CHAPTER 5

RESULTS AND DISCUSSION

In Computer Science, the term implementation means deploying specific computing systems using computer programming to evaluate resources or techniques and achieve the expected outcome. The results of the evaluation help understand whether the experiments have achieved the expected outcome. This chapter presents the evaluation results and related discussion of knowledge bases (SenticNet 5 and BanglaSenticNet), polarity lexicon (English, Bengali), proposed algorithm (MCSAlgo), created and baseline datasets. This chapter also describes the results obtained by implementing different pre-processing techniques and their combinations. The chapter then illustrates the results of different extraction techniques (feature and concept). A comparative study within the study and state-of-art research in each section is also presented in this chapter. Finally, this chapter gives a detailed result summary and description of the studies done in this thesis.

5.1 Performance Analysis of Bengali Knowledge Base (BanglaSenticNet), Polarity Lexicon, and Proposed Algorithm (MCSAlgo)

The main objective of this thesis is to create the resources (a Bengali knowledge base (BanglaSenticNet) and Bengali polarity lexicon). Besides, another objective is to propose an algorithm (MCSAlgo) that could evaluate the performance of MLSA at the concept level. Therefore, this section tries to present the performance of the BanglaSenticNet, Bengali polarity lexicon, the relative performance of the English knowledge base (SenticNet 5), and the English polarity lexicon to figure out either these resources are worthy of research. These evaluations are done using NB, SVM, LSTM, and MCSAlgo algorithms. However, MCSAlgo is used for evaluating only the performance of both lingual knowledge bases. The evaluation parameters are accuracy, precision, recall, and F-score of the confusion matrix (details in Section 3.9.1). This section then presented the comparative analysis of the performance of MCSAlgo and other algorithms in the above list. The results are shown in Table 5.1-Table 5.4.

As per the discussion in Section 3.2, this research has experimented on all three English datasets (based on SenticNet 5) and two Bengali datasets (based on BanglaSenticNet). Firstly, all the algorithms mentioned above were trained with the knowledge bases such as SenticNet 5 and BanglaSenticNet separately. For testing, concepts are extracted as per the rules defined in Section 3.2.1 and match the corresponding knowledge bases. For instance, SenticNet 5 was used to test English concepts, and BanglaSenticNet was used for testing Bengali concepts.

The highest classification performance achieved using BanglaSenticNet on Bengali datasets is accuracy (71.14%), precision (72%), recall (76%), and F-score (69%). Whereas, The highest classification accuracy achieved using SenticNet 5 on English datasets is 79.13%, precision (66%), recall (75%), and F-score (69%). Moreover, on different algorithms and evaluation parameters, the knowledge bases were responding differently. For instance, the best performance for SenticNet 5 on English datasets using the precision value is 66% for the MCSAlgo, whereas recall and F-score values are highest for LSTM, which is 75% and 69%, respectively. However, the best performance for BanglaSenticNet on Bengali datasets using evaluation parameters such as precision is 72% for LSTM, whereas, F-score value is the highest and the same (69%) for two algorithms such as LSTM and MCSAlgo. Finally, on the scale of recall, NB is the best algorithm with an accuracy of 76%.

The results in Table 5.1 shows that, with the limited number of concepts (30000) and its semantics (150000) in BanglaSenticNet compared to SenticNet 5 (100000 concepts and 500000 its semantics), the two knowledge bases shows very similar performance. SenticNet 5 outperforms BanglaSenticNet only on one evaluation parameter (accuracy), though these two knowledge bases work on different data types. From the comparative performance evaluation perspective, it could be concluded that the performance with BanglaSenticNet is satisfactory and significant compared to SenticNet 5. It is also evident that MCSAlgo performs well with SenticNet 5, having very similar classification accuracy compared to NB, SVM, and LSTM. Besides, it outperforms NB and SVM with BanglaSenticNet. The reason behind the better performance of MCSAlgo is the method it uses for classifications (details in Section 3.4). Therefore, MCSAlgo could be considered as a significant contribution to this field.

Table 5.1: Performance of BanglaSenticNet, MCSAlgo, and Relative Evaluation with Baseline

Concepts	**Knowledge Base**	**Evaluation Parameter**	**Average**	**NB**	**SVM**	**LSTM**	**MCSAlgo**
English	SenticNet 5	Accuracy	76.66	76.54	76.98	**79.13**	74.00
		Precision	60.26	58.00	56.05	61.00	**66.00**
		Recall	64.57	62.67	59.60	75.00	61.00
		F-score	61.29	60.33	57.82	**69.00**	58.00
Bengali	BanglaSenticNet	Accuracy	61.49	49.51	61.65	71.14	63.67
		Precision	60.75	62.00	58.00	72.00	51.00
		Recall	63.75	**76.00**	59.00	64.00	56.00
		F-score	64.25	62.00	57.00	69.00	**69.00**

As per the objectives, this research also created a Bengali polarity lexicon (details in Section 3.3). The lexicon has 72433 multi-domain concepts. To better understand this lexicon's performance, the performance of the English polarity lexicon is evaluated side-by-side. The English polarity lexicon has 100000 multi-domain concepts. The lexicon's performance is not compared with any existing Bengali polarity lexicon as the literature review does not prove to have any concept-level Bengali polarity lexicon. Table 5.2 shows the performance with or without using Bengali and English Polarity Lexicons. The study shows the improved performance with both lingual data once polarity lexicons are adopted. Compared to without polarity lexicon, performance improvement is 0.9% and 18.79% for English and Bengali lingual data, respectively. Moreover, the performance of the Bengali polarity lexicon is found significant compared to the English polarity lexicon as the performance is very close even with fewer concepts in the Bengali polarity lexicon.

Table 5.2: Performance of Bengali and English Polarity Lexicon

	With Polarity lexicon			**Without Polarity lexicon**		
Datasets	**NB**	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**
English Dataset	97.00	96.97	**99.32**	93.94	93.94	98.42
Bengali Dataset	63.33	64.50	**93.16**	68.07	74.37	71.41

The study also evaluated the significance of the proposed algorithm about other algorithms on their complexity level, as shown in Table 5.3. The table shows that on the scale of time complexity and in the worst-case NB (i.e., O (n^2)) is faster. Whereas SVM and MCSAlgo have the same time complexity of O (n^3). On the same scale, LSTM is shown to be a slower algorithm with complexity O (n^4*c). However, on the scale of average classification accuracy on both types of languages and knowledge bases, the performance of LSTM is better than other algorithms, and NB is the worst algorithm. The performance of the SVM and MCSAlgo remain stable on both scales. From the above analytical analysis, it could be concluded and recommended that the performance of MCSAlgo is significant. This result also clearly reveals that MCSAlgo could be widely used along with these algorithms for multilingual concept-level sentiment analysis and maybe improved its performance by rethinking and regenerating the logic.

Table 5.3: Significance Analysis of MCSAlgo and Other Applied Algorithms

Algorithm	Complexity	Average Accuracy (%)
Naïve Bayes	$O(n^2)$	63
SVM	$O(n^3)$	69
LSTM	$O(n^4*c)$	75
MCSAlgo	$O(n^3)$	69

5.1.1 Comparing Performance of BanglaSenticNet, Bangla Polarity Lexicon, and MCSAlgo with State-art-Art Research

The performance of BanglaSenticNet, Bangla Polarity Lexicon, and MCSAlgo with State-art-Art Research is shown in Table 5.4. Only baseline datasets such as IMDB[12] (English) and Cricket[13](Bengali) are considered for comparison and ignored created student feedback datasets of English and Bengali lingual data. The reason is the unavailability of a similar student feedback dataset in the Bengali language. The comparison on polarity lexicon shows that the created English polarity lexicon outperforms three other recent polarity-based research on the IMDB dataset. The current research has achieved an accuracy of 99.32%, where existing research has achieved 92.80% (Cambria et al., 2018), 89.20% (Ali et al., 2019), and 91.90% (Shaukat et al.2020). Besides, on the Cricket dataset, created Bengali polarity lexicon outperforms three other recent polarity-based research. The current research has achieved an accuracy of 93.16%, where existing research has achieved 71% (Rahman et al., 2018), 88% (Bodini et al., 2019), and 82.21% (Bhowmik et al., 2021). The reason for outperforming with new polarity lexicon has two reasons: 1) New lexicons are multidomain and enriched with more widely used concepts 2) Existing research has used a feature-based approach; however, this research has used a concept-based approach.

Similarly, the comparison of the performances of knowledge bases shows that there is no proof of having any Bengali knowledge bases in the literature review. Bengali SA is done either using feature polarity lexicon or using labeled data splitting method. Therefore, some state-of-art research that used the above methods is compared with the performance of the Bengali baseline dataset. The comparative analysis shows the SA performance using the Bengali knowledge base (BanglaSenticNet) has outperformed existing research of Rahman et al. (2018) and Haque et al. (2020) by 5%. The research also does the same experiment using the English knowledge base (SenticNet 5) on the English baseline dataset to validate whether the BangalSenticNet evaluation process is worthy. The research has achieved 98.42% accuracy.

In contrast, other similar research has achieved 92.80% (Cambria et al., 2018), 89.20% (Ali et al., 2019), and 91.90% (Shaukat et al., 2020) accuracy. The comparative performance with current research shows, this research has outperformed by approximately 6%. The reason for these improvements are: 1) Applying a large set of preprocessing technique combinations and 2) Applying different feature and concept extraction technique combinations.

The table also shows the comparative performance of algorithms that allows SA at the concept level. However, the literature review section shows, there seems to be no customized algorithm that explicitly deals with SA at the concept level. Therefore, this section has compared the performance with existing work that emphasizes the concept level SA using standard algorithms. The result shows that, with both lingual baseline datasets, this research has outperformed. This research has achieved 76% and 99.32% accuracy on Bengali and English lingual datasets, respectively. The result does not align with Table 5.1 (average results of datasets) as these results are from individual datasets. Whereas Rahman et al. (2018) has achieved 71% accuracy on the Bengali dataset, Poria et al. (2018) and Cambria et al. (2018) has achieved 74.59% and 92% accuracy on the English dataset. The reason for outperforming is due to the application of proper resources and techniques.

Table 5.4: Performance Comparison with State-of-Art Research

Resources or Algorithms	Dataset Language)	Reference	Accuracy/Precision/ F-score
Polarity Lexicons	Cricket (Bengali)	Rahman et al., 2018	71.00
		Bodini et al., 2019	88.00
		Bhowmik et al., 2021	82.21
		Bengali Polarity Lexicon	**93.16**
	IMDB (English)	Cambria et al. 2018	92.80
		Ali et al. 2019	89.20
		Shaukat et al.2020	91.90
		English Polarity Lexicon	**99.32**
Knowledge Bases	Cricket (Bengali)	Rahman et al., 2018	71.00
		Haque et al., 2020	40.00
		BanglaSenticNet	**76.00**
	IMDB (English)	Cambria et al. 2018	92.80
		Ali et al. 2019	89.20
		Shaukat et al.2020	91.90
		SenticNet 5	**98.42**
Algorithms on Concept-level SA	Cricket (Bengali)	Rahman et al., 2018	71.00
		Proposed (MCSAlgo)	**76.00**
	IMDB (English)	Poria et al., 2018	74.59
		Cambria et al. 2018	92.00
		Proposed (MCSAlgo)	**99.32**

5.2 Performance Evaluation of Created Bengali and English Datasets with Baseline Datasets

This research has created three datasets mentioned in Section 3.5 to meet one of its objectives and support its experimentations. Two datasets are in English (English dataset1 and English Dataset2 in Table 5.5), and one dataset is in Bengali (Bengali dataset1 in Table 5.5). English dataset3 (IMDB[12]) and Bengali dataset2 (Cricket[13]) are baseline datasets (details in Section 3.5.2). There are imbalance in sample data size and are resolved by using same number of data in each loop of the experiments. The following table shows the average performance achieved on the created and baseline datasets once tested with different knowledge bases, classifiers, preprocessing techniques, and feature or concept extraction techniques. The table shows that both lingual datasets and types are interchangeably better. However, defining explicitly, English dataset1 and Bengali dataset2 are better performing datasets with the accuracy of 99.94% and 93.16%, respectively. In Bengali datasets, the baseline dataset is working better due to the structuredness in the data. Good labeling also plays a vital role in the improvement of performance.

