CHAPTER 2

LITERATURE REVIEW

This research aims to find the resources, algorithms, and optimal techniques, for sentiment analysis by analyzing the multilingual (Bengali and English) text (student education-related feedback) collected from Facebook. Therefore, this chapter provides a brief idea of NLP tasks, as the problem addressed by this thesis (sentiment analysis) is associated with natural language processing (NLP). This chapter also presents a thorough discussion of sentiment analysis (SA), its concept, types, different tasks related to sentiment analysis and different approaches (especially the idea of concept-based approach and its impact over other approaches) to deal with those sentiments. This chapter then discussed on Bengali language structure, problems related to SA, and related SA work. The next section of this chapter emphasizes on related work correspond to multilingual data and different approaches related to multilingual sentiment analysis (MLSA).

Moreover, as this thesis is highly concerned with the lexicon and knowledge base creation, the next section of the chapter gives a detailed picture of the theory and the tasks done so far based on the lexicons and knowledge bases. Besides, this chapter presents a comprehensive study of different customize and machine learning algorithms (such as NB, SVM, and LSTM) of MLSA. The chapter then gave the overview of the datasets used so far in the literature and their performance in SA domain. The next sections of this chapter emphasize the current research on feature extractions and preprocessing

techniques of SA for examining the impact of those techniques. Finally, this chapter tries to give a brief idea about NLP tools.

2.1 Natural Language Processing

This section gives a brief idea about NLP and illustrates how it is related to this research (especially MLSA). NLP is the study and research of computer science, linguistics, statistics, and artificial intelligence related to the acquaintances between computers and natural (human) languages (NL), intending to design and develop a computing system that can study, comprehend and synthesize NL (Ranjan et al., 2016; Manaris, 1998; Khurana et al., 2017).

Computers can easily and quickly process the harmonized and structured data as in database tables and the like than the human could. However, the human does not follow well-formed rules in communication. The humans communication languages are not either structured or in binary form and are highly unstructured, which the computers could not understand as there is a lack of standard tools and techniques to process them (Gupta, 2014; Khurana et al., 2017).

Thus, NLP in computing systems possesses some challenges, such as natural language understanding (NLU), which maps and analyzes the different aspects of the given natural language input into a valid interpretation. Moreover, natural language generation (NLG) creates expressive phrases and sentences in NL's structure from several internal interpretations (Khurana et al., 2017). These two are NLP components, and as per literature, NLU is difficult to process than NLG (Khurana et al., 2017).

NL is exceptionally ambiguous and has vibrant form and structure. There are many types of ambiguity in NL, such as lexical level, syntax level, and referential ambiguity. Lexical ambiguity is very primitive and causes at the word level. Syntax level ambiguity takes place in sentences, as the sentences may be parsed in diverse ways. Referential ambiguity takes place at the level of the pronoun (Ranjan et al., 2016).

Natural Language processing may be done at the following five levels[6](Samta et al., 2017; Gupta, 2014; Khurana et al., 2017), namely Lexical analysis is the identification and analysis (separation of whole text into words, sentences, and paragraphs) of the structure of words. Syntactic analysis or parsing is the study of grammar (having the relationships between words) in the words of the sentences for grammar and assembling words to show the relationship among the words. An English syntactic analyzer avoids the sentence, such as "The school goes to the boy." Semantic analysis extracts the actual meaning or the dictionary meaning from the text by matching syntactic structures in the domain of interest. The semantic analyzer disregards sentences such as "hot ice cream." Discourse integration helps in finding the relationship or meaning of the sentence using its pre and post sentences. Finally, pragmatic analysis helps in finding a different aspect of the languages using real-world knowledge.

The above levels are sometimes sub-divided to solve some more significant tasks. For example, *syntax analysis* is sub-divided into grammar induction, lemmatization, morphological segmentation, part-of-speech tagging, parsing, sentence boundary disambiguation, stemming, word segmentation, and terminology extraction.

[6]https://www.tutorialspoint.com/natural_language_processing/index.htm

Semantic analysis is sub-divided into lexical semantics, distributional semantics, machine translation, named entity recognition (NER), natural language generation, natural language understanding, optical character recognition (OCR), question answering, recognizing textual entailment, relationship extraction, sentiment analysis, topic segmentation and recognition, and word sense disambiguation. *Discourse integration* is sub-divided into automatic summarization, coreference resolution, and discourse analysis. *Speech analysis* is divided into speech recognition, speech segmentation, text-to-speech, and dialogue.

This thesis has processed the collected NL text mostly at the semantic analysis level from among these levels. However, as these levels are interconnected, so, it is difficult to process the text only at one level; therefore, this thesis also undertakes the process at some other levels like syntactic analysis or parsing level, if and when needed. A brief description of some sub-categories related to this thesis is presented below:

2.1.1 Syntax

2.1.1.1 Parts-of-Speech Tagging

It is the task of determining grammatical categories such as nouns, verbs, adjectives, etc., of each word of a given sentence. Many words have multiple parts-of-speech; prevalent ones such as "book" can be a noun ("the book is on the chair") or a verb ("to book a seat"). Some languages like English and Bengali contain more ambiguity than others languages (Gupta, 2014; Khurana et al., 2017; Ranjan et al., 2016).

2.1.1.2 Parsing

Parsing shapes the grammatical analysis of a given sentence through the parse tree as the natural language sentences typically have many meanings and are highly ambiguous. As per literature, two types of parsing is generally used such as dependency parsing- which highlights the associations between words in a sentence, whereas in constituency parsing, the parse tree is built using a probabilistic context-free grammar (PCFG) (Ranjan et al., 2003; Sha et al., 2003).

2.1.1.3 Sentence Breaking or Sentence Boundary Disambiguation

Sentence breaking is the task of finding the sentence boundaries from the given hunk of text and is usually done by following the periods or other punctuation marks (Walker et al., 2001).

2.1.1.4 Stemming

It is the task of cutting off the end of words to form their root or base words. (e.g. "closed", "closing", "closer" will be returning to its base or root word "close")[7]

2.1.1.5 Word Segmentation

Word segmentation is the task of dividing a given sentence to form a bag of words (BOW) with unigram[8], bigram8, trigram[8], opinionated words, etc., and is helpful in

[7] https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html

sentiment analysis. For both the languages, such as Bengali and English, the process of word segmentation is the same, as in both languages, sentences are formed by words separated by space (Gupta, 2014; Khurana et al., 2017; Ranjan et al., 2003).

2.1.1.6 Terminology Extraction

Terminology extraction is the task of automatically extracting significant terms from a given corpus (Alrehamy et al., 2017).

2.1.2 Semantics

2.1.2.1 Named Entity Recognition

It is also known as entity extraction, or entity identification, which identifies and classifies, named entities to the pre-specified categories from the given unstructured text. In the case of English, the NER is done mainly by using capitalization in consideration. In some languages, i.e., Bengali, which has no capitalization in words, the text's context is widely used for named entity identification (Ekbal et al., 2008; Gupta, 2014; Khurana et al., 2017).

