1)	60
4.29	Dataset(2)	60
4.30	Dataset Columns	61
4.31	Importing libraries	61
4.32	Reading the Dataset	62
4.33	Removing Null Values	62
4.34	Active Learning Model	62
4.35	Describing Categories	63

4.36	TF-IDF implementation	63
4.37	Finding unigrams and bigrams	63
4.38	Splitting Dataset in Test-Train	64
4.39	Implementing LinearSVC	64
4.40	Describing Confusion Matrix	65
4.41	Getting Correlated terms	65
4.42	Correlated terms obtained	66
4.43	Bigrams and Unigrams Obtained	67
4.44	Results for query-1	68
4.45	Results for query-2	68
4.46	No. of complaints in a category	68
5.1	Mean Accuracy	70
5.2	Classification Metrics	71
5.3	Comparative Results	71
5.4	Confusion Matrix	72

## **Abstract**

Natural language processing (NLP) and neural networks (NNs) have both undergone significant changes in recent years. For active learning (AL) purposes, NNs are, however, less commonly used – despite their current popularity. By using the superior text classification performance of NNs for AL, we can either increase a model’s performance using the same amount of data or reduce the data and therefore the required annotation efforts while keeping the same performance. We review AL for text classification using deep neural networks (DNNs) and elaborate on two main causes which used to hinder the adoption: (a) the inability of NNs to provide reliable uncertainty estimates, on which the most commonly used query strategies rely, and (b) the challenge of training DNNs on small data. To investigate the former, we construct a taxonomy of query strategies, which distinguishes between data-based, model-based, and prediction-based instance selection, and investigate the prevalence of these classes in recent research. Moreover, we review recent NN-based advances in NLP like word embeddings or language models in the context of (D)NNs, survey the current state-of-the-art at the intersection of AL, text classification, and DNNs and relate recent advances in NLP to AL. Finally, we analyze recent work in AL for text classification, connect the respective query strategies to the taxonomy, and outline commonalities and shortcomings. As a result, we highlight gaps in current research and present open research questions.

Active Learning (AL) aims to reduce the amount of data annotated by the human expert. It is an iterative cyclic process between an oracle (usually the human annotator) and an active learner. In contrast to passive learning, in which the data is simply fed to the algorithm, the active learner chooses which samples are to be labeled next. The labeling itself, however, is done by a human expert, the so-called human in the loop. Having received new labels, the active learner trains a new model and the process starts from the beginning. Using the term active learner, we refer to the composition of a model, a query strategy, and a stopping criterion. In this work the model is w.l.o.g. a text classification model, the query strategy decides which instances should be labeled next, and the stopping criterion defines when to stop the AL loop. According to Settles there are three main scenarios for AL:

- (1) **Pool-based**, in which the learner has access to the closed set of unlabeled instances, called the pool
- (2) **Stream-based**, where the learner receives one instance at a time and has the options to keep it, or discard
- (3) **Membership query synthesis**, in which the learner creates new artificial instances to be labeled.

If the pool-based scenario operates not on a single instance, but on a batch of instances, this is called batch-mode AL . Throughout this work we assume a pool-based batch-mode scenario because in a text classification setting the dataset is usually a closed set, and the batch-wise operation reduces the number of retraining operations, which cause waiting periods for the user.

The underlying idea of AL is that few representative instances can be used as surrogate for the full dataset. Not only does a smaller subset of the data reduce the computational costs, but also it has been shown that AL can even increase the quality of the resulting model compared to learning on the full dataset . As a consequence, AL has been used in many NLP tasks, e.g. text classification named entity recognition or machine translation and is still an active area of research.