Table 5.5: Significance Analysis of Datasets with Baseline

Datasets	NB	SVM	LSTM
English dataset1	92.28	93.50	**99.94**
English dataset2	96.97	96.97	**99.86**
English dataset3	97.00	96.97	**99.93**
Bengali Dataset1	63.33	64.50	**86.65**
Bengali Dataset2	68.07	74.37	**93.16**

5.2.1 Comparison with State-of-Art Research on Datasets

Table 5.6 represents the comparison with state-of-art research on datasets. The literature review shows there is some research already on the student feedback dataset in English. The datasets are primarily on survey data. However, this research has created two student feedback datasets using review comments from Facebook. The performance of those two datasets outperforms existing survey-based datasets (Lwin et al., 2020) by 2.94%. The improvements were achieved due to applying different knowledge bases, classifiers, preprocessing techniques, and feature or concept extraction techniques on these datasets. Moreover, the literature review does not prove to have any Bengali student feedback datasets. However, the created dataset has achieved a satisfactory classification accuracy of 86.65%.

This research also used the same algorithms, techniques, and resources on baseline datasets to test whether the created dataset's evaluation is worthy. The baseline datasets also outperform the existing research. For instance, the English baseline dataset achieved 99.93% accuracy, and state-of-art research (Hameed et al., 2020) achieved 97.00% accuracy. Moreover, Bengali baseline datasets achieved 93.16% accuracy, and state-of-art research (Bhowmik et al., 2021) achieved 82.20% accuracy. The results indicate that the datasets and methodology applied in this study are well suited and worthy of further use.

Table 5.6: Comparison with State-of-Art Research on Datasets

Datasets (Language)	**References**	**Performance (Accuracy/precision)**
Students' feedback (English)	Lwin et al., 2020	97.00%
	This research (dataset1&2)	**99.94%**
IMDB (English)	Hameed et al., 2020	90.50%
	This research (dataset3)	**99.93%**
Students' feedback (Bengali)	No dataset found	NA
	This research	**86.65%**
Cricket dataset (Bengali)	Bhowmik et al., 2021	82.20%
	This research (Dataset2)	**93.16%**

5.3 Testing with Different Feature and Concept Extraction Techniques

This research aims to analyze and evaluate the best extraction techniques among feature (s) and concepts and their combinations for MLSA. The objective is met by applying different extraction techniques individually and in conjunction. In this section, the results of varying feature and concept extraction techniques and their combinations on different English and Bengali datasets (as discussed in the methodology chapter) were presented to demonstrate the impact of applying them in those datasets. Besides, this section compares the performance of the feature-based and concept-based classifier on different data sizes of both types of data. This section then presented the results on whole English and Bengali datasets after applying the distinct feature and concept extraction techniques. Finally, this section has shown the average and maximum performance of the classifier on the feature and concept-based approach for both types of datasets.

5.3.1 Comparison of Feature and Concept Extraction Techniques Performance on Different Data Sizes

Mostly used features and concepts in this thesis are TD - IDF (Simple), unigram, bigram, trigram, parts of speech, and concepts. This section first tried to determine whether the data size has any impact (positive or negative) on the performance of features and concept extraction techniques. A comparative result is shown in Table 5.7. The performance is evaluated using three classifiers, NB, SVM, and LSTM. The results reveal that by applying feature and concept extraction techniques on English datasets, the performance sometimes improved with increased data sizes (number of comments) and vice-versa.

However, the performance mostly improved with the increase in data sizes of the Bengali datasets. Also, the feature-based approach is found to be better than the concept-based approach for Bengali datasets due to the lack of necessary concepts in concept dictionaries; however, the result indicates, for large datasets, this gap will be too minor.

It is also seen from the table that, overall, the performance of the concept extraction technique improved more once data sizes are increased. Besides, the performance improvement varied with datasets (newly created datasets responded positively than the baseline datasets). The consequences are due to the availability of more features and concepts in the newly created dataset. It is mentionable that the performance of the features and concept extraction techniques also depends on the algorithms. For instance, with the same size of data (English and Bengali) and feature or concept extraction techniques, LSTM has the highest average classification accuracy. This algorithm outperforms two famous algorithms NB and SVM.

Table 5.7: Performance of Feature and Concept Extraction Techniques on Different Data Size

Dataset	Data Size	Feature-Based			Concept-Based		
		NB	SVM	LSTM	NB	SVM	LSTM
English Dataset1	100	50.00	40.00	**98.64**	50.00	55.00	**98.87**
	200	70.00	70.00	94.33	62.50	60.00	94.62
	300	66.67	70.00	97.39	65.00	63.33	98.12
English Dataset2	100	55.00	55.00	96.51	75.00	75.00	93.96
	200	60.00	67.50	**97.06**	75.00	77.50	**95.50**
	300	63.33	61.67	93.19	66.67	65.00	92.38
English Dataset3	100	75.00	75.00	**95.87**	95.00	95.00	91.50
	200	87.50	90.00	93.46	80.00	82.50	90.50
	300	88.33	88.33	92.50	90.00	90.00	**98.92**
Bengali Dataset1	100	70.00	62.50	92.59	52.50	55.00	70.71
	200	75.00	65.00	94.59	57.50	59.68	79.09
	300	75.00	75.00	**99.92**	63.33	64.50	**86.65**
Bengali Dataset2	100	67.00	70.00	92.72	44.00	52.00	80.52
	200	80.00	82.00	**99.61**	53.00	60.00	83.19

Dataset	Data Size	Feature-Based			Concept-Based		
		NB	SVM	LSTM	NB	SVM	LSTM
	300	80.00	85.50	96.47	51.00	52.00	**93.16**

5.3.2 Experiment with Different Feature and Concept Extraction Techniques on Whole English Datasets

English and Bengali data have experimented with two methods: based on different data sizes of datasets and the whole dataset(s) to understand the impact of varying feature and concept extraction techniques in MLSA. The previous section has described the first method. This section presents the impact of using different extraction methods solely and as a combination.

The extraction impact on English datasets (dataset1, dataset2), and dataset3 (Baseline)) are shown in Table 5.8, and Bengali datasets (dataset1) and dataset2 (Baseline)) are shown in Table 5.9. Table 5.8 shows that applying the feature extraction techniques in combination with English datasets is better on the scale of classification accuracy than using them individually in most cases. For instance, the highest classification accuracy was achieved with the concept and trigram combination (98.87%) for dataset1, bigram, and trigram combination (98.42%) for dataset2.

The result may sometimes vary. For dataset3, the highest performing technique is the concept (simple) with an accuracy of 99.32%. The performance is evaluated using three classifiers such as NB, SVM, and LSTM. However, LSTM helps get optimal performance by applying feature or concept extraction techniques individually or in combination. The table and above discussion indicate that, among two extraction techniques, the concept extraction method works better than a feature extraction method on the scale of classification accuracy either individually or in combination. The reason for improved performance by concept extraction technique is its rich set of training data and appropriateness of concepts and corresponding labels. Overall, the combination between features and concepts shows improved performance on all the English datasets.

Table 5.8: Result on Different Feature and Concept Extraction Techniques on Whole English Datasets

Feature (s)	**Dataset1**			**Dataset2**			**Dataset3**		
	NB	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**
TD - IDF (Simple)	88.21	89.02	96.71	87.88	88.00	95.43	72.50	74.00	94.82
Unigram	84.15	84.96	92.57	84.85	85.00	96.38	84.50	83.00	96.48
Bigram	86.18	86.59	91.37	90.91	90.91	95.72	65.00	63.00	95.57
Trigram	87.40	87.40	81.57	84.85	84.85	95.81	55.50	57.00	91.75
Unigram + Bigram	86.59	86.18	96.02	82.00	81.82	91.87	79.00	77.00	97.08
Bigram + Trigram	85.77	85.77	90.11	90.91	91.00	**98.42**	60.50	59.00	91.56
Unigram + Bigram + Trigram	89.84	90.90	94.68	81.82	82.00	92.32	79.50	78.00	98.36
Emphasizing Parts of Speech	88.62	88.21	90.24	93.94	94.00	96.08	46.50	46.00	91.48
Concept (Simple)	86.92	87.34	92.88	74.11	76.14	91.49	97.00	96.97	**99.32**
Concept + Unigram	62.50	60.00	94.62	75.00	77.50	95.50	80.00	82.50	90.50
Concept + Bigram	65.00	63.33	98.12	75.00	75.00	93.96	95.00	95.00	91.50
Concept + Trigram	50.00	55.00	**98.87**	66.67	65.00	92.38	90.00	90.00	98.92

Table 5.9 show that applying the feature extraction techniques in combination with Bengali datasets is better on the scale of classification accuracy than using them individually in most cases. For instance, the highest classification accuracy is achieved using concept and trigram combination (86.65%) for dataset1. Some other techniques or combinations that work better are concept+unigram (71.41%) and simple concepts (71.07%). This result may sometimes vary, and only feature extraction techniques could work better. For instance, the bigram features were found to be working better with an accuracy of 74.37% for dataset2. However, unlike dataset1, the optimal technique or their combination on dataset2 is concept+trigram with 93.16% accuracy. Some other techniques or combinations that work better are unigram+concept (71.45%) and bigram+concept (71.69%).

The table's values and above discussion indicate that, between two extraction methods, the concept extraction method works better than a feature extraction method on the scale of classification accuracy. The result also shows that the concept extraction method works better on highly preprocessed data. The result of concept+trigram with dataset2 is significantly the best among all. The improvement of results are due to size of the training and testing data. Moreover, concept dictionary also plays a good role with its large volume of concepts. The feature and concept extraction techniques were tested using three classifiers, such as NB, SVM, and LSTM, where LSTM and SVM worked better.

Table 5.9: Result of Different Feature and Concept Extraction Techniques on Whole Bengali Datasets

Feature	Dataset1			Dataset2		
	NB	SVM	LSTM	NB	SVM	LSTM
TD - IDF (Simple)	50.40	50.81	69.34	67.65	65.55	66.57
Unigram	50.81	53.23	69.37	68.07	65.13	67.05
Bigram	49.60	50.81	69.32	65.97	**74.37**	66.52

Feature	Dataset1			Dataset2		
	NB	SVM	LSTM	NB	SVM	LSTM
Tigram	47.98	48.39	69.26	62.18	63.45	66.69
Unigram + Bigram	56.05	55.65	69.44	67.23	70.17	66.49
Bigram + Trigram	49.19	61.70	69.29	65.97	64.29	66.72
Unigram + Bigram + Trigram	60.13	52.02	69.37	64.29	66.81	66.80
Concept (Simple)	49.52	49.52	71.07	50.33	60.17	69.65
Concept + Unigram	49.52	49.52	**71.41**	50.00	53.20	71.45
Concept + Bigram	47.18	47.18	70.68	53.00	48.15	71.69
Concept + Trigram	63.33	64.50	**86.65**	47.64	47.47	**93.16**

5.3.3 Average Performance of Feature and Concept-Based Approach on English and Bengali Datasets

To better understand how the feature and concept-based approach works on English and Bengali datasets, the average performance of both approaches was studied and shown in Table 5.10 and Table 5.11.