[8]N-grams of texts in text mining are a set of co-occurring words/tokens within a given document. When n=1, this is called unigrams and which are the individual words in a sentence. When n=2, this is known as bigrams, and when n=3 this is referred to as trigrams. When n>3 is usually referred to as four grams or five grams and so on (Taher et al., 2018)

2.1.2.2 Word Sense Disambiguation

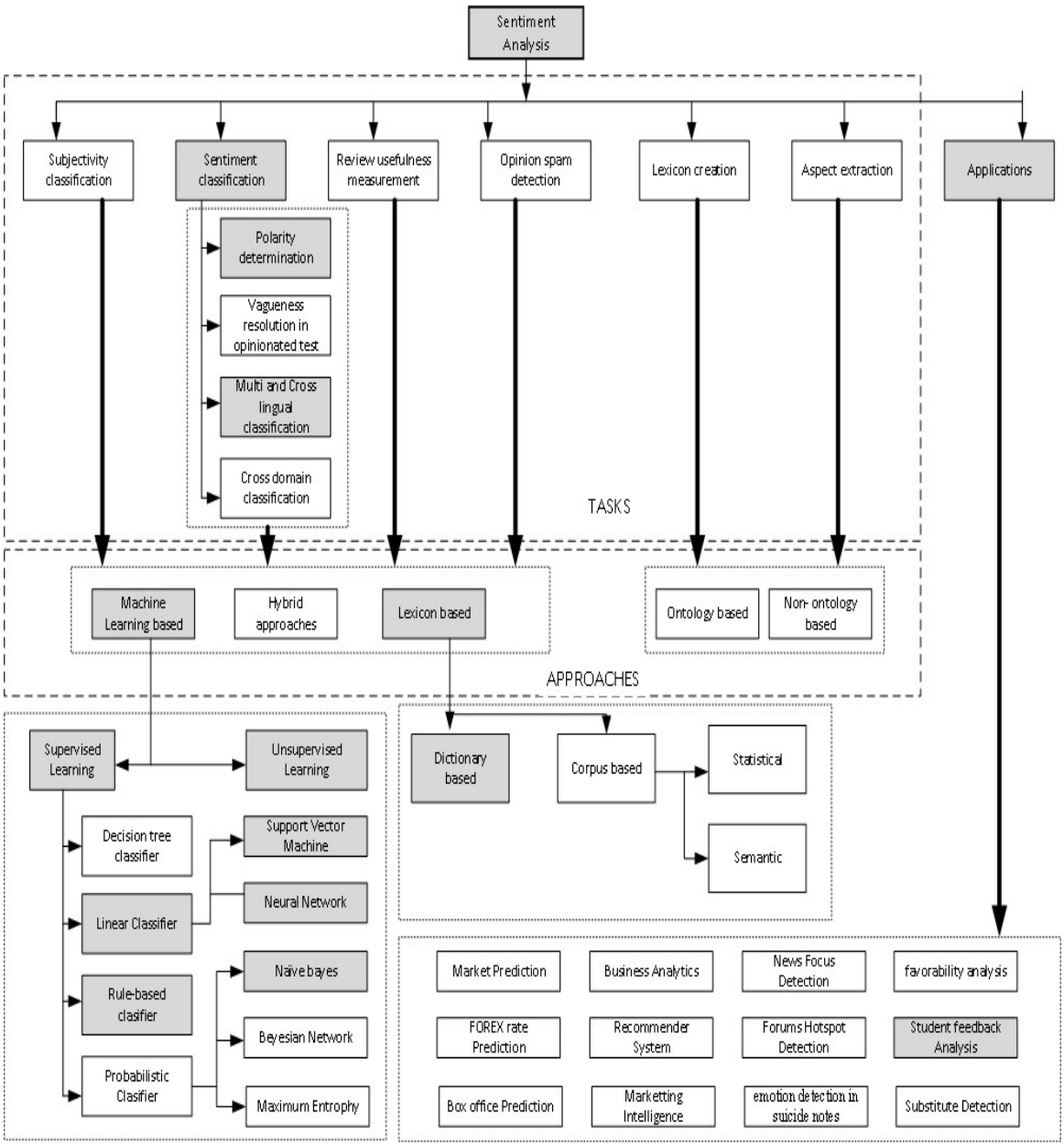
Word sense disambiguation is the task of selecting the meaning of the words using intensity to context due to the same word's many meanings. For this type of problem, enlisting words and related word senses, e.g., from a pre-defined dictionary or online sources such as SenticNet, are used[9] (Gupta, 2014; Khurana et al., 2017).

2.2 Sentiment Analysis Overview

This section emphasizes the basic concept of SA, the necessity of research on SA, and the different tasks related to sentiment analysis. This section also includes a detailed review of varying levels of SA, emphasizing its importance in this research.

SA is considered one of the preferable research fields for people of computer science in recent years. SA facilitates the organization with the necessary ability to monitor social networks such as Facebook, Twitter, blogs, message boards, and user forums in real-time (Feldman, 2013). Figure 2.1 shows SA tasks, approaches, and applications where the study focuses are highlighted. The figure gives the idea about the hierarchical position of this study. This research has performed sentiment classification on student feedback data to classify multilingual data and determine polarity using both machine learning-based (both supervised and unsupervised learning methods) and lexicon-based (dictionary-based method) approaches. NB, SVM, and LSTM were used among supervised methods, and among unsupervised methods, LSTM and novel algorithm (MCSAlgo) were used for sentiment classification.

[9] http://www.scholarpedia.org/article/Word_sense_disambiguation



Source: Ullah et al., 2017; Medhat et al., 2014

Figure 2.1: Sentiment Analysis Tasks, Approaches, and Applications

As per Liu (2011), opinions or sentiments consist of the following parameter: (1) Entities and their related features/aspects called opinion targets (2) Different types of sentiments such as positive, negative, or neutral (3) People holding the opinions, also

called opinion holders (4) the time of opinions expressed. However, in general, opinion is considered as

quintuple (*ej, ajk, soijkl, hi, tl*)

Here,*ej* represents a target entity;ajk represents aspect/feature of the entity *ej*. The sentimental value of the expressions or opinion (such as positive, negative or neutral, or any other granular ratings) from the opinion holder *hi* on feature *ajk* of entity *ej* at time *tl* is represented by *soijkl*; *hi* and *tl* represent the opinion holder and time of opinion expression respectively (Liu, 2011).

As per Appel (2015) and Kumar & Sebastian (2012), the critical task (opinion summarization (Liu, 2011)) performs in sentiment analysis is divided into three different categories based on the extraction of sentiments from a given corpus: (1) subjectivity classification (2) sentiment classification (3) complimentary functions such as opinion holder extraction object/feature extraction. This research has adopted two latter categories of opinion summarization.

Subjectivity classification: A document may contain both subjective and objective opinions. Therefore, as the document may have many sentences, some may be subjective, and some are objective (i.e., factual information) in nature. So, the job of classifying the sentences to opinionated and not opinionated from such a document is known as subjectivity classification (Kumar & Sebastian, 2012).

Sentiment classification: From the opinionated text, identification and classification of polarity as positive, negative, or neutral are known as sentiment classification. The polarity could also be expressed as excellent, very good, poor or

happy, moderately happy, sad etc. Sometimes these classifications are called multi-class classification (Appel, 2015).

