Table 3. Retweet Count from various sources

S.No	Retweet	Count
1	0	2452
2	1	575
3	2	245
4	3	89
5	4	72
6	5	44
7	6	32
8	7	23
9	8	17
10	9	10
11	10	13
12	11	13
13	12	7
14	13	8
15	14	9
16	15	9
17	16	6
18	17	2
19	18	6
20	19	7
21	20	3
22	21	4
23	22	5
24	23	5
25	24	2
26	25	2
27	26	2
28	29	2
29	30	2
30	31	2
31	32	2
32	34	3
33	36	2
34	36	2
35	40	2
36	41	4
37	43	1
38	46	1
39	47	1
40	49	1
41	50	1
42	52	2
43	54	4
44	55	1
45	56	2
46	57	4
47	62	1
48	64	1
49	66	1

50	69	4
51	75	1
52	77	1
53	87	1
54	89	2
55	95	1
56	97	1
57	98	2
58	105	1
59	107	2
60	110	1
61	111	1
62	116	1
63	118	1
64	120	1
65	123	1
66	133	2
67	134	1
68	140	1
69	147	1
70	157	2
71	159	1
72	171	1
73	177	1
74	181	1
75	188	1
67	197	1
68	219	1
69	253	1
70	294	1

III Results and Discussions

In this section focuses on the results and discussions of this research work.In this approach towards sentiment analysis of a collection of 3749 tweets regarding the ongoing farmers' protest in New Delhi. But before that, let us define sentiment analysis. Sentiment Analysis is the use of NLP algorithms, text analysis, computational linguistics, and biometrics to extract and quantify affective states and subjective information. Here observing sentiment on a day by day basis from the 26th of December 2020 to New Year's Day 2021.