The average performance achieved using different feature combinations on English dataset1, dataset2, and dataset3 is 88.71%, 89.83%, and 76.55%. Besides, the average performance achieved using different concepts on English dataset1, dataset2, and dataset3 are 71.94%, 79.56%, and 90.38%, respectively. The results also reveal that the concept extraction technique works better on the datasets having formal English comments, for instance, the baseline dataset (dataset3). It could be concluded that using appropriate features and concepts positively impacts the performance of the sentiment analysis. It could also be recommended that the LSTM classifier is better for SA using both features and concepts or their combinations. Also, the concept-based approach gives better performance once applied to individual datasets. For instance, 97.2% accuracy has been achieved using the concept extraction technique, which is higher than all feature extraction techniques.

Moreover, the average performance achieved using different feature combinations on Bengali dataset1 and dataset2 is 57.19% and 61.30%, respectively. Besides, the average performance achieved using different concepts on Bengali dataset1 (65.44%) and dataset2 (63.21%). As with the English datasets, the concept extraction technique does not require formal Bengali comments. However, unlike English datasets, using appropriate features and concepts positively impacts the performance of the sentiment analysis. A similar recommendation could be given for Bengali data; that is, the LSTM classifier is better for SA using both features and concepts or their combinations. Also, if individual datasets are considered, the concept-based approach is better with Bengali datasets too. For instance, 85.62% accuracy has been achieved using the concept extraction technique, which is higher than all feature extraction techniques.

Reviewing Table 5.11, the overall performance of the feature-based approach (85.03%) is better for English datasets. However, the performance of the concept-based approach (64.32%) is better for Bengali datasets. Therefore, looking into the individual dataset, classifier performance, and average performance, it could be concluded that, comparatively concept-based approach is better for SA at a large scale. The reason for better performance by concept-extraction is that it could work with affective information related to natural language.

Table 5.10: Average Performance of the Feature and Concept-Based Approach on Individual Dataset

	Feature-Based				Concept-Based			
Datasets	**NB**	**SVM**	**LSTM**	**Average**	**NB**	**SVM**	**LSTM**	**Average**
English Dataset1	87.09	87.38	91.66	88.71	59.17	59.44	97.2	71.94
English Dataset2	87.12	87.12	95.25	89.83	72.22	72.5	93.95	79.56
English Dataset3	67.88	67.13	94.64	76.55	88.33	89.17	93.64	90.38
Bengali Dataset1	50.62	50.99	69.96	57.19	57.78	59.73	78.82	65.44
Bengali Dataset2	56.85	55.65	71.41	61.30	49.33	54.67	85.62	63.21

Table 5.11: Average Performance of Feature and Concept-Based Approach on All datasets

Datasets	Feature-Based	Concept-Based
English Dataset	85.03	80.62
Bengali Dataset	59.25	64.32

5.3.4 Maximum Performance of Feature and Concept-Based Approach on English and Bengali Datasets

This section (Table 5.12) represents the maximum performing extraction techniques on English and Bengali datasets. The table reveals that by applying different feature extraction technique and their combinations on English datasets, the maximum performance achieved is 98.42% with bigram+trigram combination and LSTM classifier. Therefore, this combination could be considered as the optimal feature extraction combination. However, applying the concept extraction technique, concept (simple) produces the optimal result of 99.32%. Moreover, the application of concept and feature extraction techniques in combinations on English datasets shows that optimal performance is achieved by concept+trigram combination with 98.87% accuracy. Overall, the performance of dataset2 is better than the other two datasets once feature extraction techniques are applied. The reason for the better result is the small amount of data in the dataset. However, the performance of concept extraction techniques on dataset3 is better with LSTM than the other two datasets due to its structuredness.

The table also reveals that, after applying different feature extraction techniques and their combination on the Bengali dataset, the bigram feature technique has achieved a maximum of 74.37% accuracy. Therefore, for Bengali data, bigram could be considered as an optimal feature technique. Overall, the performance of Dataset2 is better than dataset1. The reason is, dataset1 is unstructured than dataset2. However, the performance of the concepts with LSTM is better for both datasets (accuracy of 86.65% and 93.16%). Overall, the performance of dataset1 is better than dataset2. The reason is, dataset1 avails more concepts than dataset2.

Table 5.12: Maximum Performance of the Classifier on Feature and Concept-Based Approach

Datasets	**Feature-Based**			**Concept-Based**		
	NB	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**
English Dataset1	89.84	90.90	96.71	86.92	87.34	**98.87**
English Dataset2	93.94	93.94	**98.42**	75.00	77.50	95.50
English Dataset3	84.50	83.00	98.36	97.00	96.97	**99.32**
Bengali Dataset1	60.13	61.70	69.44	63.33	64.50	**86.65**
Bengali Dataset2	68.07	74.37	71.41	53.00	60.17	**93.16**

5.3.5 Comparison with State-of-Art Feature and Concept Extraction Technique Research

In Table 5.13, the comparison is shown about different datasets and state-of-art research on feature and concept extraction techniques. This research has outperformed state-of-art research on both feature and concept extractions by 7.86% (Hameed et al., 2020) and 7.32% (Cambria et al., 2018), respectively, on the English baseline dataset. However, considering the Bengali baseline dataset, state-of-art research on feature extraction has outperformed this research by 7.83% (Bhowmik et al., 2021). Moreover, if the performances of concept extraction techniques are considered, there is no proof of having any work on Bengali datasets. Therefore, this research compares performance between state-of-art research (Bhowmik et al., 2021) on feature and concept extraction technique of this research. The comparison reveals that this research is outperforming state-of-art research by 10.96%.

Moreover, due to the lack of research on student feedback datasets in Bengali, this section could not compare with state-of-art researches. Also, there was no work found in the literature; hence, no comparison was made on the English feedback dataset. Therefore, the comparisons shown are: 1) the performance of feature extraction techniques between this research and state-of-art research (Lwin et al., 2020) 2) the performance of feature extraction techniques of state-of-art research and concept extraction techniques of this research. The comparative evaluation shows this research outperformed state-of-art research by 1.42% on the performance of the feature extraction technique. The performance comparison of concept extraction techniques of this research with feature extraction technique of state-of-art research shows this research outperformed by 1.87%. Overall, this research outperformed in the scale of accuracy once compared to state-of-art research.

Table 5.13: Comparison with State-of-Art Feature and Concept Extraction Technique Research

Datasets (Language)	Feature/concepts extraction Techniques	References	Performance (Accuracy/precision)
Students' feedback (English)	**Feature**	Lwin et al., 2020	97%
		This research (dataset1&2)	**98.42%**
	Concepts	No Research found	NA
		This research (dataset1&2)	**98.87%**
IMDB (English)	**Feature**	Hameed et al., 2020	90.50%
		This research (dataset3)	**98.36%**
	Concepts	Cambria et al., 2018	92.00%
		This research (dataset3)	**99.32%**
Students' feedback (Bengali)	**Feature**	No dataset found	NA
		This research	**69.44%**
	Concepts	No dataset found	NA
		This research	**86.65%**
Cricket dataset (Bengali)	**Feature**	Bhowmik et al., 2021	**82.20%**
		This research (Dataset2)	74.37%
	Concepts	No Research found	NA
		This research (Dataset2)	**93.16%**

5.4 Testing with Different Preprocessing Techniques

One of the objectives of this thesis is to examine the preprocessing techniques and their combinations for MLSA and search for the optimal one. In this respect, different experiments were carried out applying the preprocessing techniques individually and in combination. In this section, the results obtained by applying preprocessing techniques and their combinations on English and Bengali datasets (as discussed in the methodology chapter) were presented to demonstrate the impact of preprocessing techniques on both data types. This section then represents the average and maximum performance shown by the different combinations of pre-processing techniques on those datasets. Finally, a comparative analysis is done with existing work that emphasizes preprocessing techniques and their combinations.

5.4.1 Experiment with the Different Preprocessing Combination on English and Bengali Datasets

This thesis adopted the following types of preprocessing techniques from among many types of existing preprocessing techniques (details in Section 3.7), such as removing punctuation, negation, reduction of letter repetition, stop word deduction, stemming, and tokenization for both English and Bengali datasets. Additionally, case conversion for English datasets as the Bengali data has no cases. Table 5.14 represents the abbreviations of different preprocessing techniques. In this section, the abbreviated form was used for the smooth presentation of data and results.

Table 5.14: List of Abbreviations of Different Preprocessing Techniques

Preprocessing techniques	Abbreviation
Remove Punctuation	RP
Negation	N

Preprocessing techniques	Abbreviation
Reduction of Letter Repetition	RLR
Stop Word Deduction	SWD
Stemming	S
Tokenization	T
Case Conversion	CC

The performance of preprocessing techniques and their combinations for both lingual datasets were evaluated using three famous classification algorithms or classifiers such as NB, SVM, and LSTM (details in Section 3.1 and 3.7). Moreover, MCSAlgo was used in rare cases, such as for evaluating SenticNet 5 and BanglaSenticNet.

The preprocessing techniques were evaluated using three English datasets, where dataset1 and dataset2 were created in this thesis, and dataset3 is the IMDB[12] dataset and used as a baseline. Meanwhile, two Bengali datasets were used, where dataset1 was created in this thesis, and dataset2 is collected cricket dataset[13] and used as a baseline (details in Section 4.2 and Table 4.4). All the datasets are well processed with proper annotations. Here, English dataset1, dataset2 and Bengali dataset1 are student feedback datasets. The baseline datasets are different due to the unavailability of student feeback datasets in public especially Bengali. Also, few student feedback datasets available having different formation (mostly structured and survey based) from the created datasets (unstructured and unbiased user comments).

5.4.1.1 Experiment with No Preprocessing and Applying All Pre-Processing Techniques at Once

The data sets were tested using many preprocessing technique formations. Before testing with a different composition, the datasets were also tested without preprocessing. Table 5.15 shows the results obtained with varying classifiers without applying any preprocessing on English datasets. This table also shows the results obtained by applying all (as mentioned above) preprocessing types. It is evident from the result that, for all the classifiers and the datasets, the classification accuracy is better once preprocessing is applied. The accuracy values in the table reveal that the average performance of sentiment classification improves up to 4.85 % once all preprocessing techniques are applied in combination. Notably, new datasets have achieved improved performance of 3.1% compared to the baseline dataset once all preprocessing techniques were applied. The reason is, the new datasets contain more noisy data than the baseline dataset. The performance increased once preprocessing techniques are applied because some sentiment keywords revealed once converted to the basic form, such as 'gooood' is converted to 'good.'

Table 5.15: Result of No Preprocessing and All Preprocessing Type Formation on English Datasets

Preprocessing Formation	**Dataset1**			**Dataset2**			**Dataset3**		
	NB	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**
No preprocessing	82.93	83.33	93.14	81.82	81.82	90.86	80.00	78.00	90.37
CC+T+S+SWD+RLR+N+RP	84.96	85.37	**98.63**	90.91	90.91	**95.02**	87.50	80.00	**92.60**

Meanwhile, for Bengali datasets, a similar process was applied. Table 5.16 shows the results achieved by different classifiers using preprocessing techniques (all types or no preprocessing) on Bengali datasets. With the application of a similar process, the output for Bengali datasets is less than the English datasets due to the unavailability of preprocessing tools. Moreover, the table reveals that the average performance improvement is 6% once all the preprocessing techniques are applied to a new dataset. Also, the performance improvements of the created dataset are less than baseline once all preprocessing techniques are applied. The improvement is due to the inefficiency of existing preprocessing tools in Bengali. The top results achieved with or without preprocessing technique applications are highlighted in the table.