Complimentary functions: (a) Opinion holder extraction: one of the most critical tasks in SA is an extraction of opinion holders, as a document may contain many opinion holders and could express an opinion on many different matters (Chiclana, 2015). (b) Object/feature extraction: if a document contains many potential entities, the task of identifying the effecting objects along with their related features from the document is known as object/feature extraction. For example, many issues are addressed in social media, such as product reviews, blogs, etc. (Carter, 2015). So it is crucial to extract the influential objects from those reviews, such as product car or camera, etc.

As per Feldman (2013), one of the most crucial resources for most SA algorithms is the sentiment lexicon acquisition. Three approaches are mainly helpful, such as (1) manual approaches- where the researchers should code lexicon by hand (2) dictionary-based approaches- where WordNet (Ellbaum, 1998) is used to expand a set of seed words (3) corpus-based approaches- from a single domain, a large corpus of documents is used to develop seed words.

Among the three approaches, the latter two approaches are feasible, and a manual approach is prohibited, as it requires laborious effort (Ronen, 2013). Therefore, this research has adopted a dictionary-based approach.

2.2.1 Levels of Sentiment Analysis

Research shows that there are many levels of analysis related to the sentiments mentioned above (Kumar & Sebastian, 2012; Liu, 2012; Feldman, 2013; Cambria et al., 2013), such as:

2.2.1.1 Document-Level Sentiment Analysis

One of the simplest forms of SA is document-level sentiment analysis. The main concept of document-level sentiment analysis is to derive the emotions on one primary entity from the whole document, that being expressed by the originator of the document. Document-level sentiment analysis has been researched using different learning approaches such as supervised, unsupervised, and semi-supervised (a learning approach containing both labeled and unlabeled data) and applied on online reviews and news blogs (Sharma & Dey, 2012). Turney (2002) proposed an approach to unsupervised learning in a seminal work, where the first task is to extract adjectives and adverbs, and then semantic orientations (SO) were calculated on extracted phrases with the use of point-wise mutual information (PMI). Because of English resources' shortness, some researchers use NLP to do the document-level analysis in Spanish and Chinese (Wan, 2008; Brooke, 2009).

2.2.1.2 Sentence-Level Sentiment Analysis

Another critical form of SA is being done on sentence-level. Generally, a document contains multiple subjective and objective sentences being expressed about the

same entity, and sentence-level focuses mainly on subjective sentences with some exceptions, as objective sentences are difficult to interpret (Appel, 2015; Feldman, 2013). In classifying sentences, a supervised approach is mainly used, and in rare cases, the unsupervised approach is used (Yu, 2003). Research shows that (Narayanan, 2009), because of the sentences' diversity (such as sarcastic sentences, question sentences, conditional sentences), different techniques, approaches, and strategies should be used to handle them.

Hai (2011) suggested a bootstrapping approach to decrease the manual work burden in preparing a vast training corpus. Neviarouskaya (2010) has developed a system using Izard's (1971) affect categories and Martin and White's (2005) appraisal theory, where the word in the sentence is considered for sentiment computation. Nakagawa (2010) developed a model like a dependency parse tree model, known as a conditional random field model for sentences. This model uses opinionated words and polarity shifters to classify the polarity of sentences. Depending on the corpus they have tested, they were said to receive 77% to 86% accuracy at categorizing sentences.

2.2.1.3 Comparative Sentiment Analysis

Considering the following sentences, "The display of the iPhone is better than HTC," or "Green Tea helps reduce cholesterol more than Lemon juice," in both the sentences, users do not provide a direct opinion about any specific product; instead, it contains multiple comparable opinions. Thus, comparative sentiment analysis's main target is to detect the factors inside the desired entity from each sentence (Ronen, 2013).

In today's world, this analysis has also become a more crucial issue in this domain of interest and has become a significant research choice. Work done on comparative sentiment analysis by Jindal and Liu (2006) suggested that if the researcher uses a small number of words (such as comparative adjectives, adverbs, i.e. 'less' or 'more,' superlative adjectives and adverbs, i.e. 'least' or 'most,' additional phrases, i.e., 'exceed,' 'prefer' or, 'than'), they could cover 98% of comparative opinions.

2.2.1.4 Aspect or Feature-Based Sentiment Analysis

Document-level and sentence-level sentiment analysis are two crucial research pieces in sentiment analysis literature, especially when working with a single entity or object. However, if the document contains multiple features or attributes referring to the same entity or entity set and people express their views on each of the attributes (such as weight, size, color, etc.) of the entities (such as laptops, mobile phones, cars, cameras, etc.). Therefore, it becomes necessary to recognize all the expressions they have given to the documents by categorizing the document's implicit aspects. Fine-grained analysis for doing so is called aspect-level sentiment analysis (also known as feature-based sentiment analysis) (Ronen, 2013).

As per Hajmohammadi (2012), aspect-based sentiment analysis could be done in two significant aspects: aspect extraction and aspect sentiment orientation detection. As extracting the features (aspects) automatically from the opinionated documents is a complicated job, there is in need to use natural language processing techniques. Some of such techniques are association mining and web PMI (Etzioni, 2005), likelihood ratio

(Niblack, 2003), CRF approach (Gurevych, 2010), SVM (Nicolov, 2009), Hidden Markov Model (HMMs) (Srihari, 2009), naïve Bayes (Mubarok et al., 2017), neural network (Wang et al., 2018).

One of the earliest works was done by Hu and Liu (2004) to find the frequently used features in product reviews. They have used pruning strategies and association rule mining techniques with the assumption that the product features are nouns or noun phrases. Yi et al. (2003) developed a complete system for aspect extraction (an unsupervised aspect extraction technique) along with likelihood ratio and mixture language model-based feature term selection algorithms for testing.

An approach known as "OpinionMiner" was developed by Jin et al. (2009)to find target features and emotions expressed. They have built the approach based on the framework of lexicalized hidden Markov model (LHMMs) -a machine learning approach. They merge various significant linguistic features such as part of speech, contextual clues, phrases, and internal information patterns in an automatic learning method. Designing and using a bootstrapping approach (through self-learning) could extract high confidence labeled data), the training data is labeled. Kessler and Nicolov (2009) found that individual relatedness between opinion and aspect in the product review sentences, i.e., feature vectors, was formed based on the syntactic and semantic relationship between the emotions a particular aspect.

On the other hand, as per Hajmohammadi (2012), the second task in sentiment analysis at an aspect level is to detect the sentiment orientation expressed (i.e., positive, negative, or neutral) on each feature in the sentence of the review by dividing the total job into several sub-jobs. The sub-jobs are (1) Opinion words or phrase extraction, (2)

Identification of polarity for every opinion words or phrases, (3) Management of opinion intensifiers (e.g., very, educational) and opinion shifters (e.g., no, not, do not), (4) Control of 'but' clauses, (5) Sum up the opinions, where the sentence contains more than one emotion (opinion) word or phrase.

Hu and Liu (2004) used a distance-based approach to extract emotion or opinion words (in this research, the authors used adjacent adjective words) and phrases. Without obtaining intensifiers, a WordNet lexicon was used to calculate the polarity of every emotion extracted. The authors also considered the 'but' clause in their research as it implies opinioned feature changes in the clause and resolves the problem by using strong opinions to select the orientation of the aspects in the sentence. Etzioni (2005) was using the same idea except for syntactic dependency rule templates. Instead of a distance-based approach to recognized object features to possible opinion phrases and words, semantic orientations of possible opinion words or phrases were identified by relaxation. Relaxation is the labeling technique from the entity features and sentences of reviews by ignoring the intensifiers.