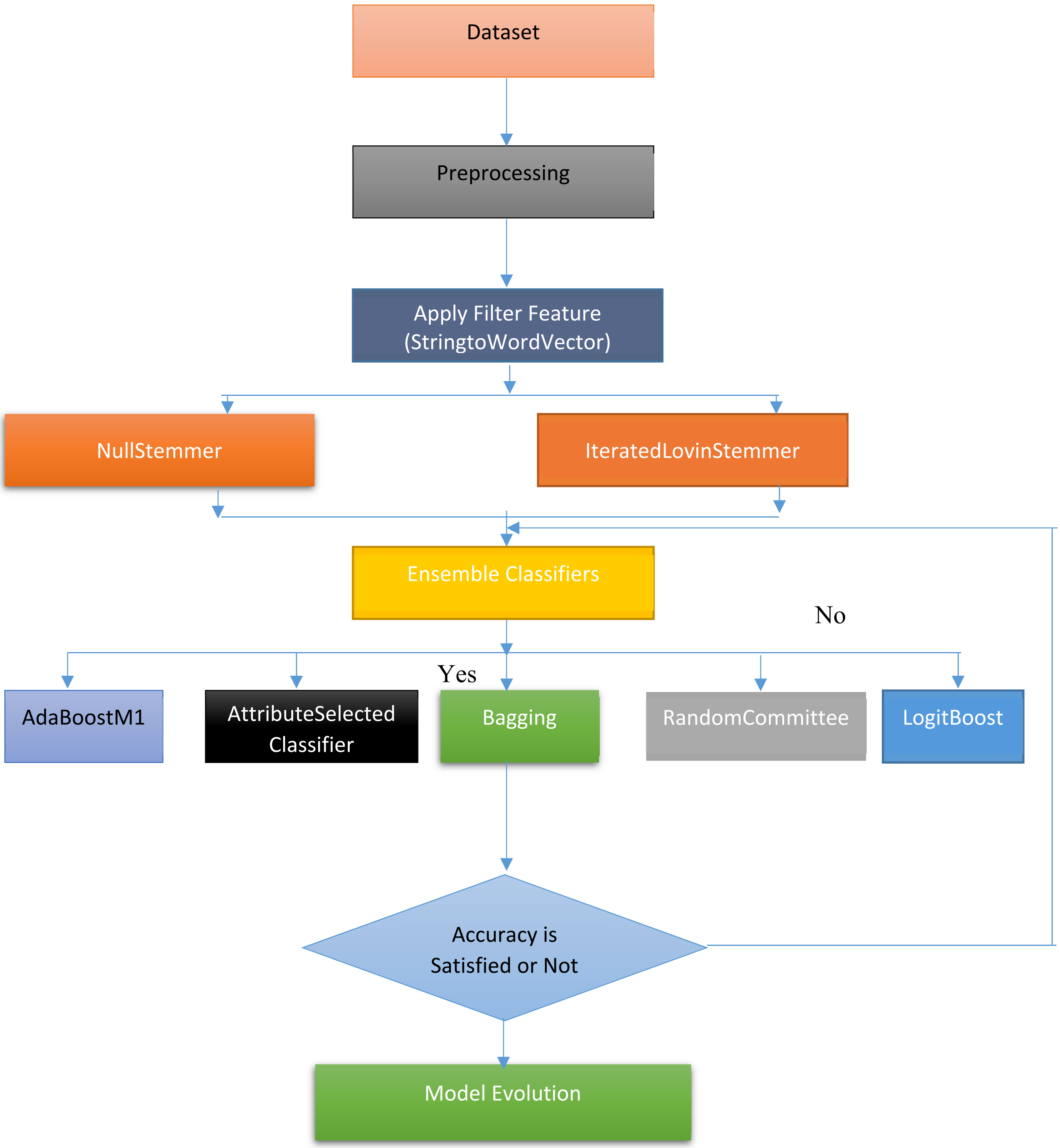

Figure 1. Proposed System Structure

Table 4: Ensemble Models Vs. Measurements Phase I

Ensemble classifiers	Accuracy		Precision		Recall		ROC		Time(In Seconds)	
	NS	ILS	NS	ILS	NS	ILS	NS	ILS	NS	ILS
AdaBoostM1	65.76%	65.76%	-	-	0.66	0.66	0.52	0.52	**0.41**	0.13
AttributeSelectedClassifier	65.76%	65.76%	-	-	0.66	0.66	0.5	0.5	**1.44**	1.38
Bagging	**70.16%**	**70.17%**	0.80	0.80	0.70	0.70	**0.63**	**0.62**	**5.26**	3.74
LogitBoost	67.76%	67.76%	0.78	0.78	0.68	0.68	**0.57**	**0.56**	**0.41**	0.05
RandomCommitte	69.82%	69.82%	0.74	0.74	0.70	0.70	0.64	0.64	**1**	0.16

*** NS- NullStemmer ** ILS-IteratedLovinStemmer**

The above table represents that the ensemble models produce various measurements like,accuracy,precision, recall, ROC, Time taken to build the optimal model which is measured by seconds. These are all values not varying frequently while applying the without stemming and with stemming (IteratedLovinstemming). Slight variations only produced by the ensemble classifersBaaging and LogitBoost classifier.

AdaBoostM1 ensemble classifier has 65.76% of accuracy level when applying both without stemmer (NullStemmer) and with Stemmer (IteratedLovinStemmer).

AttrbuteSelectedClassifier classifier has 65.76% of accuracy level when applying both without stemmer (NullStemmer) and with Stemmer (IteratedLovinStemmer).

Bagging classifier has 70.16% of accuracy level when applying the without stemmer (NullStemmer) and 70.17% of accuracy level when applying the Stemmer (IteratedLovinStemmer).

LogitBoost classifier has 67.76% of accuracy level when applying both without stemmer (NullStemmer) and with Stemmer (IteratedLovinStemmer).

RandomCommittee classifier has 69.82% of accuracy level when applying both without stemmer (NullStemmer) and with Stemmer (IteratedLovinStemmer).

Bagging classifier has 0.80 of precision Value for applying the Nullstemmer and IteratedLovinStemmer. LogitBoost classifier produces 0.78 of precision value for applying the Nullstemmer and IteratedLovinStemmer. RandomCommittee produces 0.78 of precision value for applying the Nullstemmer and IteratedLovinStemmer.