Table 5.16: Result on No Preprocessing and All Preprocessing Type Formation Applied to Bengali Datasets

Preprocessing formation	**Dataset1**			**Dataset2**		
	NB	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**
No Preprocessing	53.23	51.61	69.26	55.65	52.82	69.78
T+S+SWD+RLR+N+RP	55.65	52.82	**69.29**	62.34	65.16	**83.11**

5.4.1.2 Experiment with Single Preprocessing Technique

This research has emphasized preprocessing techniques in evaluating the performance primarily from both individuals and in combination perspectives. Table 5.17 present the results of a single pre-processing technique applied to English datasets. The classification accuracy seems to be a bit higher for all the classifiers, and this is because of the inability of preprocessing techniques applied. The techniques sometimes cannot process the raw datasets to a required extent, thus impose higher accuracy. Among applied preprocessing techniques, removing punctuation shows the highest classification accuracy with 99.85%. Surprisingly, converting cases show the second-highest accuracy with 99.02%. Besides, on average, removing punctuation and dataset1 (once preprocessing techniques are applied) offers better classification accuracy as more real words revealed once punctuations are removed. Notably, the improvement of performance from no preprocessing technique (Table 5.15) to single preprocessing techniques (Table 5.17) is 3.2%.

Table 5.17: Result of Single Preprocessing Technique on English Dataset

Preprocessing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
RP	85.77	86.59	95.33	90.91	91.11	**98.50**	81.50	78.50	**99.85**	**89.78**
N	82.93	82.52	93.09	81.82	81.82	97.96	82.50	82.50	97.23	86.93
RLR	86.59	88.21	93.35	87.00	87.88	90.17	81.00	78.00	91.49	87.08
SWD	84.96	86.18	92.12	85.00	84.85	92.33	80.00	79.50	95.85	86.75
S	85.37	85.77	94.56	87.88	88.00	96.95	77.50	78.50	93.29	87.54
T	88.62	88.21	91.95	87.50	88.00	97.70	82.50	79.50	97.23	89.02
CC	86.18	86.99	**99.02**	85.00	85.45	93.66	80.50	79.50	94.88	87.91

Whereas, in the case of Bengali datasets, the results (as presented in Table 5.18) are very similar to English datasets except for the classification accuracy. It is less than that of the English dataset due to the inaccurate application and actions taken by the preprocessing techniques such as stemming, where stemmers cannot retain root words sometimes due to the lack of standardization. It is evident from Table 5.18 that, reduction of letter repetition shows better classification accuracy for Bengali datasets with 69.65% accuracy. Removing punctuation shows the second-highest classification accuracy of 69.42%. It is evident that, on average, dataset1 shows better classification accuracy once single preprocessing is applied. However, the improvement of performance from no preprocessing technique (Table 5.16) to single preprocessing techniques (Table 5.18) is less than 1% and not remarkable than English datasets.

Table 5.18: Result of Single Preprocessing Technique Applied to Bengali Datasets

Preprocessing formation	**Dataset1**			**Dataset2**			**Average**
	NB	**SVM**	**LSTM**	**NB**	**SVM**	**LSTM**	
RP	50.81	53.23	**69.57**	50.4	51.61	69.38	57.50
N	56.45	56.45	69.28	49.19	51.21	69.35	**58.66**
RLR	48.79	53.63	**69.65**	50	49.19	69.4	56.78
SWD	52.02	49.6	69.46	49.19	56.45	69.33	57.68
S	50.00	50.4	69.27	56.45	56.45	69.38	**58.66**
T	50.81	52.82	69.31	46.77	48.39	69.42	56.25

It is remarkable that, with the use of a single preprocessing technique, the performance of created datasets improved more than baseline datasets for both English and Bengali cases.

5.4.1.3 Experiment with Two Preprocessing Technique Combinations

The second experiment is on the application of two preprocessing technique combinations. Table 5.19 shows the results of the two preprocessing technique combinations on English datasets. It is seen from the table that the impact of applying two preprocessing techniques combination are very close for all the datasets. Overall, among all the applied two preprocessing technique combinations, stemming and negation combinations improved performance accuracy. Applying the preprocessing combinations on individual datasets, tokenization, and negations combinations give the highest accuracy of 99.81% for dataset1. The stemming and negation combination works better for dataset2 with an accuracy of 99.43%. Moreover, stemming and reduction of letter repetition combination works better for dataset3 with an accuracy of 96.75%.

For dataset1, case conversion and stemming combination, negation and punctuation removal combination, tokenization and reduction of letter repetition combination, tokenization and stemming combination also works well with 97.11%, 99.30%, 97.54%, 99.11% accuracy, respectively. For dataset2, better performing preprocessing technique combinations are negation and punctuation removal (98.33%), stop word deduction and negation (97.11%), tokenization and punctuation removal (97.33%), case conversion and reduction of letter repetition (99.07%), case conversion and stemming (99.30%). Finally, for dataset3, well-performing combinations are stemming and stop word deduction (95.42%), case conversion, and tokenization (96.05%). Overall, performance improvement by applying two preprocessing technique combinations (Table 5.19) is 2.5% compared to applying no preprocessing (Table 5.15).

Table 5.19: Result of Two Preprocessing Technique Combinations on English Datasets

Preprocessing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
N+RP	87.40	87.40	97.11	84.85	84.85	98.33	81.00	81.00	88.81	87.86
RLR+RP	85.37	84.96	93.15	87.88	87.88	90.35	81.00	79.50	90.16	86.69
RLR+N	86.59	88.21	91.51	84.85	84.85	92.54	83.50	82.50	90.16	87.19
SWD+RP	87.80	88.21	94.07	81.82	81.82	94.85	73.50	75.50	80.47	84.23
SWD+N	84.96	84.96	90.66	75.76	75.76	97.11	78.00	76.50	91.73	83.94
SWD+RLR	85.37	86.18	92.97	90.91	90.91	92.26	80.50	81.50	91.01	87.96
S+RP	89.02	89.43	95.33	84.85	84.85	92.97	81.50	81.50	92.60	88.01
S+N	87.40	87.80	96.18	93.94	93.94	**99.43**	83.00	80.00	90.19	**90.21**
S+RLR	84.96	86.18	91.35	90.91	90.91	91.93	77.50	77.50	**96.75**	87.55
S+SWD	83.74	83.74	93.62	87.88	87.88	91.31	84.00	80.50	95.42	87.57
T+RP	87.80	88.21	91.95	93.94	93.94	97.33	82.00	76.00	88.81	88.89
T+N	86.59	87.80	**99.81**	72.73	72.73	91.23	81.00	74.00	91.49	84.15
T+RLR	86.99	87.40	97.54	87.88	87.88	94.40	82.00	78.00	90.16	88.03
T+SWD	85.37	86.18	92.35	87.88	87.88	91.28	81.00	81.00	91.73	87.19
T+S	86.59	85.77	99.11	87.88	87.88	95.48	80.50	80.50	90.19	88.21
CC+RP	87.80	87.40	90.73	84.85	84.85	91.32	78.50	78.00	91.01	86.05
CC+N	86.18	86.18	92.39	90.91	90.91	95.15	76.00	75.50	96.05	87.70
CC+RLR	86.18	88.21	96.29	78.79	78.79	99.07	76.00	80.00	93.77	86.34
CC+SWD	85.77	86.99	96.46	93.94	93.94	94.08	76.00	77.00	91.34	88.39
CC+S	87.80	88.62	99.30	88.00	88.62	99.30	81.00	77.50	94.90	89.45
CC+T	86.59	84.96	93.99	84.85	84.85	91.28	78.50	76.50	96.05	86.40

In Bengali datasets, presented in Table 5.20, only 15 combinations of preprocessing techniques were found as case conversion does not work on Bengali data. From all these 15 combinations, tokenization and reduction of letter repetition combinations work better than other combinations with an accuracy of 59.87%. Other best performing two preprocessing combinations are stop word deduction and negation (59.15%), negation, and punctuation removal (59.06%). The best performing combination for dataset1 is stemming and negation (69.44%). Moreover, for dataset 2, stop word deduction and reduction of letter repetition with 69.48% accuracy.

Table 5.20: Result of Two Preprocessing Technique Combinations Applied to Bengali Datasets

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
N+RP	54.84	55.65	69.32	50.40	54.84	69.31	59.06
RLR+RP	51.61	54.03	69.38	54.03	52.02	69.33	58.40
RLR+N	52.02	50.40	69.32	47.58	50.40	69.36	56.51
SWD+RP	50.00	48.39	69.24	55.24	53.63	69.32	57.64
SWD+N	52.82	50.81	69.39	58.06	54.44	69.35	59.15
SWD+RLR	52.82	52.82	69.27	49.60	52.42	**69.48**	57.74
S+RP	50.00	50.00	69.42	54.03	54.44	69.43	57.89
S+N	46.37	51.21	69.44	50.40	54.03	69.27	56.79
S+RLR	50.40	50.40	69.31	52.42	50.00	69.42	56.99
S+SWD	53.23	52.02	69.34	52.02	51.61	69.45	57.95
T+RP	54.44	52.82	69.35	50.81	50.40	69.31	57.86
T+N	50.40	49.19	**69.58**	53.23	55.24	69.42	57.84
T+RLR	54.03	53.63	69.27	58.06	54.84	69.41	**59.87**
T+SWD	53.23	51.61	69.28	51.21	48.79	69.31	57.24
T+S	47.58	50.40	69.25	52.82	52.82	69.34	57.04

5.4.1.4 Experiment with Three Preprocessing Technique Combinations

With the available preprocessing techniques considered in this research, 35 different combinations were found while experimenting with three pre-processing technique combinations on English datasets and are shown in Table 5.21. Case conversion, stemming, and stop word deduction combination on Dataset1 shows the highest overall classification accuracy (99.94%). By applying stemming, reduction of letter repetition, and punctuation removal combination on Dataset3, the second-highest classification accuracy of 99.93% is achieved.

Other promising combinations are case conversion, stemming, and reduction of letter repetition (91.06%). In addition, case conversion, stemming, and reduction of letter repetition (91.87%). Additionally, stop word deduction, negation, and punctuation removal (93.94%). Moreover, stop word deduction, Negation, and punctuation removal (93.94%), stemming, stop word deduction, and reduction of letter repetition (99.83%), stemming, reduction of letter repetition, and punctuation removal (85.00%). Overall, it is clear from Table 5.21 that the three preprocessing technique combinations improve classification accuracy, especially for dataset1 and dataset2. Therefore, the conclusion is that performance improvement from no preprocessing technique (Table 5.15) to three preprocessing techniques combination (Table 5.21) is 2.67%.