Godbole et al. (2007) done the same task as Hu and Liu (2004) with an assumption that all the recognized sentiment words in the same sentence can be entitled to the same object or entity. A unique method proposed by Qiu and Liu (2011) that could extract both opinions and features simultaneously is known as a propagation-based method. Generally, the known fact is that aspects or features could be derived from opinion words. The author used this fact in their research and assumed that the natural relation between opinion phrases or words and features exists. They have used a bootstrapping approach, the work process starts with a word known as a seed, and then

they have used many syntactic relations to link opinion words and features, thus finding new features. The process was repeated until they concluded that no more opinion words or new features could be recognized.

2.2.1.5 Concept-Level Sentiment Analysis

This section emphasized the basic idea of concept-level sentiment analysis, several existing influential knowledge bases and their uses in many potential research works, different special tasks related to concept-level analysis, and their use in various work done by researchers. Finally, a comparative study discusses the differences between the concept-based approach and other approaches.

With the advent of social networking today, a few dozen exabytes of information are created weekly. This vast amount of data is mainly unstructured and not directly machine-readable (Cambria, 2013). Concept-level sentiment analysis is a step towards such an approach based on the semantic review of text using a semantic network or web ontology. Besides, it allows for combining important and conceptual information that is related to NLP opinions. One of the main components of a concept-based approach is its knowledge base. Additionally, its reliability and validity are solely dependent on its components. Without the knowledge bases enriched with human knowledge, it is impossible to adjust with NLP text semantics (Cambria, 2013).

Recent approaches to sentiment analysis at the concept level are highly dependent on some of the most influential knowledge bases (such as Senti- WordNet, ISEAR, ANEW, SenticNet, and WordNet-Affect, etc.) of today (Cambria, 2013). Hung et al.

(2013) proposed a system to re-assess the objective words in SentiWordNet by reviewing the sentimental significance of the objective words and their related sentiment sentences. For sentiment classification with SVM, two case strategies are planned and integrated. From the practical tests, an approach that considers words outperform the traditional opinion mining methods.

Bosco et al. (2013) proposed a system; the main task was to develop the corpus for sentiment analysis using different methods (such as surveying the existing work and presenting a case study) in this field of interest. It was an Italian project called Senti-TUT. The reasons for these corpora's development were to find the irony surrounding politics using the social network as sources. Tsai et al. (2013) built a dictionary of the concept-level sentiment using a two-step method, namely a random walk with in-link normalization and iterative regression. SenticNet and ANEW were divided for spreading sentiment values, with the assumption, semantically related concepts share common sentiments. Instead of using mean error to assess sentiment dictionaries, those projects used average-maximum ratio, polarity accuracy, and Kendall distance. Poria and others (Poria et al., 2014; Musto et al., 2014; Poria et al., 2018; Satapathy et al., 2019) followed the same method. They have offered an approach that assigned a sentiment label using emotional affinity and point-wise mutual information (PMI) to each SenticNet concept to enrich the concepts' affective information.

Few works emphasize applying statistical methods and knowledge bases proposed by Weichselbraun (2013) to adjust with ambiguity and aggregate the aspects of sentimental items, known as a hybrid approach, where two different approaches (lexical analysis and machine learning) are mingled. These approaches identify the elements with

ambiguity using the polarity of the items. Besides, it saves the elements as contextualized sentiment lexicons, which with semantic knowledge bases help eliminate ambiguity in concepts according to polarity. Concept-level sentiment analysis includes different types of applications such as opinion summarization (Fabbrizio, 2013), multimodal sentiment analysis (Rosas, 2013; Wollmer, 2013), and domain adaptation (Xia et al., 2013).

Xia et al. (2013) proposed a sample selection and feature ensemble (SS-FE) approach, a comprehensive approach, where the feature ensemble model uses feature re-weighting to make learning a new labeling function. Furthermore, another method known as PCA-based sample selection was used to support the feature ensemble model. Fabbrizio (2013) proposed a novel approach towards concept-level summarization. This approach considers some features for summary (such as language modeling and aspects rating distributions), which helps to extract multi-document summarization from the determinant text. This approach is more productive and improved in dealing with the multi-document summary from the determinant text than the other approaches.

2.2.1.5.1 Impact of Concept-Based Approach over Other Approaches

Several approaches exist to deal with sentiments and provide certain information. As per Cambria (2013), sentiment analysis could be conducted using analytical tokens and the implicit information related to those tokens. The approaches used for sentiment analysis are grouped into four distinct types: statistical methods, keyword spotting, lexical affinity, and concept-based techniques. However, most of these approaches pose different problem issues (Problems with keyword spotting are poor identification of affect with negation, the need for obvious affect words. Problems with lexical affinity are working

exclusively on the word level, and probabilities are subject to specific genre text. Statistical methods are semantically weak and not working well for smaller text units such as sentences or clauses (Cambria, 2013)). Also, among all of these approaches, the concept-based approach shows relatively few problems and excellent output (Cambria et al., 2014; Bajpai et al., 2016; Cambria et al., 2016; Bisio et al., 2017; Cambria et al., 2018; Chen et al., 2019). Therefore, the following section emphasizes the difficulty of all these approaches and the relative advantages of a concept-based approach.

With SVM and Bayesian inference, a statistical approach is very famous for affect text classification. However, this method has some problems. Statistical text classifiers work well on the page or paragraph level rather than a sentence or clause level (Havasi et al., 2013). On the other hand, keyword spotting works well on the user's text that consists of explicit affect words (i.e., bored, afraid, happy, and sad). However, this approach also presents some problems; for instance, it cannot identify negated affect words. It could correctly classify a sentence "The display of iPhone is good" as a positive, but "overall condition of this iPhone is not good" is incorrectly classified as positive, though it reflects negative sentiments (Havasi et al., 2013). Moreover, lexical affinity outperforms the previous approach; however, it possesses two problems; one with negated sentences and others with sentences that contain different meanings. This problem occurs as this approach only works on a word level (Cambria et al., 2013).

The concept-based approach proved helpful in adorning the relationship between features and products by using implicit meaning related to natural language concepts rather than using the word co-occurrence counts and keywords as used by other approaches (Ravi et al., 2015). Their research shows that the concept-based approach

shows more classification accuracy than any other approach (Martinez et al., 2014). Lau et al. (2014) developed an ontology-based learning system to support sentiment analysis at the contextual aspect level, where the authors used two sampling methods such as Gibbs and LDA (Gibbs sampling is a method in which Markov chain is built to sample from a required joint (conditional) distribution. Gibbs sampling can be used in language processing problems such as approximate inference in Latent Dirichlet Allocation (LDA). LDA is the generative model for determining the topics from the collection of text documents (Wei et al., 2006)) to extract explicit and implicit features. They articulated that this approach improved sentiment classification accuracy by 11.6% than the non-ontology-based approaches.