Recall values are not varying while applying the NullStemmer and IteratedLovinStemmer. AdaBoostM1 and AttributeSelectedClassifier algorithms have 0.66 of recall value. Bagging classifier and RandomCommittee classifiers have 0.7 of recall value. LogitBoost Classifier has 0.68 of recall value.

ROC values are having few measurements only varying while applying the nullstemmer and IteratedLovinStemmer of Bagging and LogitBoost classifiers. Rest of the classifiers have same ROC of nullstemmer and IteratedLovinStemmer.

The AdaBoostM1 classifier has taken the time to build the model is 0.41 seconds for without stemmer and 0.13 seconds has taken the time consumption for IteratedLovinStemmer implementation. The AttributedSelectedClassifier model has taken the time to build the model is 1.44 seconds for without stemmer and 1.38 seconds has taken the time consumption for IteratedLovinStemmer implementation.The Bagging classifier has taken the time to build the model is 5.26 seconds for implementation of without stemmer; with stemmer it has taken the time to build the model is 3.74 seconds.The LogitBoost Algorithm while implementing without stemmer, it has taken the time to build the model is 0.41 seconds. While implementing with stemmer it has taken the time to build the model is 0.05 seconds. The RandomCommittee classifier 1 second has taken the time to build the model for without stemmer, when applying the stemmer time consumption is 0.16 seconds.

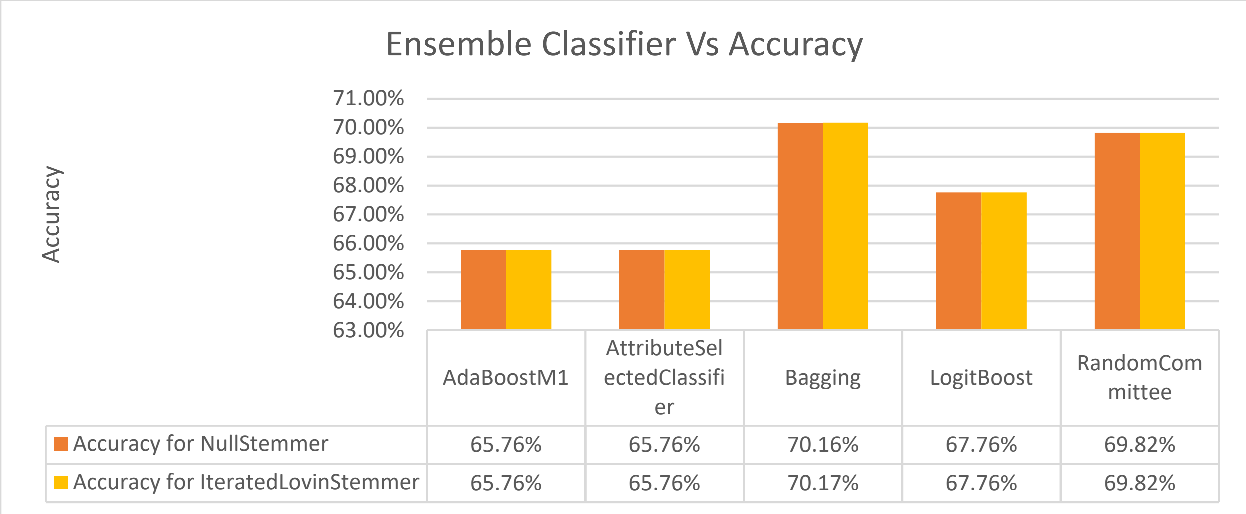

Figure 3: Ensemble Classifier Vs Accuracy

The above diagram Bagging classifier has high accuracy which is 70.16% of accuracy level for Nullstemmer and 70.17% of accuracy level for IteratedLovinStemmer. The rest of the model have below 70% of accuracy level. All the classifiers except Bagging have the accuracy level from 65.76% to 69.82%.