Table 5.21: Result of Three Preprocessing Technique Combinations on English Datasets

Preprocessing Formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
RLR+N+RP	84.15	83.33	96.62	84.85	84.85	94.05	79.00	78.00	83.16	85.33
SWD+N+RP	88.21	89.02	90.82	93.94	93.94	97.43	77.50	75.00	94.88	88.97
SWD+RLR+RP	87.80	87.80	94.78	87.88	87.88	91.09	82.00	79.00	96.05	88.25
SWD+RLR+N	85.77	86.59	90.55	81.82	81.82	93.30	81.00	77.50	98.36	86.30
S+N+RP	85.37	86.18	93.94	93.94	93.94	98.06	80.00	75.00	91.34	88.64
S+RLR+RP	88.62	89.84	95.39	81.82	81.82	92.78	84.50	85.00	99.93	88.86
S+RLR+N	88.21	86.59	99.25	84.85	84.85	93.29	75.50	71.50	96.13	86.69
S+SWD+RP	84.96	84.96	97.03	84.85	84.85	90.21	80.00	76.50	90.80	86.02
S+SWD+N	83.33	84.55	97.13	81.82	81.82	97.41	82.50	81.50	93.37	87.05
S+SWD+RLR	85.77	85.77	92.13	81.82	81.82	**99.83**	82.50	80.50	98.86	87.67
T+N+RP	89.02	90.24	98.14	87.88	87.88	99.53	80.50	79.00	90.16	89.15
T+RLR+RP	86.99	87.40	97.93	81.82	81.82	90.61	80.00	80.00	83.16	85.53
T+RLR+N	84.15	84.96	97.64	93.94	93.94	92.10	78.50	79.50	95.85	88.95
T+SWD+RP	87.80	88.21	99.01	84.85	84.85	93.91	74.50	76.50	94.88	87.17
T+SWD+N	86.18	86.99	99.41	78.79	78.79	95.14	79.50	79.50	91.01	86.15
T+SWD+RLR	89.84	90.24	92.74	78.79	78.79	98.23	75.00	78.50	98.36	86.72
T+S+RP	83.33	83.74	92.29	81.82	81.82	94.31	74.50	79.50	91.34	84.74
T+S+N	84.55	84.55	98.33	75.76	75.76	92.82	79.50	79.00	96.75	85.22
T+S+RLR	84.96	84.55	94.04	84.85	84.85	95.18	80.00	76.00	96.13	86.73
T+S+SWD	86.18	86.18	96.41	81.82	81.82	99.77	75.50	77.50	93.37	86.51
CC+N+RP	89.02	89.02	97.85	81.82	81.82	97.09	81.50	80.00	98.36	88.50
CC+RLR+RP	87.80	88.21	96.43	78.79	78.79	92.07	80.00	74.50	93.29	85.54

Preprocessing Formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
CC+RLR+N	89.43	89.84	90.43	87.88	87.88	97.53	76.50	75.50	92.60	87.51
CC+SWD+RP	86.59	86.59	95.43	78.79	78.79	95.74	80.00	79.00	96.75	86.41
CC+SWD+N	85.77	86.99	92.40	87.88	87.88	99.34	82.00	79.50	**99.93**	89.08
CC+SWD+RLR	86.99	86.99	99.61	87.88	87.88	98.20	74.50	73.50	95.36	87.88
CC+S+RP	85.77	85.37	94.50	87.88	87.88	93.22	78.50	78.50	98.86	87.83
CC+S+N	84.55	84.55	96.28	87.88	87.88	95.10	79.50	79.50	97.21	88.05
CC+S+RLR	91.06	91.87	99.18	84.85	84.85	91.92	79.50	78.00	99.85	89.01
CC+S+SWD	87.40	86.59	**99.94**	87.88	87.88	94.82	80.00	79.00	90.16	88.19
CC+T+RP	88.21	89.02	95.97	84.85	84.85	92.70	76.50	77.00	98.36	87.50
CC+T+N	87.40	88.62	98.75	87.88	87.88	98.07	81.50	83.00	93.77	**89.65**
CC+T+RLR	86.99	88.62	94.92	87.88	87.88	99.10	79.00	77.50	92.60	88.28
CC+T+SWD	85.37	85.37	93.85	81.82	81.82	95.89	76.50	74.00	95.36	85.55
CC+T+S	86.18	87.80	95.19	87.88	87.88	90.69	80.00	79.00	95.85	87.83

With the available preprocessing techniques considered in this research, 20 different combinations were achieved while experimenting with three pre-processing technique combinations on Bengali datasets and are shown in Table 5.22. Reduction of letter repetition, negation, and punctuation removal preprocessing combination has attained the highest classification accuracy (69.70%) for dataset1. In addition, stemming, stop word deduction, and punctuation removal have achieved the highest classification accuracy for dataset2 (69.80%).

Other promising combinations are tokenization, stop word deduction and punctuation removal (56.85%), Tokenization, stop word deduction, and punctuation (57.66%), tokenization, stemming, and stop word deduction (56.05%), tokenization, stemming, and stop word deduction (57.26%). Overall, the performance of created dataset improves more than the baseline dataset (dataset2) once three preprocessing technique combinations have been applied. Therefore, the conclusion is that performance improvement from no preprocessing technique (Table 5.16) to three preprocessing techniques combination (Table 5.22) is significant.

Table 5.22: Result of Three Preprocessing Technique Combinations Applied to Bengali Datasets

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
RLR+N+RP	52.02	57.26	**69.70**	50.00	50.00	69.46	58.07
SWD+N+RP	50.81	54.44	69.28	51.61	54.84	69.25	58.37
SWD+RLR+RP	52.82	54.84	69.28	52.82	56.45	69.40	59.27
SWD+RLR+N	54.44	56.05	69.42	50.81	50.81	69.33	58.48
S+N+RP	53.63	55.24	69.25	54.84	53.63	69.41	59.33
S+RLR+RP	47.98	49.19	69.41	54.03	47.98	69.27	56.31
S+RLR+N	50.40	49.60	69.29	53.63	54.44	69.27	57.77
S+SWD+RP	52.42	55.65	69.26	52.42	55.24	**69.80**	59.13
S+SWD+N	50.00	53.63	69.35	54.44	52.42	69.25	58.18
S+SWD+RLR	54.84	57.66	69.35	55.24	54.03	69.27	60.07
T+N+RP	50.81	47.58	69.48	54.44	54.44	69.30	57.68
T+RLR+RP	50.81	52.42	69.32	52.02	53.63	69.34	57.92

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
T+RLR+N	49.60	50.40	69.39	51.61	50.40	69.28	56.78
T+SWD+RP	56.85	57.66	69.32	51.61	53.23	69.42	59.68
T+SWD+N	46.37	50.00	69.39	50.81	50.00	69.28	55.98
T+SWD+RLR	54.03	56.05	69.62	47.18	48.39	69.26	57.42
T+S+RP	52.82	54.03	69.51	49.60	49.60	69.40	57.49
T+S+N	55.24	56.45	69.38	53.23	54.84	69.61	59.79
T+S+RLR	53.63	54.03	69.33	51.21	52.42	69.31	58.32
T+S+SWD	54.03	55.24	69.36	56.05	57.26	69.33	**60.21**

5.4.1.5 Experiment with Four Preprocessing Technique Combinations on English Datasets

Four Pre-processing techniques on English datasets were tested on 35 different combinations and are shown in Table 5.23. A good classification accuracy achieved with the application of four preprocessing techniques combinations to dataset1, dataset2, and dataset3 are case conversion, tokenization, stemming, stop word deduction combination (99.93%), case conversion, tokenization, negation, punctuation removal combination (99.86%), case conversion, tokenization, negation, reduction of letter repetition (99.93%) respectively. Overall, tokenization, stemming, negation, and punctuation removal combination give the highest average accuracy (90.25%) on three datasets.

The table reveals that datasets' performance improvement is different due to the level of noise in the dataset. Performance improvement on created datasets (3.1% and 5.1% for dataset1 and dataset2, respectively) was visible once four Pre-processing technique combinations were applied. However, the improvement of performance on baseline datasets is only 0.44% as the authors of the dataset well processed the data. The average improvement in performance is 2.9%.

Table 5.23: Result of Four Preprocessing Technique Combinations on English Datasets

Preprocessing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
SWD+RLR+N+RP	88.62	88.21	91.51	87.88	87.88	95.78	80.50	78.00	93.77	88.02
S+RLR+N+RP	85.37	86.59	99.62	84.85	84.85	91.16	78.00	78.50	95.36	87.14
S+SWD+N+RP	88.62	89.02	98.34	87.88	87.88	98.13	78.00	78.50	94.90	89.03
S+SWD+RLR+RP	83.74	85.37	97.45	90.91	90.91	95.93	80.50	79.50	97.21	89.06
S+SWD+RLR+N	83.74	85.37	99.11	87.88	87.88	90.83	83.00	79.00	90.37	87.46
T+RLR+N+RP	89.84	89.43	94.33	93.94	93.94	93.60	80.00	81.00	80.47	88.51
T+SWD+N+RP	88.62	89.84	97.71	90.91	90.91	93.52	82.00	79.50	96.05	89.90
T+SWD+RLR+RP	88.21	87.80	91.76	93.94	93.94	93.72	80.50	79.50	93.77	89.24
T+SWD+RLR+N	85.37	86.59	91.07	84.85	84.85	98.65	78.50	78.00	93.29	86.80
T+S+N+RP	86.99	88.21	92.03	93.94	93.94	99.73	78.50	79.00	99.93	**90.25**
T+S+RLR+RP	86.59	87.40	90.74	90.91	90.91	96.23	77.00	76.00	95.36	87.90
T+S+RLR+N	88.21	88.21	96.77	81.82	81.82	98.90	76.50	76.50	95.42	87.13
T+S+SWD+RP	84.96	86.18	98.40	78.79	78.79	97.67	71.50	74.00	94.90	85.02
T+S+SWD+N	92.28	93.50	94.78	81.82	81.82	97.10	77.00	77.00	98.86	88.24
T+S+SWD+RLR	85.77	86.18	93.83	90.91	90.91	99.45	78.00	75.00	83.16	87.02
CC+RLR+N+RP	86.18	88.21	90.10	87.88	87.88	90.89	80.00	76.50	90.19	86.43
CC+SWD+N+RP	84.15	86.18	95.43	90.91	90.91	96.10	75.50	76.00	96.13	87.92
CC+SWD+RLR+RP	88.21	89.02	99.66	87.88	87.88	96.86	77.00	76.50	95.42	88.71
CC+SWD+RLR+N	87.40	86.59	92.44	90.91	90.91	91.79	72.50	77.00	90.80	86.70
CC+S+N+RP	87.40	86.59	96.21	84.85	84.85	98.42	78.00	77.00	90.37	87.08
CC+S+RLR+RP	83.33	84.55	98.84	84.85	84.85	94.70	85.00	82.50	97.23	88.43
CC+S+RLR+N	85.77	86.18	93.07	81.82	81.82	95.61	77.50	73.50	88.81	84.90

Preprocessing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
CC+S+SWD+RP	83.33	84.55	93.16	78.79	78.79	92.20	79.00	79.50	90.16	84.39
CC+S+SWD+N	86.99	87.40	99.67	84.85	84.85	98.71	77.00	75.50	83.16	86.46
CC+S+SWD+RLR	85.37	85.77	94.10	90.91	90.91	99.05	80.00	77.00	80.47	87.06
CC+T+N+RP	85.77	85.37	97.50	87.88	87.88	**99.86**	81.00	78.50	93.29	88.56
CC+T+RLR+RP	86.18	86.99	95.66	81.82	81.82	93.45	81.50	80.50	90.19	86.46
CC+T+RLR+N	86.99	87.40	92.33	81.82	81.82	93.59	82.00	78.50	**99.93**	87.15
CC+T+SWD+RP	86.99	88.62	92.45	87.88	87.88	99.52	82.00	81.00	95.42	89.08
CC+T+SWD+N	85.37	87.40	96.63	87.88	87.88	92.83	83.00	81.50	90.80	88.14
CC+T+SWD+RLR	84.96	84.96	93.70	84.85	84.85	92.86	79.00	79.00	94.90	86.56
CC+T+S+RP	85.77	86.18	94.76	90.91	90.91	95.06	75.50	74.00	80.47	85.95
CC+T+S+N	88.21	86.99	97.98	87.88	87.88	98.86	83.50	84.00	91.73	89.67
CC+T+S+RLR	82.11	82.11	95.14	84.85	84.85	92.68	83.00	83.00	91.01	86.53
CC+T+S+SWD	85.77	86.99	**99.93**	84.85	84.85	97.39	82.00	79.50	93.29	88.29

Fifteen different preprocessing technique combinations were found while experimenting with four pre-processing techniques on Bengali datasets and are shown in Table 5.24. A good classification accuracy achieved with the application of four Pre-processing techniques combinations to Bengali dataset1, and dataset2 are 69.57% (tokenization, reduction of letter repetition, stemming, negation, and punctuation removal combination) and 69.59% (stemming, stop word deduction reduction of letter repetition, and punctuation removal combination), respectively. Overall, stemming, stop word deduction reduction of letter repetition, and negation combination gives the highest average accuracy (59.4%) on three datasets.