Liu et al. (2004) created ontology to derive the overall review polarity using a bottom-up approach on Pang and Lee's dataset (Pang & Lee, 2002). Their approach was said to achieve 7.55% more accuracy than SVM and outperformed several other approaches (11 approaches). Peñalver-Martinez et al. did another work using the same data set as in previous work (Liu et al., 2004) and extracted aspects from movie reviews using the domain ontology viz. They have calculated sentiment scores using SentiWordNet. The authors were said to have found an accuracy of 89.6% in classification, which was much better than the non-ontology-based approach (Martinez et al. 2014).

The literature discussed above indicates that document-level and sentence-level analysis of sentiments seem to underperform compared to feature and concept-level analysis, so this thesis has worked on feature and concept-level sentiment analysis from the collected corpus of multilingual nature, keeping future opportunities into

consideration. Moreover, the research considered the proposal of Hajmohammadi (2012) and Cambria et al. (2013) in handling with corpus and finding the potential features and concepts from the corpus. Table 2.1 shows different multilingual approaches used to date. The table shows that the concept-based approach is applied in English, China, and 39 other languages; however, it is not applied to the Bengali language. The approaches applied so far for the Bengali language are corpus and lexicon-based and are mostly domain-specific, such as some on newspaper and some on online shop data etc. However, the approaches are not applied Bengali student feedback dataset.

Table 2.1: Multilingual Approaches

References	Approach	Language	Challenges
Mihalcea et al., 2007	Bilingual dictionary translation along with rule-based classifier	English, Romanian	Uncertainty of word sense
Denecke, 2008	Corpus-based approach and translation by LingPipe classifier	English, German	Negative text recognizing the problem
Boiy & Moens, 2009	Aspect focused corpus-based approach	English, Dutch, French	The major problem(lack of training examples) with informal languages used in blogs
Boyd-Graber & Resnik, 2010	Corpus-based MS-LDA	English, German, Chinese	In need of diverse resources as a bridge to associate the different corpus
Xia et al., 2014	Concept-based along with translation	English, Chinese	Translation problem, untranslated words, and out-of-vocabulary concepts
Balahur et al., 2014	Corpus-based together with translation and machine learning	English, Spanish, French, German	In need of translators for the target language
Sharma et al., 2016	Dictionary-based, Lexicon-based	Hindi-English	Scarcity of resources
Lo et al., 2016	Singlish sentic patterns	English-Malay-Chinese-dialects	Multiple meaning of same words for different languages
Kaur et al., 2017	Dictionary-based	Hindi-English	Informal languages or slangs

References	Approach	Language	Challenges
Wang et al., 2017	Factor graph model + belief propagation	Chinese-English	The same word having different sentiment
Pravalika et al., 2017	Lexicon Based + Machine Learning	Hindi-English	Informal languages or slangs
Cambria et al., 2018	Concept-based	English and 40 other language	Need for language-specific concepts
Rahman et al., 2019	Corpus-based	Bangla, English	Completeness in vocabulary
Cambria et al., 2020	Concept-based	English and 40 other language	Need for language-specific concepts
Ali et al. 2020	Lexicon Based	Bangla, English	To cope with other domains
Milu et al., 2020	Corpus-based	Bangla, English	To cope with other domains

2.3 Bengali Language

The Bengali Language is also known as the Bangla language, and it belongs to the "Indo-European" language family. This language is historically connected to Portuguese, English, French, Dutch, and other Indian languages[10]. Bangla is the language of Bangladesh, and with 230 million native speakers, it is the world's seventh-ranked language. The evolution history of Bangali could be divided into three phases such as Old (900/1000-1350), Medieval (1350-1800), and Modern (1800-)[11].

[10] http://en.banglapedia.org/index.php/Bangla_Language

[11] http://en.banglapedia.org/index.php/Bangla_Language

2.3.1 Division of Bangla Vocabulary

Bangla vocabulary is divided into the following strata; Figure 2.2 illustrates the subdivision of the vocabulary (Kar et al., 2019):

1. Tadbhaba

Tadbhaba is a native Bangla word collected from Sanskrit and Prakrit.

Example: রস (ras) - ‘juice’, ফুল (phul) - ‘flower’ etc.

2. Tatsama

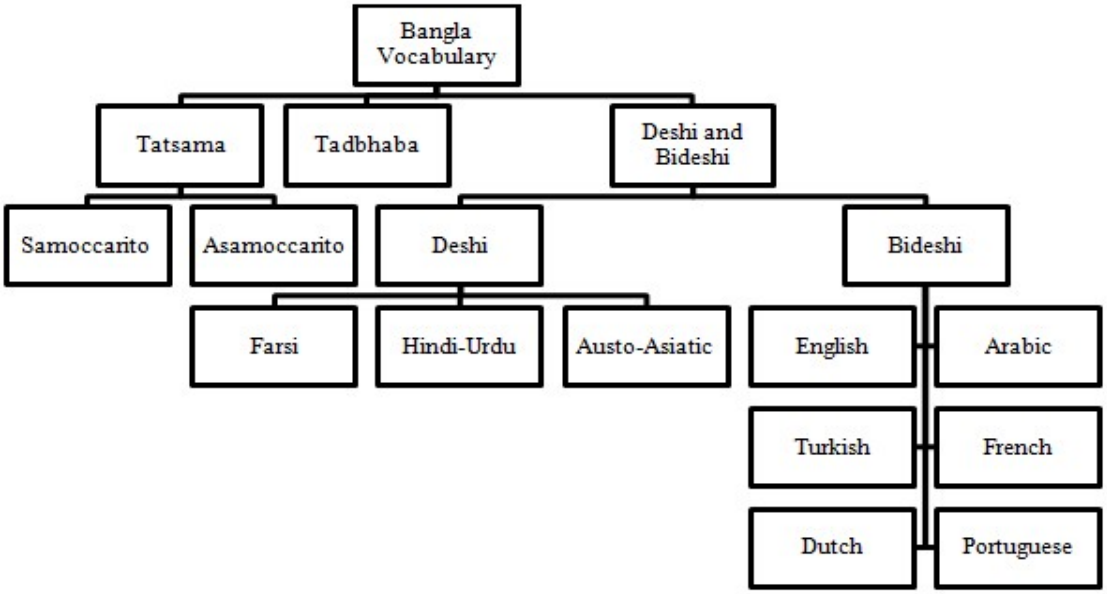
Tatsama are the words directly collected from only Sanskrit.

Example: grām (gram) - 'village’etc.

3. Deshi and Bideshi

Deshi and Bideshi words are borrowed from different Indian and foreign languages.

E.g., আনারস (ānāras) - ‘pineapple’ (from Portuguese), হরতাল (haratāl) - ‘strike’ (from Gujarat), etc.



Source: Kar et al., 2019

Figure 2.2: Division of Bangla vocabulary

Basic Sentence Structure of Bengali and English Language (Kar et al., 2019)

1. The basic sentence pattern

English sentence pattern: subject (S) + Verb (V) + Object (O) (SVO)

Bengali sentence pattern: subject + object + verb (SOV)

Example:

English: I (S) like (V) university (O).

Bengali: আমি (S) বিশ্ববিদ্যালয় (O) পছন্দ করি (V).

2. Auxiliary verb

There is no auxiliary verb in the Bengali language.

Example:

English: I (Pronoun) am (Auxiliary verb) doing (Main verb) homework (Noun).

Bengali: আমি (Pronoun) বাড়ির কাজ (Noun) করছি (Main verb).