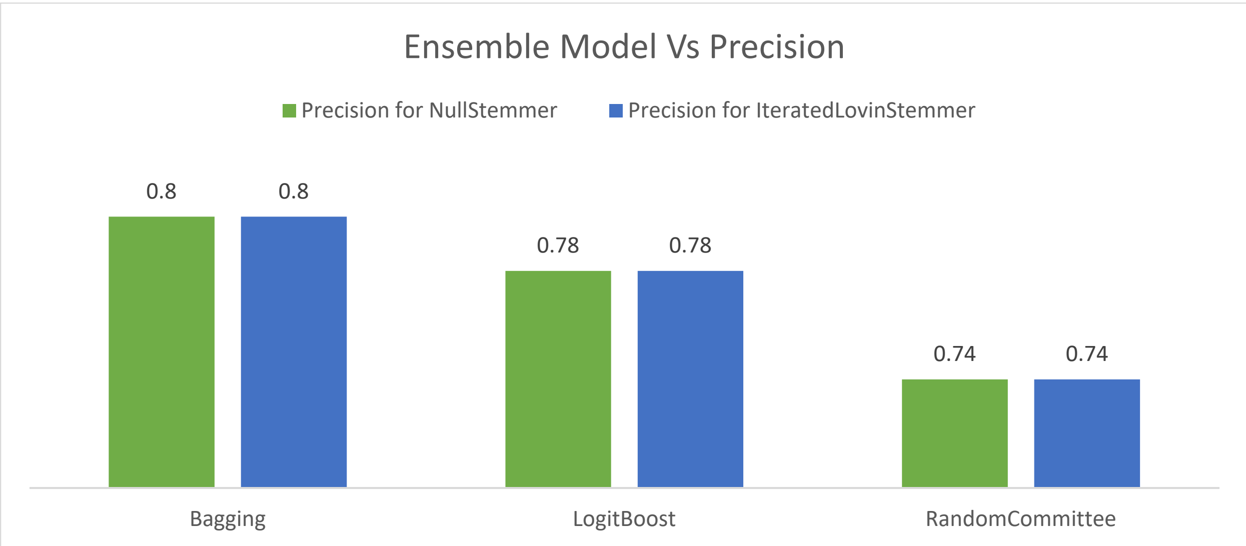

Figure 4: Ensemble Model Vs Precision

The AdaBoostM1, AttributeSelectedClassifier have zero precision value. The Rest of the Bagging value is 0.80 for both Nullstemmer and IteratedLovinStemmer. The LogitBoost value is 0.78 for both NullStemmer and IteratedLovinStemmer. The RandomCommittee