The table also shows that the performance improvement is different due to noise levels in the datasets. The improvement is easily visible on created datasets once four Pre-processing technique combinations were applied. However, the improvement of performance on baseline datasets is very insignificant as the authors of the dataset well processed the data.

Table 5.24: Results of Four Preprocessing Technique Combinations Applied to Bengali Datasets

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
SWD+RLR+N+RP	54.03	52.42	69.37	51.61	51.61	69.31	58.06
S+RLR+N+RP	55.65	50.00	69.29	50.81	52.02	69.35	57.85
S+SWD+N+RP	53.63	53.63	69.36	54.03	53.23	69.42	58.88
S+SWD+RLR+RP	49.60	47.18	69.27	56.05	54.84	**69.59**	57.76
S+SWD+RLR+N	53.63	54.44	69.31	54.44	55.24	69.34	**59.40**
T+RLR+N+RP	50.00	50.81	**69.57**	47.58	46.77	69.41	55.69
T+SWD+N+RP	50.81	52.42	69.36	53.23	56.85	69.31	58.66
T+SWD+RLR+RP	54.03	52.82	69.36	50.00	51.61	69.30	57.85
T+SWD+RLR+N	53.23	53.23	69.31	55.24	54.84	69.27	59.19
T+S+N+RP	52.02	54.44	69.47	52.42	56.85	69.26	59.08
T+S+RLR+RP	50.40	52.42	69.49	53.23	51.61	69.26	57.74
T+S+RLR+N	55.24	52.42	69.39	51.61	54.44	69.30	58.73
T+S+SWD+RP	50.00	49.60	69.30	51.61	50.81	69.31	56.77
T+S+SWD+N	50.40	54.44	69.27	52.42	52.02	69.46	58.00

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
T+S+SWD+RLR	52.82	56.05	69.33	47.98	50.81	69.24	57.71

5.4.1.6 Experiment with Five Preprocessing Technique Combinations

This section covers the experiment result of five pre-processing technique combinations in English (Table 5.25) and Bengali datasets (Table 5.26). In Table 5.25, the best performing preprocessing technique combinations are 99.44% (stemming, stop word deduction, reduction of letter repetition, negation, and punctuation removal combination), 99.30% (case conversion, tokenization, Stemming, Stop Word Deduction, and Negation combination), and 99.85% (stemming, stop word deduction, reduction of letter repetition, negation, and punctuation removal combination) for dataset1, dataset2, and dataset3 respectively.

On average, the best performance on five preprocessing technique combinations is shown by case conversion, tokenization, stop word deduction, reduction of letter repetition, and negation combination (89.75%). All the preprocessing technique combinations used in this category have shown relatively better performance than the previous mixes. The performance improvement in dataset1, dataset2, and dataset3 is 2.8%, 4.2% and 1.24% respectively. However, the average performance improvement is 2.7%.

Table 5.25: Result of Five Preprocessing Technique Combinations on English Datasets

Preprocessing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
S+SWD+RLR+N+RP	85.77	86.59	**99.44**	78.79	78.79	91.56	75.5	76	**99.85**	85.81
T+SWD+RLR+N+RP	88.21	88.21	93.33	81.82	81.82	93.64	79.5	79.5	92.6	86.51
T+S+RLR+N+RP	85.77	85.77	91.05	96.97	96.97	97.33	79.5	78.5	90.8	89.18
T+S+SWD+N+RP	86.59	86.18	98.73	87.88	87.88	93.3	79	79	97.21	88.42
T+S+SWD+RLR+RP	84.96	84.96	98.86	84.85	84.85	90.82	77	77	95.85	86.57
T+S+SWD+RLR+N	87.8	88.62	91.35	87.88	87.88	98.6	82	82.5	80.47	87.46
CC+SWD+RLR+N+RP	89.02	89.43	92.72	84.85	84.85	90.43	78.5	78.5	93.37	86.85
CC+S+RLR+N+RP	84.15	84.96	97.31	87.88	87.88	91.49	81	81	91.49	87.46
CC+S+SWD+N+RP	83.74	84.96	96.29	90.91	90.91	95.47	81.5	79.5	95.85	88.79
CC+S+SWD+RLR+RP	85.77	85.77	97.58	84.85	84.85	92.16	76	78	91.73	86.30
CC+S+SWD+RLR+N	85.77	87.4	93.06	87.88	87.88	92.35	73.5	79	94.88	86.86
CC+T+RLR+N+RP	88.62	89.02	93.6	84.85	84.85	97.43	85.5	83	96.13	89.22
CC+T+SWD+N+RP	83.33	83.33	93.77	90.91	90.91	95.23	80	79	93.37	87.76
CC+T+SWD+RLR+RP	89.43	89.02	96.55	87.88	87.88	97.85	81	77.5	98.86	89.55
CC+T+SWD+RLR+N	86.59	86.59	93.66	90.91	90.91	97.38	83	81.5	97.21	**89.75**
CC+T+S+N+RP	88.21	86.99	91.37	87.88	87.88	91.65	77	75.5	94.88	86.82
CC+T+S+RLR+RP	84.55	85.37	90.11	78.79	78.79	97.97	79	77	96.05	85.29
CC+T+S+RLR+N	84.15	84.15	90.7	87.88	87.88	98.2	79	78	98.36	87.59
CC+T+S+SWD+RP	90.24	90.65	93.61	81.82	81.82	91.12	76	77.5	92.6	86.15
CC+T+S+SWD+N	88.21	89.43	94.87	84.85	84.85	**99.3**	80	79.5	90.19	87.91
CC+T+S+SWD+RLR	86.18	86.59	92.36	81.82	81.82	92.53	76	73	96.75	85.23

As per the results shown in Table 5.26, the average performance achieved by applying the fifth preprocessing technique combinations is 58.2% and 57.8% for dataset1 and dataset2. However, the maximum performance achieved on dataset1 and dataset2 are by tokenization, stemming, stop word deduction, reduction of letter repetition, and punctuation removal combination (69.66%), and tokenization, stemming, reduction of letter repetition, negation, and punctuation removal combination (69.63%) respectively. For this category, six different combinations were achieved. Surprisingly all the combinations show very similar classification accuracy with LSTM. However, the improvement with the fifth preprocessing technique combinations is not very significant compared to the fourth preprocessing technique combinations. This insignificance is due to the Bengali preprocessing tools and resources, as these cannot work after a certain level of preprocessing.

Table 5.26: Result of Five Preprocessing Technique Combinations Applied to Bengali Datasets

Preprocessing formation	Dataset1			Dataset2			Average
	NB	SVM	LSTM	NB	SVM	LSTM	
S+SWD+RLR+N+RP	58.47	55.65	69.3	48.39	46.37	69.4	57.93
T+SWD+RLR+N+RP	56.85	56.45	69.3	50	50.4	69.35	**58.73**
T+S+RLR+N+RP	50	51.21	69.26	52.82	52.02	**69.63**	57.49
T+S+SWD+N+RP	51.61	50.4	69.27	50.81	54.44	69.26	57.63
T+S+SWD+RLR+RP	49.6	49.6	**69.66**	54.44	54.44	69.24	57.83
T+S+SWD+RLR+N	51.21	49.6	69.41	56.45	53.63	69.35	58.28

5.4.1.7 Experiment with Six Preprocessing Technique Combinations

The study found only seven different combinations using six preprocessing techniques on English datasets. The combinations are presented in Table 5.27. On average, case conversion, tokenization, stemming, reduction of letter repetition, negation, punctuation removal combination shows the best result with 90.60%

accuracy. The best performing other combination accuracy is 88.82% (case conversion, tokenization, stemming, stop word deduction, negation, punctuation removal), 88.29% (case conversion, stemming, stop word deduction, reduction of letter repetition, negation, punctuation removal). Besides, other combinations with the best accuracy are 88.22% (case conversion, tokenization, stemming, stop word deduction, reduction of letter repetition, punctuation removal), 87.72% (case conversion, tokenization, stemming, stop word deduction, reduction of letter repetition, negation), 87.54% (tokenization, stemming, stop word deduction, reduction of letter repetition, negation, punctuation removal).

From among all the classifiers, LSTM is performing best for all the preprocessing combinations and the datasets. The average performance of dataset2 is better than other datasets with an accuracy of 91.26%, and then the next highest performance is shown by dataset1 with an accuracy of 90.54%. The average accuracy of dataset3 is 82.07%. The result indicates that created datasets outperform the baseline dataset (Dataset3) once six preprocessing combinations are applied. Moreover, six preprocessing combination works better on dataset1 once case conversion+tokenization+stemming+ reduction of letter repetition+negation+punctuation removal combination applied on an individual dataset. It could be concluded from the table that the more preprocessing combinations are applied, the fewer improvements are seen in the accuracy. The reason is that once low order preprocessing is already done, the higher-order preprocessing gets significantly less noisy data to preprocess. Thus, the improvement becomes very insignificant.

Table 5.28 shows the result of six preprocessing technique combinations applied to Bengali datasets. This study tested with different combinations and got a very similar result with one exception shown in Table 5.28. The result indicates that, on average, dataset1 is better than dataset2. This preprocessing outperforms other combinations that have been removed from this table. Moreover, LSTM outperforms other classifiers for both the data sets. The conclusion for six preprocessing technique combinations for Bengali data is similar to that of English data. However, the improvement is significant in comparison to the fifth preprocessing technique combination.

Table 5.27: Result of Six Preprocessing Technique Combinations on English Datasets

Pre-processing formation	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
T+S+SWD+RLR+N+RP	86.59	85.37	97.20	90.91	90.91	97.12	76.50	71.50	91.73	87.54
CC+S+SWD+RLR+N+RP	89.02	89.02	92.48	87.88	87.88	**99.35**	80.00	78.00	91.01	88.29
CC+T+SWD+RLR+N+RP	84.55	86.59	92.07	84.85	84.85	93.62	76.00	75.00	83.16	84.52
CC+T+S+RLR+N+RP	87.80	89.02	**99.45**	93.94	93.94	97.97	79.50	80.00	**93.77**	**90.60**
CC+T+S+SWD+N+RP	91.46	91.87	95.36	90.91	90.91	90.53	78.50	78.50	91.34	88.82
CC+T+S+SWD+RLR+RP	86.99	86.18	98.21	84.85	84.85	97.67	80.00	81.50	**93.77**	88.22
CC+T+S+SWD+RLR+N	87.40	87.40	97.30	90.91	90.91	91.75	73.50	77.00	93.29	87.72

Table 5.28: Result of Six Preprocessing Technique Combinations Applied to Bengali Datasets

Preprocessing formation	Dataset1			Dataset2		
	NB	SVM	LSTM	NB	SVM	LSTM
T+S+SWD+RLR+N+RP	55.65	52.82	**69.29**	62.34	65.16	**83.11**

5.4.1.8 Average Performance on Different Preprocessing Techniques for English and Bengali Datasets

This section presents average results ((Table 5.29) generated by different preprocessing techniques and their combinations for English and Bengali datasets. It was evident from Table 5.29 that, with different combinations, it behaved differently. However, seven preprocessing technique combinations are optimal preprocessing techniques with an average classification accuracy of 89.1%. Besides, some other technique combinations could be considered optimal such as six and second, with a sentiment classification accuracy of 87.95% and 87.86%, respectively. The general pattern of improvement in performance for preprocessing technique combinations is that the more, the better.