3. Preposition

In place of prepositions, in the Bengali language, *bibhakti* is used. It is placed after noun or pronoun

Example:

English: The student sat on the table.

Bengali: ছাত্রটি টেবিলটিতে বসল. Here 'তে (te)' is *bibhakti*

2.3.2 Bengali Language Problems Related to Sentiment Analysis

Dialects- the Bengali language has many dialects, such as chittagonean, Sylheti etc., and are widely used in social media. However, due to the lack of uniformity in such dialects, it is not easy to deal with those dialects (Sazzed et al., 2019).

Translation issue- In the research, enormous challenges need to be tackled in translating the data gathered in different languages, for instance, English and Bengali. The processing of English data is easier than Bengali data. Therefore, Google Translators are used; however, it is sometimes not effective (Magueresse et al., 2020).

Annotation Problem- annotated data are challenging to attain in low-resource languages (i.e., Bengali). The Bengali language faces the lack of a publicly available lexicon-based tool for sentiment analysis. However, a resource-rich language like English possesses a large volume of annotated data. Therefore, the Bengali language leverages sentiment lexicon and annotated data from English for sentiment analysis (Sazzed, 2020).

Dealing with Bengali English - most of the expressions in social media are written in Bengali English; that means Bengali words are written in English. For example, ‘আমি’

is written as ‘ami.’ Dealing with this form of Bengali English is challenging due to the unavailability of appropriate lexicons or dictionaries (Ullah et al., 2020; Sazzed, 2020).

Lack of Datasets- an obstacle to Bengali sentiment analysis research is the lack of appropriate datasets. Most of the available datasets are smaller and need more data to be added (Ullah et al., 2020; Sazzed, 2020).

Exploration need of cross-lingual approach - the cross-lingual approach needs in-depth investigation as it is not explored largely (Sarkar et al., 2020; Sazzed, 2020).

2.3.3 Bengali Sentiment Analysis

The popularity of E-commerce sites, social media, and microblogging have had emerged the opportunity for Bengali sentiment analysis. However, very few researches have been conducted so far due to the different problems mentioned above. Das et al. (2010) has developed a Bengali sentiment dictionary by translating the English polarity lexicon. Sarkar et al. (2015) has performed a sentiment analysis on SAIL 2015 dataset using multinomial Naïve Bayes (MNB) and SVM. A similar dataset was utilized by Seshadri et al. (2016) on three languages, such as Bengali, Hindi, and Tamil, using RNN to find the sentiment in three polarities (positive, negative, and neutral) and found 65.16% accuracy for the Bengali Language. Prasad et al. (2016) proposed a model to find Bengali and Tamil tweets’ sentiment using C4.5, Naïve Bayes, and Decision tree. The proposed model used unigram, bigram, and features from Wordnets.

Paul et al. (2016) have applied Mutual Information for feature selection and MNB for classifying sentiments from Bengali and English reviews. The author achieved good

performance for the Bengali dataset (a translated version of Amazon's watch English dataset) compared to the English dataset. Supervised learning such as naïve Bayes is used by Islam et al. (2016) to determine the sentiment of user comments on Facebook. They have tested the algorithm on a small dataset and keep the door open for future research. Islam et al. (2017) classified the Bengali document with different features such as normalized TF-IDF and chi-square distribution using SVM, NB, Stochastic Gradient Descent, and achieved a better result with chi-square + NB combination.

Asimuzzaman et al. (2017) has conducted fuzzy sentiment analysis on Bengali text. They have applied Adaptive Neuro-Fuzzy Inference System to find the polarity of Bengali tweets and achieved satisfactory results. Banik et al. (2018) have used ML algorithms such as NB and SVM to find Bengali movie reviews' polarity and got an 86% precision value. Taher et al. (2018) used linear and non-linear SVM with n-gram features on web-based Bengali diverse data to find sentiment. Due to the unavailability of appropriate datasets in Bengali, Mahtab et al. (2018) created a Bengali cricket dataset and labeled it with three sentiment classes. The authors have applied vectorization using TF-IDF vectorizer, classified the data using SVM, and achieved 64.5% accuracy. In a paper, Haydar et al. (2018) has created a dataset on e-commerce and restaurant review on Facebook pages. They have evaluated sentiment at word level using RNN and achieved an overall accuracy of 78%.

Sazzed et al. (2019) used two datasets, n-gram (i.e., unigram and bigram) features and ML algorithms in Bengali and English corpus, and found satisfactory SA results on both lingual data. The cross-lingual method for Bengali SA requires more investigations (Das et al., 2010a; Sazzed et al., 2019; Sazzed, 2020). Sazzed (2020) has explored

several labeled and unlabeled data and created a Benchmark dataset for investigating the usability of the cross-lingual method in Bengali sentiment analysis. The author then examined different classification algorithms, transfer learning and provides the way for future research.

Bengali feature words list was developed and done the sentiment analysis at document or sentence level using Random Forest algorithm by Tabassum at al. (2019). The authors improved the performance (accuracy 67%) by using unigram, handling negation, and POS tagging. Rahman et al. (2019) compared the SA result of different ML classifiers (NB, DT, KNN, SVM, and K-means) on annotated Bengali Text (socio-economic and political) collected from different Facebook groups and achieved an overall accuracy of 52.98%. Sarkar et al. (2019) have applied CNN and deep belief networks on SAIL 2015 dataset for detecting sentiment polarity. They have achieved 46% accuracy and claimed that the result is better than baseline evaluation.

Haque et al. (2020) have claimed that due to the improper annotated dataset, polarity lexicons, and text corpora, SA research has been hampered. They have also claimed to improve the SA result of two famous datasets such as Cricket and Restaurant, using different supervised ML algorithms (RF, SVM, and KNN). A lexicon-based dataset (*BanglaSenti*) was generated by Ali et al. (2020) to identify sentiment from text data. A dataset of 61582 positive, negative, and neutral Bengali words was formulated using the English SentiWordNet dataset for easy regeneration.

Ahmed et al. (2021) have created a Bengali dataset and shown the hyperparameter tuning effects on the dataset. The author determined the sentiment using the LSTM model. The model achieved an accuracy of 94%. Sharmin et al. (2021) has presented an

attention-based CNN model for Bengali SA. They have used the sparsity of the data matrix. Their proposed model has a convolutional layer that has been used in feature extraction. They have achieved an overall accuracy of 72.06%. A research work (Bhowmik et al., 2021) has developed a Bengali SA data dictionary using different preprocessing techniques like tokenizing, normalizing, and stemming. The same research has developed a rule-based algorithm for sentiment score (BTSC) calculation that could calculate the sentiment using TF-IDF and POS tagging. Besides, the performance is evaluated using SVM beside BTSC, where the performance of BTSC was found better with 82.21% accuracy.

2.4 Multilingual Sentiment Analysis

This research aims to provide multilingual sentiment analysis (MLSA) from students' feedback in social media; therefore, the section gives the basic idea about the MLSA and some relevant work descriptions.

Most of the sentiment analysis is currently done in a single language, mainly in English. However, as the internet grows, the diversity of communication in web content also prevails. Moreover, analyzing sentiments using one language creates the chance to miss helpful information written in other languages (Lo et al., 2016). Therefore, MLSA tools and techniques are developed (Boiy et al., 2009). The process of analyzing a multilingual corpus with these tools to find useful information is known as multilingual sentiment analysis.