e-ISSN : 0976-5166
p-ISSN : 2231-3850

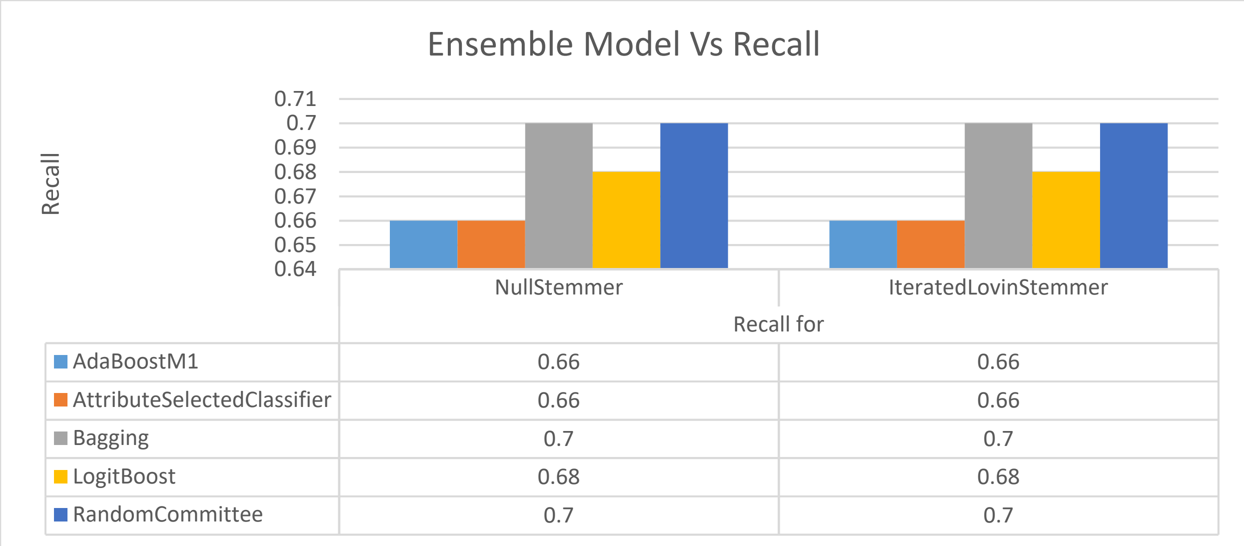

Figure 5: Ensemble Model Vs Recall

The above diagram clearly shows that whether the NullStemmer or IteratedLovinStemmer are implementing ensemble classifiers have no variations for producing the recall value. The AdaBoostM1 ensemble model ,AttributeSelectedClassifier, and LogitBoost Classifiers have 0.66,0.66 & 0.68 respectively.The Bagging and RandomCommittee have same recall value whichi is 0.7. These two classifiers only produce the optimal recall value.

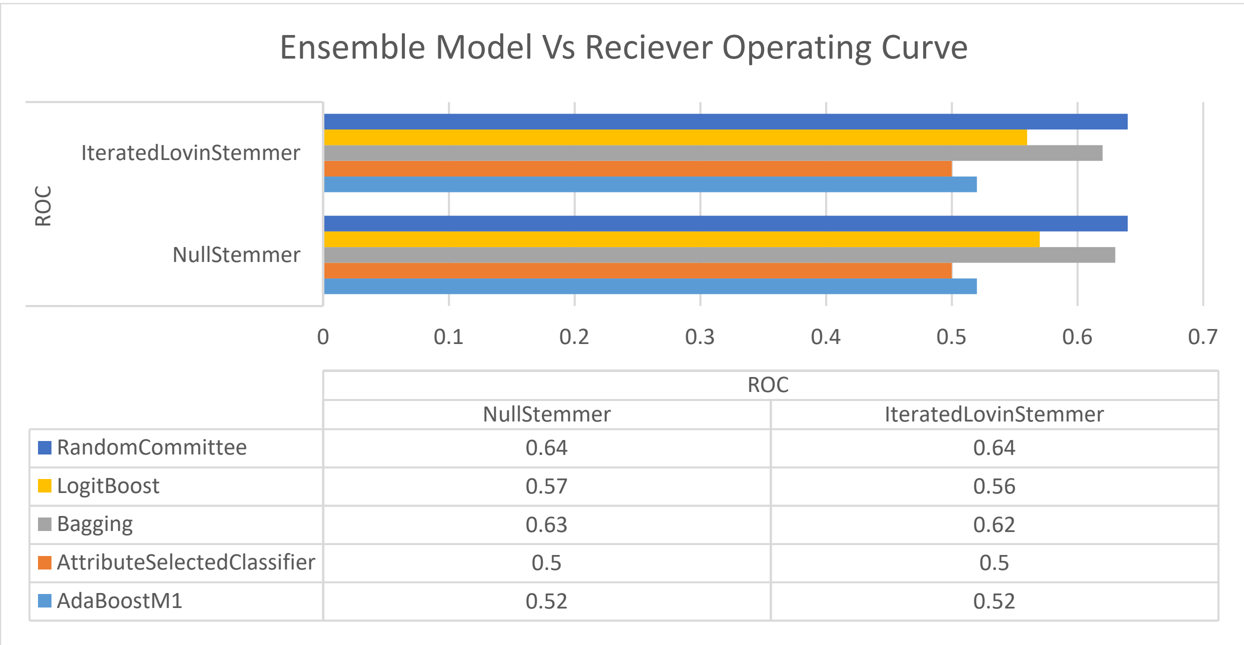

Figure 6: Ensemble Model Vs ROC

This graphical representation shows that the ROC values of ensemble models. The RandomCommittee and Boost Classifiers have 0.64 and 0.63 of ROC value. The rest of the classifiers have the ROC values are below 0.6. Which lies in between 0.5 to 0.57.