It is also apparent from Table 5.30 that the performance improvement applying preprocessing techniques combinations on Bengali datasets is not contiguous. Performance improvement was highly varied from combinations to combinations. However, the more the preprocessing, the more is an improvement. This theorem came true. The highest average performing combination is the six preprocessing technique combinations. With dataset2, this combination achieved an accuracy of 83.11%. At the same time, dataset1 has achieved the highest accuracy of 69.43% on a single technique. These results reveal that due to the insufficiency of preprocessing tools for the Bengali language, the performances of the classifiers are not uniform. For this particular research, six preprocessing technique combinations could be considered as an optimal combination.

Table 5.29: Average Performance of Preprocessing Techniques for English Datasets

Experiment setup	Dataset1			Dataset2			Dataset3			Average
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM	
No preprocessing	82.93	83.33	93.14	81.82	81.82	95.02	80.00	78.00	90.37	85.16
Single Technique	85.77	86.35	94.20	86.58	86.58	95.32	80.79	79.43	**95.69**	87.86
Two Techniques	86.49	86.90	94.57	86.44	86.47	94.33	79.81	78.57	91.56	87.24
Three Techniques	86.68	87.06	95.84	84.85	84.85	95.16	79.07	78.06	94.64	87.35
Four Techniques	86.38	87.03	95.32	87.01	87.01	95.74	79.24	78.27	92.19	87.57
Five Techniques	86.53	86.86	94.30	86.29	86.29	94.56	79.02	78.57	94.22	87.40
Six Techniques	87.69	87.92	96.01	89.18	89.18	**95.43**	77.71	77.36	91.15	87.95
Seven Techniques	84.96	85.37	**98.63**	90.91	90.91	90.86	87.50	80.00	92.60	**89.08**

Table 5.30: Average Performance of Preprocessing Techniques on Bengali Datasets

Experiment setup	Dataset1			Dataset2		
	NB	SVM	LSTM	NB	SVM	LSTM
No-preprocessing	53.23	51.61	69.26	55.60	52.82	69.78
Single Technique	51.48	52.69	**69.43**	50.34	52.22	69.38
Two Techniques	51.59	51.56	69.34	52.66	52.66	69.37
Three Techniques	52.18	53.87	69.38	52.38	52.70	69.36
Four Techniques	52.37	52.42	69.36	52.15	52.90	69.34
Five Techniques	52.96	52.15	69.37	52.15	51.88	69.37
Six Techniques	55.65	52.82	69.29	62.34	65.16	**83.11**

5.4.1.9 Maximum Performance on Different Types of Preprocessing Techniques for English and Bengali Datasets

The following tables (Table 5.31 and Table 5.32) present the maximum performance of classifiers on both types of datasets. As per Table 5.31, it is evident that seven preprocessing technique combinations show maximum classification accuracy (98.63%) for all the algorithms and English datasets. Also, the performance of dataset1 improved more once the preprocessing technique combinations were applied. The reason for the improvement is the amount of noise and outliers in the dataset. As per Table 5.32, for Bengali datasets, six preprocessing technique combinations was outperforming other combinations. However, the performance of dataset2 improved more once the preprocessing technique combinations were applied due to the same reason as the English dataset. Overall, for all the experiments, LSTM gave better classification accuracy once preprocessing was applied.

Table 5.31: Maximum Performance of Classifiers for English Datasets

Experiment Setup	Dataset1			Dataset2			Dataset3		
	NB	SVM	LSTM	NB	SVM	LSTM	NB	SVM	LSTM
Maximum	87.69	87.92	**98.63**	90.91	90.91	**95.74**	87.50	80.00	**95.69**
Best Preprocessing combination	Six	Six	**Seven**	Seven	Seven	**Four**	Seven	Seven	**Single**

Table 5.32: Maximum Performance of Classifiers for Bengali Datasets

Experiment Setup	Dataset1			Dataset2		
	NB	SVM	LSTM	NB	SVM	LSTM
Maximum	55.65	53.87	**69.43**	55.65	52.90	**83.11**
Best Preprocessing combination	Six	Three	**Single**	No-preprocessing	Four	**Six**

5.4.1.10 Comparison with State-of-Art Research That Emphasized Preprocessing

Table 5.33 represents a comparison with state-of-art research on preprocessing techniques. The comparative evaluation shows that the performance of English and Bengali baseline datasets by this research is 95.69% and 83.11%, respectively, which outperformed state-of-art research by 5.19% (Hameed et al., 2020) and 0.91% (Bhowmik et al., 2021). Moreover, the performance of the English student feedback dataset outperformed state-of-art research (Lwin et al., 2020) by 1.63%. Moreover, the performance of the Bengali student feedback dataset has not been compared due to a lack of research on this domain. Overall, it could be concluded that preprocessing plays an essential role in performance enhancement, especially when doing SA at the concept level. Therefore, future studies could consider many other preprocessing technique combinations and hope to achieve more satisfactory performance improvement.

Table 5.33: Comparison with State-of-Art Research on Preprocessing Techniques

Datasets (Language)	References	Performance (Accuracy/precision)
Students' feedback (English)	Lwin et al., 2020	97.00%
	This research (dataset1&2)	**98.63%**
IMDB (English)	Hameed et al., 2020	90.50%
	This research (dataset3)	**95.69%**
Students' feedback (Bengali)	No dataset found	NA
	This research	**69.43%**
Cricket dataset (Bengali)	Bhowmik et al., 2021	82.20%
	This research (Dataset2)	**83.11%**

5.5 Results Summary and Discussion

5.5.1 Results Summary and Discussion on the Performance of BanglaSenticNet, Polarity Lexicon, and MCSAlgo

The main objective of this thesis is to create a Bengali knowledge base, Bengali polarity lexicon, and propose an algorithm to support MLSA at the concept level. Therefore, this thesis has created a knowledge base (BanglaSenticNet), polarity lexicon, and proposed an algorithm (MCSAlgo). These resources and algorithms are evaluated using three English datasets and two Bengali datasets. The evaluated results are shown in Table 5.1- Table 5.4.

Table 5.1 exposes that the two knowledge bases show very similar performance with the limited number of concepts in BanglaSenticNet compared to SenticNet 5. SenticNet 5 outperforms BanglaSenticNet only on one evaluation parameter (accuracy). The accuracy achieved using BanglaSenticNet on Bengali datasets is 71.14%, and using SenticNet 5 on English datasets is 79.13%. The precision of Bengali and English lingual datasets are 72% and 66%, respectively. Moreover, a comparison with state-of-art research in Table 5.4 revealed that BanglaSenticNet outperforms existing feature-based work by approximately 6%. The rationale for these improvements is 1) Applying a large set of preprocessing technique combinations and 2) Applying different feature and concept extraction technique combinations. The comparison is made with feature-based work due to the insufficient proof of having any Bengali knowledge bases in the literature review. However, with all these limitations, it could be concluded from the tables that the performance with BanglaSenticNet is satisfactory and significant compared to SenticNet 5 and state-of-art research.

The performances of polarity lexicons are also very close, even having fewer concepts in the Bengali polarity lexicon. Table 5.2 shows the performance with or without using Bengali and English Polarity Lexicons. The table shows that performance improvement after adopting the polarity lexicon is 0.9% and 18.79% for both lingual datasets. Moreover, the comparison on polarity lexicon in Table 5.4 shows that the created English polarity lexicon outperforms recent polarity-based research on the IMDB dataset. This research has achieved an accuracy of 99.32%, where recent research has achieved 92.80%. Besides, on the Cricket dataset, created Bengali polarity lexicon outperforms recent polarity-based research. This research has achieved an accuracy of 93.16%, where existing research has achieved 88%. Bengali polarity lexicon outperformed for two reasons: 1) New lexicons are multidomain and enriched with more widely used concepts 2) Existing research has used a feature-based approach; however, this research has used a concept-based approach. Therefore, this research would like to recommend using polarity lexicon, especially Bengali, in MLSA research, which will surely add a significant contribution in future research.

The knowledge bases and polarity lexicons were tested on all three English datasets and two Bengali datasets using four different classifiers such as NB, SVM, LSTM, and the proposed algorithm (MCSAlgo). It is also evident from Table 5.3 that MCSAlgo performs well with SenticNet 5, having very similar classification accuracy compared to NB, SVM, and LSTM. Besides, it outperforms NB and SVM with BanglaSenticNet. The reason behind the better performance of MCSAlgo is the method it uses for classifications.

The study also evaluated the significance of the proposed algorithm shown in Table 5.3. The table shows that on the scale of time complexity and in the worst-case, SVM and MCSAlgo have the same time complexity of O (n3). On the same scale, LSTM is shown to be a slower algorithm with complexity O (n4*c). The performance of MCSAlgo remains stable on both lingual data.

The comparative analysis of performance with state-of-art research in Table 5.4 shows no customized concept-level SA algorithms. Therefore, this research has compared the performance of existing work that emphasized standard algorithms for the concept-level SA and outperformed by 76% and 99.32% accuracy on both lingual data. The performance improvement compared to state-of-art research is 5% and 7.32% for both lingual data. The reason for outperforming is due to the application of proper resources and techniques. The above analysis shows that the performance of MCSAlgo is significant. It could be widely used for MCSA and improve its performance by rethinking and regenerating the logic.

5.5.2 Results Summary and Discussion on the Performance of Different Datasets

This research has created three datasets mentioned in Section 3.5 to meet one of its objectives and support its experimentations. The datasets are tested with different knowledge bases, algorithms (both customized and standard), preprocessing techniques and their combinations, and feature or concept extraction techniques. The experiment in Table 5.5 shows that both lingual datasets are interchangeably better. However, English dataset1 and Bengali dataset2 are better performing datasets. In Bengali datasets, the baseline dataset is working better due to the structuredness and good labeling of data. The comparison with state-of-art research in Table 5.6 shows no Bengali dataset on student feedback, and the English student feedback dataset is primarily survey-based and does not analyze the sentiment at the concept level.

Therefore, this research created a Bengali and English student feedback dataset and analyses the sentiments at the concept and feature level. However, the comparison is mainly made on feature-level state-of-art research. The comparison shows that the performances of created datasets are outperforming state-of-art datasets by 2.94%.

The improvements were achieved due to applying different resources and techniques (as stated earlier) to these datasets. This research also used the same algorithms, techniques, and resources on baseline datasets to test whether the created dataset's evaluation is worthy. The baseline datasets also outperform the existing research by 2.93% (English dataset) and 10.96% (Bengali dataset). The results indicate that the datasets and methodology applied in this study are well suited and worthy of further use.

5.5.3 Results Summary and Discussion of Performance on Feature and Concept Extraction Technique and Their Combinations

This research aims to analyze and evaluate the best extraction techniques among feature (s), concepts, and their combinations for MLSA. The objective is achieved by applying different extraction techniques individually and in conjunction. Therefore, this section presents the results summary and discussion of different feature and concept extraction techniques and their combinations on English and Bengali datasets.

In Section 5.3, the experiment was conducted on both English and Bengali data sets. The distinct feature and concept extraction technique used in the experiment are TD - IDF (Simple), unigram, bigram, trigram, parts of speech, concept (Simple), and their combinations are unigram + bigram, bigram + trigram, Unigram + bigram +

trigram, concept + unigram, concept + bigram, concept + trigram. In each of these experiments, the data were processed applying all the preprocessing techniques mentioned in Section 5.4.

From among all the features (applying individually), a simple concept-based approach showed the highest classification accuracy for English datasets, whereas, for Bengali datasets, bigram shows the highest classification accuracy. The highest performance for both English and Bengali datasets is shown by concept + Trigram in feature combinations. On average, combining the features shows improved performance, except in some cases.