One of the critical problems with MLSA is an insufficient amount of resources (Balahur, 2014). Therefore, sentiment analysis in multiple languages requires transferring the knowledge from and resource-rich languages because of resource unavailability in other languages (Denecke, 2008). Most MLSA systems use English lexical resources such as SentiWordNet (Dashtipour et al., 2016); this approach is known as a machine translation system (Denecke, 2008). The translation system poses some problems, such as sparseness and noise in data (Balahur, 2014).

Another problem with this system is missing the translation of an essential part of the text and dividing the well-structured sentences into fragments (Bautin, 2008). Therefore, the alternative solution for MLSA was developed, which uses multilingual lexical resources (Dashtipour et al., 2016). One such resource is the NTCIR corpus, which contains the data on sentiment subjectivity and opinion holder in English, Japanese and Chinese. These corpora are related news of politics and sports (Seki, 2008). These resources could only be used for the language for which it was developed.

SA with multilingual data poses many challenges, such as statistical approaches need training material and are generally rare for different languages or sometimes unavailable. Moreover, lexical approaches demand language restricted lexical as well as linguistic resources (Denecke, 2008). Both approaches are time-consuming and often in need of manual work. There are some approaches to deal with the problem mentioned above, such as the corpus-based approach (create the subjectivity-annotated corpus of the intended or target language), a lexicon-based approach (translate an existing lexicon to generate subjectivity classifier) (Mihalcea et al., 2007), and local grammar approach (sentiment holding phrases and words are extracted) (Ahmad et al., 2006).

Subjectivity, as well as polarity detection, is two main approaches to sentiment analysis. Subjectivity detection is on realizing if the subject matter contains personal observations and views instead of factual information (Lo et al., 2016). In contrast, polarity detection is about learning subjectivity with diverse intensities or polarities such as highly positive or positive, highly negative or negative (Pang et al., 2008), or human emotion, for example, joy or anger (Cambria et al., 2014).

Subjectivity and polarity analysis are not limited to English content anymore due to social media's popularity worldwide. In actuality, 28.6 % of the web users speak English. It is, therefore, essential to discover or make resources and tools in languages other than English, such as Chinese, Japanese, and Bengali. As multilingual subjectivity, as well as polarity study, has become popular, some studies were done in Chinese (Hu et al. 2005; Tan et al., 2008; Zhao et al. 2012), Japanese (Kobayashi et al. 2005), Arabic (Mageed et al. 2011), German, Italian, Spanish, French (Balahur et al., 2013), Swedish (Rosell et al., 2010) and Romanian (Mihalcea et al. 2007).

As aforementioned, some sentiment analysis studies use multilingual twitter data for MLSA rather than only English. An approach was proposed by Volkova et al. (2013) that bootstrap subjectivity clues from Tweets and assesses their method on English, Russian, and Spanish Twitter streams. They have employed the MPQA lexicon (Wilson et al., 2005b) on the proposed approach for bootstrapping sentiment lexicons from a large volume of unlabeled data by applying a small sum of label data to conduct the process. However, their proposed approach faces some challenges in classifying subjective tweets with rational thoughts. Moreover, words with ambiguous sense with differing polarity are established to be especially error-prone (Lo et al., 2016).

By using tweets of SemEval 2013 Task 2 (Nakov et al., 2013) as training and testing datasets, Balahur and Turchi (2013) construct a secure sentiment analysis system in English. The data sets were then translated from English to other languages such as German, Italian, Spanish, and French. It is shown that combined training datasets from the same structures' languages help achieve improved results over individual language. The approach is not proven to solve the error generated by translation, though it is considered worthy.

Instead of using a translation machine, Cui et al. (2011) concentrate on constructing SentiLexicon by using emoticons, punctuations, and letter repetitions. They have first extracted emotion tokens to build the co-occurrence graph. Then, they have labeled positive and negative lexicons using the graph propagation algorithm. Their relative assessment with SentiWordNet (Baccianella et al., 2010) specifies that emotion tokens are useful for Twitter sentiment analysis in English and non-English.

Sazzed et al. (2020) evaluated ML algorithms' performance on the machine-translated corpus (Bengali- English) and compared the result with the original corpus. The evaluation outperforms the performance of the lexicon and transfer learning-based approach. Lwin et al. (2020) find students' sentiments using naïve Bayes classifier on English or Burmese data. The data were too noisy as it contains a mix of both lingual data. The authors have translated it to Unicode and claimed to achieve a better result.

It could be concluded from above discussion that, MLSA research is seldomly done. The research are mostly in English. The low resource language are less explored especially Bengali. Therefore, there are chances of exploring Bengali language by developing or finding optimal resources and techniques.

2.5 Lexicons and Knowledge Bases

One of this research aims to create a Bengali knowledge base and Bengali polarity lexicon that may resemble or relate to English or Bengali knowledge bases and polarity lexicon to help expand research in the future. Therefore, related literature was studied to find the gap in the existing resources and application of those resources. This section provides a detailed overview of related works on lexicons and knowledge bases, especially SenticNet. This section also highlights the reason for creating BanglaSenticNet (a Bengali knowledge base) in this thesis.

Sentiment analysis techniques are divided into two AI approaches such as symbolic (it programs the polarity related to words or multi-word expressions with the adaptation of lexicons (Bandhakavi et al., 2017), semantic networks (Poria et al., 2012), and ontologies (Dragoni, Poria & Cambria, 2018)). Furthermore, sub-symbolic (the sentiment classification is done by the adaptation of different machine-learning techniques such as supervised (Oneto et al., 2016), semi-supervised (Hussain & Cambria, 2018), and unsupervised (Li et al., 2017)). Cambria et al. (2020) have ensembled the symbolic and sub-symbolic AI tools to identify polarity from the text. They have applied a new version of SenticNet, known as SenticNet 6 (a commonsense-based knowledge resource).

Mohammad et al. (2013) developed the subjective lexica from Twitter data using the method known as point-wise mutual information. Later, these lexica were used to extract the features, train, and test the supervised classification model. More recent methods rely on deep neural networks and generative adversarial networks (Young et al., 2018; Li et al., 2018). Severyn and Moschitti (2015) obtained the best performance for

SemEval 2015 (Rosenthal et al., 2015) using the CNN with word embeddings and distant supervision. Vilares et al. (2015) presented a syntactic version in their approach. They then tested the use of dependency parsing in classification problems and obtained improved classification results.

There is a good deal of research work done recently on sentiment analysis in a wide range of languages, such as Arabic (Ibrahim & Salim, 2013), Chinese (Peng et al., 2018), French (Ghorbel & Jacot, 2011), German (Scholz & Conrad, 2013), Hindi (Medagoda et al., 2013), Italian (Neri et al., 2012). Besides, Japanese (Arakawa et al., 2014), Russian (Medagoda et al., 2013), Spanish (Vilares et al., 2015), and Thai (Inrak &Sinthupinyo, 2010). However, due to the lack of proper sentiment dictionaries in languages other than English, those researchers face huge problems when working (finding sentiments) with the text from those languages.