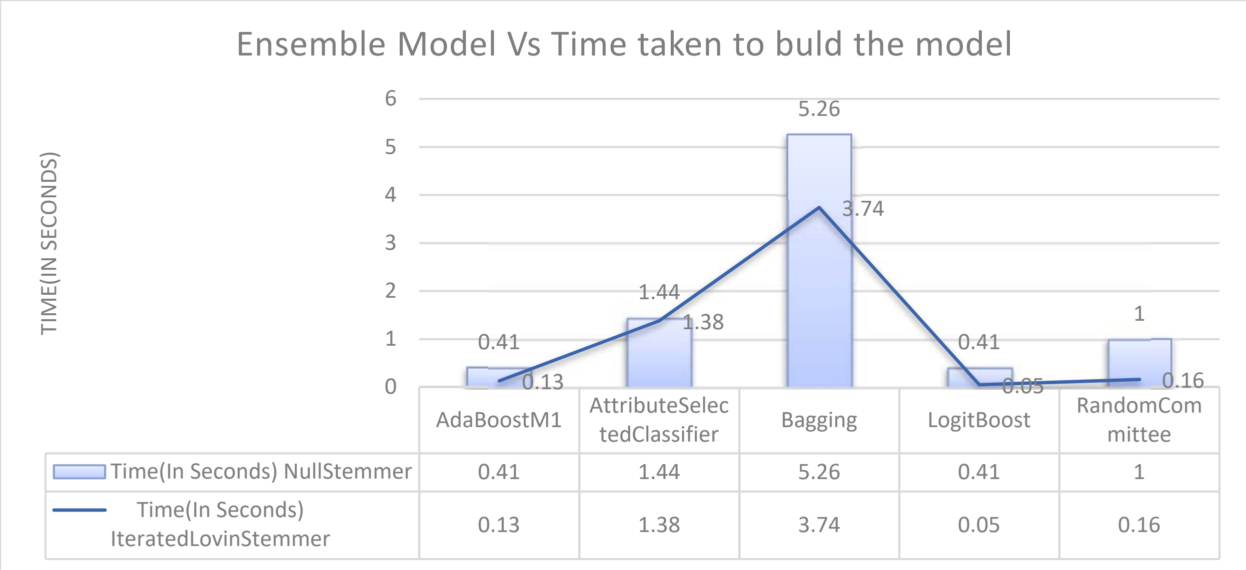

Figure 7: Ensemble Model Vs Time

The above diagram shows that the AdaBoostM1 and LogitBoost classifiers takes the time is less than 1 second. The rest of the model takes the time to build the model more than 1 second. The Bagging classifier takes more time to build the model compare with other classifiers. 5.26 seconds for build the model when applying Nullstemmer and 3.74 seconds for building the model when applying the IteratedLovinStemmer.

Table 6: Bagging Classifier Vs. Measurements Phase I

Cross Validation	Accuracy	Precision	Recall	ROC	Time(In Seconds)
10 Folds	70.16%	0.80	0.70	0.63	5.26
20 Folds	70.46%	0.80	0.70	0.63	1.67
30 Folds	70.36%	0.80	0.70	0.63	1.72
40 Folds	70.49%	0.80	0.70	0.63	1.58
50 Folds	70.49%	0.80	0.70	0.63	1.58

The above table shows that Bagging classifier implementation process with the various cross validation from 10 folds to 50 folds applies with Nullstemmer. There is no changes or variation for precision,racll, roc. It plays the major role the time consumption and accuracy level.