For instance, it is observed that combining unigram with bigram and trigram improves the performance to a reasonable extent for English data. This result is also true for Bengali datasets. It is also found that combining features with concepts sometimes increases the classification accuracy and vice-versa.

When features or concepts were extracted from whole English datasets, the most promising result is shown by a simple concept-based approach, whereas, for Bengali datasets, the same result is shown by unigram and bigram feature combinations. It is also mentionable that the performance of feature-based is better than concept-based SA. However, in the case of an individual classifier, concept-based is better. However, on average, the concept extraction approach shows the highest classification accuracy than the feature extraction approach.

The three classifiers (NB, SVM, and LSTM) were used for classifying different data sizes of the datasets. The results prevail that the increasing data size of the datasets sometimes increases the classification accuracy and vice-versa. This impact may be due to the lack of proper application of feature or concept extraction techniques. For both lingual datasets, different data sizes, feature, and concept extraction techniques, LSTM is found to be better with the highest average classification accuracy.

Moreover, it is observed that increasing data sample size in experimentation increases the performance. In addition, the feature-based approach is found to be better than the concept-based approach for Bengali datasets due to the lack of necessary concept dictionaries, and the result indicates that, for large datasets, this gap will be too minor. It was found in some cases; these extraction approaches are interchangeably better.

For both types of extraction methods, the created dataset (dataset1 and dataset2) was found better than the baseline dataset (dataset3) for English data. In addition, the baseline dataset (dataset2) is found to be outperforming Bengali data. The result also shows that the concepts extraction approach works better for highly preprocessed data (dataset3) than created datasets.

Moreover, Table 5.12 reveals that the concept+trigram with 98.87% accuracy is the optimal performing combination on English datasets. The table also shows that, after applying different feature extraction techniques and their combination on the Bengali dataset, bigram could be considered an optimal feature technique as it has achieved a maximum of 74.37% accuracy. However, the performance of the concepts with LSTM is better for both datasets (accuracy of 86.65% and 93.16%).

The comparison in Table 5.13 shows, this research has outperformed state-of-art research on both feature and concept extractions by 7.86% and 7.32%, respectively. However, considering the Bengali baseline dataset, state-of-art research on feature extraction has outperformed this research by 7.83%. Moreover, due to the lack of research on student feedback datasets in Bengali, it could not compare this research with state-of-art research. Overall, comparing with state-of-art research, we could conclude that concept extraction techniques outperformed feature extraction techniques in current and state-of-art research. Therefore, this research recommend to use concept extraction techniques in future research.

5.5.4 Results Summary and Discussion for Preprocessing Technique Combinations Using English and Bengali Data

As stated earlier, one of the objectives of this thesis is to examine the preprocessing techniques and their combinations for MLSA and search for the optimal one. As per the discussion above, the data were experimented with using different preprocessing combinations on both English and Bengali data sets (Section 5.4.1.1-5.4.1.7). The combinations were named as no-preprocessing, single technique, two techniques, three techniques, four techniques, five techniques, six techniques, seven techniques. Here, a single technique means only one preprocessing technique (such as negation or stemming, etc.) applied to the dataset; two techniques means using two preprocessing techniques in combination (such as negation and stemming) to process the datasets and so on. The results are shown in Table 5.14-5.28.

Preprocessing techniques used in this thesis are removing punctuation, negation, reduction of letter repetition, stop word deduction, stemming, and tokenization for both English and Bengali datasets. For Bengali data, only six techniques were used, and the case conversion technique is omitted from the experiments and analysis as no case exists in Bengali text. For both data types (English and Bengali), applying two and five preprocessing technique combinations found to have minimal impact on any of the algorithms results. However, without preprocessing (Table 5.15), the average classification accuracy obtained by different classifiers for English datasets is NB (81.58%), SVM (81.05%), and LSTM (92.84%). All types of preprocessing techniques (Table 5.15) applied in combination cause the average classification accuracy to be NB (87.79%), SVM (85.42%), LSTM (95.41%). These results indicate that applying all types of preprocessing techniques at a time help increase the average classification accuracy by 4%-6%. Besides, created datasets work better with an average accuracy of 86.34% (without preprocessing) and 90.27% (with preprocessing) than the baseline dataset (dataset3- accuracy 82.79% without preprocessing and 86.70% with preprocessing).

Moreover, without preprocessing (Table 5.16), the average classification accuracy obtained by different classifiers for Bengali datasets is NB (54.44%), SVM (52.22%), and LSTM (69.52%). Moreover, all types of preprocessing techniques applied in combination cause the average classification accuracy to be NB (58.99%), SVM (58.99%), LSTM (76.2%). These results indicate that applying all preprocessing techniques in combinations increases the average classification accuracy due to more sentiment terms on preprocessed data. Table 5.16 also reveals that dataset2 works better than dataset1 by 1.4% (without preprocessing) and 10.95% (with preprocessing). The results indicate that a well-processed dataset is essential for MLSA.

With a single processing technique (Table 5.17), the average classification accuracy for English datasets using a different classifier is NB (86.92%), which outperforms no-preprocessing result by 5%, SVM (84.13%), which outperform no-preprocessing result by 3%, LSTM (95.07%) which outperforms no-preprocessing result by 3%. With a single processing technique (Table 5.18), the average classification accuracy for Bengali datasets using different classifier is NB (50.91%), which shows less accuracy than no-preprocessing result by 3%, SVM (52.45%), which outperform no-preprocessing result by 1%, LSTM (69.40%), which shows less accuracy than no-preprocessing result by 1%.

With the application of two processing techniques combination (Table 5.19), the average classification accuracy for English datasets using different classifier is NB (84.25)%, which outperform no-preprocessing result by 3%, SVM (83.98%), which outperform No-preprocessing result by 3%, LSTM (93.49%) which outperform no-preprocessing result by 1%. Similarly, for Bengali datasets, the average classification accuracy using different classifier is NB (52.12%), which outperform No-preprocessing result by -2%, SVM (52.11%) & LSTM (69.36%), both show less accuracy than no-preprocessing result by 1%. The results in Table 5.20 indicate that, preprocessing alone is not sufficient for performance enhancement.

The average classification accuracy for English datasets using three processing techniques (Table 5.21) combination and different classifiers is NB (83.53%), SVM (83.32%), and LSTM (95.21%). The classifiers outperform the no-preprocessing result by NB (2%), SVM (3%), and LSTM (3%). At the same time, the performance of Bengali datasets (Table 5.22) using different classifier is NB (52.28%), which shows less accuracy than no-preprocessing result by 2%, SVM (53.29%), which outperform no-preprocessing result by 1%, and LSTM (69.37%), which shows less accuracy than no-preprocessing result by 1%.

In the case of four processing techniques (Table 5.23) combination applied to English datasets, the average classification accuracy for the different classifier is NB (84.21%), which outperform the no-preprocessing result by 3%, SVM (84.10%), which outperform no-preprocessing result by 3%, LSTM (94.42%) which outperform no-preprocessing result by 2%. Meanwhile, the performance of the different classifier for Bengali datasets with four preprocessing technique (Table 5.24) is NB (52.26%), which shows less accuracy than No-preprocessing result by 2%, SVM (52.66%), which outperform no-preprocessing result by 1% and LSTM (69.35%), which shows less accuracy than no-preprocessing result by 1%.

With the use of five processing techniques (Table 5.25) combination in English datasets, the average performance for the classifiers are NB (83.95%) and SVM (83.91%), both the techniques outperform no-preprocessing result by 3%, LSTM (94.36%) which outperform no-preprocessing result by 2%. With similar techniques applied to Bengali datasets (Table 5.26), the average performance of classifiers are NB (52.26%), SVM (52.66%), and LSTM (69.35%), all those classifiers show accuracy less than no-preprocessing result by 2%, 1%, and 1% respectively.

Six processing techniques (Table 5.27) combination has shown results that are more promising on English datasets. The average classification accuracy of classifiers is NB (84.86%), and SVM (84.82%) both outperform the no-preprocessing result by 4%, and LSTM (94.20%) outperform the no-preprocessing result by 2%. Since, in the Bengali dataset, case conversion was not applied; thereby, six processing techniques combination result presented at the beginning of this section.

Overall, adopted preprocessing techniques and their combination has produced better classification accuracy with NB, SVM, and LSTM and found LSTM as the best classifier. From testing, it is evident that the performance increases once different preprocessing techniques are applied in combinations. For example, the result achieved using simply tokenization improves once other techniques such as stemming were added with tokenization and so on. This result found to vary for some preprocessing combinations, such as punctuation removal and stop word deduction, as these techniques deduct some essential words that may express sentiments.

Table 5.29 reveals that, with different combinations, it behaved differently. However, seven preprocessing technique combinations are optimal preprocessing techniques with an average classification accuracy of 89.1%. The general trend of performance improvement is that the more, the better. Moreover, Table 5.30 shows that the performance improvement on Bengali datasets is not contiguous (vary from combinations to combinations) if applied to preprocessing technique combinations. However, six preprocessing technique combinations could be considered as an optimal combination with an accuracy of 83.11%. Moreover, it is also seen from the resulting trend of Table 5.30, due to the insufficiency of preprocessing tools for the Bengali language, the performances of the classifiers are not uniform.

Comparing the results with state-of-art research (Table 5.33), the performance of English and Bengali baseline datasets is 95.69% and 83.11%, respectively, which outperformed state-of-art research by 5.19% and 0.91%. Moreover, the performance of the English student feedback dataset outperformed state-of-art research by 1.63%. However, due to a lack of research on the Bengali student feedback dataset, no comparison has been made. Overall, it could be concluded that preprocessing plays an essential role in performance enhancement, especially when doing SA at the concept level. Therefore, this research recommend to use the suitable preprocessing techniques as much as possible to dig inside the text easily with optimal results.

5.6 Results and Discussion Chapter Summary

In this chapter, the knowledge bases such as SenticNet 5 and BanglaSenticNet were evaluated on all three English datasets and two Bengali datasets using NB, SVM, LSTM, and MCSAlgo, respectively. The detailed evaluation shows that the accuracy for all the classifiers is better using SenticNet 5 on English data than the BanglaSenticNet, which is applied to Bengali data. The reason is the lack of proper preprocessing and concept extraction techniques in Bangla and the number of concepts in the knowledge base. However, it is promising that the results are very close.

This chapter then evaluated the performance of the proposed algorithm about knowledge bases, polarity lexicons, datasets, and other standard algorithms. A comparative evaluation of the algorithm is shown between proposed and state-of-art algorithms. This chapter also showed the performance of knowledge bases, polarity lexicons, preprocessing techniques, and feature or concept extraction techniques on created and baseline datasets (Bengali and English). A comparison with the performance of existing works on datasets evaluation is also shown in this section.

This chapter then describes the experimental results obtained for feature and concept extraction techniques and their combination to search for optimal feature or concept extraction techniques or their combinations on multilingual data (English and Bengali lingual). The experiment used a distinct feature and concept extraction technique, namely TD - IDF (Simple), unigram, bigram, trigram, parts of speech, concept (Simple) and their combinations such as unigram + bigram, bigram + trigram, unigram + bigram + trigram, concept + unigram, concept + bigram, concept + trigram. An evaluation between this and existing research that emphasized feature and concept extraction techniques is shown. Besides, the performance comparison between features and concepts is also evaluated.

This chapter includes the experiment results obtained using different preprocessing techniques and their combinations and recommended the optimal preprocessing techniques with both English and Bengali data sets. The more the preprocessing techniques were combined, the more the performance achieved for both types of data. For instance, applying all the techniques in combination has produced better classification accuracy with NB, SVM, LSTM and found LSTM as the best classifier. The next chapter includes conclusive words along with the future direction of this research.