V Conclusions

This research shows that the Bagging classifier has 70.16% of accuracy level when applying the without stemmer (NullStemmer) and 70.17% of accuracy level when applying the Stemmer (IteratedLovinStemmer). Bagging classifier has 0.80 of precision Value for applying the Nullstemmer and IteratedLovinStemmer. LogitBoost classifier produces 0.78 of precision value for applying the Nullstemmer and IteratedLovinStemmer. RandomCommittee produces 0.78 of precision value for applying the Nullstemmer and IteratedLovinStemmer. lowest accuracy value produces the 10 fold cross validation which is 70.16% of accuracy. This 10 fold cross validation takes more time consumption also. Hence this proposed system recommends the Bagging ensemble model for more optimal results.

References

[1] https://rahulprakash-8122.medium.com/sentiment-analysis-of-farmers-protest-389280b64ab9
[2] https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2760764.pdf
[3] SwetavaGanguli, Jared Dunnmon, and Darren Hau. Predicting food security outcomes using cnns for satellite tasking. Stanford University CS 221 Artificial Intelligence Project Report, 2016.
[4] K. Saenkoa, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. Proc. ECCV, September 2010,Heraklion, Greece, 2010.
[5] Joseph J. Lim, RuslanSalakhutdinov, and Antonio Torralba. Transfer learning by borrowing examples for multiclass object detection. NIPS, 2011.
[6] John Roy Porter, LiyongXie, Andrew J. Challinor, Kevern Cochrane, S. Mark Howden,MuhammedMohsin Iqbal, David B. Lobell, and Maria Isabel Travasso. Food security andfood production systems. Climate Change 2014: Impacts, Adaptation, and Vulnerability, pages485–533, 2014.

[7] Peter Nagy. Lstm sentiment analysis — keras. https://www.kaggle.com/ngyptr/d/crowdflower/first-gop-debate-twitter-sentiment/ lstm-sentiment-analysis-keras.
[8] Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. volume 1631, page 1642, 2013.
[9] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. Proceedings of ACL, pp. 115–124, 2005.
[10] Yoon Kim. Convolutional neural networks for sentence classification. arXiv:1408.5882, 2014.
[11] Roman Bartusiak, Lukasz Augustyniak, Tomasz Kajdanowicz, and PrzemyslawKazienko. Sentiment analysis for polish using transfer learning approach. pages 53–59. Institute of Electrical and Electronics Engineers (IEEE), 2015.
[12] Agriculture in the united states (wikipedia). https://en.wikipedia.org/wiki/Agriculture_in_the_United_States/. Accessed: 2017-02-06.
[13] Denny Britz. Implementing a cnn for text classification in tensorflow. http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow/.
[14] Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. Science, 353(6301):790–794, 2016.
[15] Jiaxuan You, Xiaocheng Li, Melvin Low, David Lobell, and Stefano Ermon. Deep Gaussian process for crop yield prediction based on remote sensing data. Association for the Advancement of Artificial Intelligence, 2017.

Authors Profile

Dr.A. Kumaravel is working as a Professor and Dean, School of Computing, Bharath Institute of Higher Education and Research, Chennai. His research interest includes Functional Paradigm, Soft Computing, Cloud Computing, Machine Learning, Computational Intelligence, and Knowledge Engineering. He is a life Member of ISTE and IET. He is a reviewer for IEEE transactions and Elsevier journals.
E-mail: drkumaravel@gmail.com

Dr.G.Ayyappan, is working as an Associate Professor of Department of Information Technology at Prince Shri Venkateshwara Padmavathy Engineering College, Ponmar, Chennai. His research interest includes Machine Learning, Image Processing, Social Networks, and Computational Intelligence.
E-mail:ayyappangmca@gmail.com

Dr.T.Vijayan is working as an Assistant professor, Department of Electronics and Communication Engineering, Bharath Institute of Higher Education and Research, Chennai. His current research interests in the areas of Artificial Intelligence, Machine learning, Soft computing, Digital Image Processing, Signal Processing, and Communication systems.
E-mail: tvij16@gmail.com

Dr.K.Alice is working as an Assistant Professor in Bharath University, Chennai, India. Her research interests are in Digital Image Processing, Machine Learning, Artificial Intelligence and information Security.
E-mail: alicesureshme@gmail.com