A lexical resource such as SentiStrength (Thelwall et al., 2012) was translated automatically to address a variety of languages, for example, Spanish. This resource positively contributed to the improvement of the performance of sentiment analysis. Thelwall et al. (2012) used SentiStrength for dual-score sentiment analysis of English short-texts, considering the characteristics such as grammatical quality of the text, repetition of characters, and excessive use of capital letters. Hogenboom et al. (2014) found the sentiment scores in the Dutch version of the SentiWordNet through the interrelationship between English WordNet (Miller, 1995) and its Dutch correspondent (Vossen, 1998).

Ghorbel and Jacot (2011) translated English SentiWordNet items into French. They have shown, even a correct translation does not guarantee that the words from both

languages will provide the same semantic orientation in similar usage. Volkova et al. (2013) solved this problem by adopting crowdsourcing (where sentiment lexicons were learned for different languages such as English, Spanish and Russian texts from Twitter). In Spanish and Russian languages, bilingual dictionaries were used to translate seed words from English. An approach for creating sentiment lexicons to support 136 languages was proposed by Chen and Skiena (2014). They have integrated different linguistics sources for creating a knowledge graph. They have covered 45.2% of current lexicons in their experiment.

However, the resources for multilingual sentiment analysis are still too rare. Xia et al. (2014) have attempted to propose a method for creating a version of SenticNet in the Chinese language using web dictionaries. They have adopted the English equivalent concepts in Chinese and its corresponding set of semantics. They have recommended that this resource could be used for rare resource non-English languages.

Some of the knowledge resources in English and other languages are SenticNet 5 (Cambria et al., 2018), OntoSenticNet (Dragoni et al., 2018), BabelSenticNet (Vilares et al. 2018), SenticNet 6 (Cambria et al., 2020). Along these, some Bengali corpora are developed. However, the literature does not show the existence of a Bengali knowledge base. Bengali corpora are “BanglaSenti” with 61582 Bengali positive, negative, and neutral words (Ali et al., 2020), cross-lingual Bengali corpus with approximately 1000 opinion or sentiment words (Sazzed et al., 2020), Phrasal lexicon with 1200, uni-gram lexicon with 3000, code-mixed data with 1500 words (Mandal et al., 2018). Moreover, Bhowmik et al. (2021) have developed a lexicon data dictionary using categorical weighting to find sentiment from Bengali data.

Table 2.2 is about existing lexicons and corpora of mono and multilingual sentiment analysis. The table shows existing lexical and knowledge resources are primarily in English and ignored resource-poor language like Bengali. Also, concept-based knowledge bases are very rare in a language other than English. The following SenticNet section gives a clear idea of the research on concept-level sentiment analysis resources.

Table 2.2: Existing Lexicons and Corpora of Mono and Multilingual Sentiment Analysis

References	Name	Type	Size	Language
Wilson et al., 2005a	OpinionFinder	Subjectivity lexicon	6856 unique entries	English
Wiebe et al., 2005	MPQA	Polarity corpus	1471 positive sentences and 3487 negative sentences	English
Wilson et al., 2005a, b	Subjectivity clues	Polarity corpus	English terms (2718positive and 4910 negative)	English
Yao et al., 2006	StarDict	Dictionary	Ten bilingual lexicons	Chinese-English
Blitzer et al., 2007	Multidomain sentiment corpus	Polarity corpus	4000 positive and 4000 negative Amazon product reviews	English
Xu et al., 2007	OPINMINE Chinese Sentiment annotation corpus	Polarity corpus	Annotation corpus of NTCIR6	Chinese
Mihalcea et al., 2007	Romanian NLP	Romanian NLP resources	50 million words newspaper articles	English Romanian
Seki et al., 2008	NTCIR Sentiment Analysis Pilot Task	Polarity corpus	2378 positive and 1916 negative sentences	Chinese
Wan, 2008	LDC_CE_DIC2.0	Dictionary	128,366 Chinese words and their synonym English words	Chinese
Wan, 2009	Product reviews	Polarity corpus	451 positive and 435 negative IT product reviews	Chinese
Constant et al., 2009	Amherst Sentiment Corpus	Polarity corpus		English, Chinese, German
Boyd-Graber & Resnik, 2010	Ding	Dictionary		English-German
Boyd-Graber & Resnik, 2010	HanDe	Dictionary		Chinese-German
Baccianella et al., 2010	SentiWordNet	Polarity lexicon		English

References	Name	Type	Size	Language
Che et al., 2010	Tongyici Cilin	Lexical database	17,817 synsets of 77,3443 Chinese words	Chinese
Boyd-Graber & Resnik, 2010	Movie reviews	Polarity corpus		German
Rosell & Kann, 2010	The People's Dictionary	Dictionary		English Swedish
Pan et al., 2011	Ling Pipe movies reviews	Polarity corpus	1000 positive and negative terms, respectively	English
Prettenhofer & Stein, 2011	Cross Lingual Sentiment	Polarity corpus	Amazon product reviews (800,000 reviews in four languages)	English, German, French, Japanese
Pan et al.,2011; NTCIR8, 2015	HowNet	Polarity lexicon	11,000 chinese sentences and 60,000 Chinese words	Chinese
Balahur & Turchi, 2014	NTCIR 8 Multilingual Sentiment Analysis Task	Polarity corpus	6223 opinion units	English
NTCIR8, 2015	Bing English-Chinese dictionary	Dictionary		Chinese
Dragoni et al., 2018	OntoSenticNet	Dictionary		English
Vilares et al., 2018	BabelSenticNet	Dictionary	Same as SenticNet	40 languages
Mandal et al., 2018	Phrasal lexicon	Polarity lexicon	3000 words	Bengali, English
Cambria et al. 2018	SenticNet 5	Dictionary	100000 concepts, 500000 semantics	English
Ali et al., 2020	BanglaSenti	Polarity corpus	61582 Bengali positive, negative and neutral words	Bengali
Sazzed et al., 2020	cross-lingual corpus	Polarity corpus	1000 opinion or sentiment words	Bengali, English
Cambria et al. 2020	SenticNet 6	Dictionary	200000 words	English
Bhowmik et al., 2021	lexicon data dictionary	Dictionary		Bengali

2.5.1 SenticNet

This research considered SenticNet as the source for Bengali knowledge base and polarity lexicon creation; therefore, this section described detailed literature of SenticNet

to expose the gap of similar Bengali resources. SenticNet is a concept-level knowledge base that plays an essential role in sentiment analysis jobs such as sentiment classification and is employed to create commonsense reasoning algorithms. SenticNet (Cambria et al., 2018) consists of semantic and affective data related to different commonsense knowledge arrangements. Therefore, the idea behind SenticNet is the creation of resources for NLP problems, such as sentiment analysis at the semantic level rather than syntactic. The journey of SenticNet was started in 2010. The first version of SenticNet was named as SenticNet 1, which consists of approximately 6,000 ConceptNet concepts having just polarity scores associated with them. The following extended version is SenticNet 2 with almost 13000 entries. All of these commonsense concepts consist of semantics and sentics, along with polarity. The knowledge base was further improved to a semantic network of 30,000 concepts in SenticNet 3. SenticNet 4 initiated the semantic primitives' concept and expanded the knowledge base to 50,000 concepts. Finally, in SenticNet 5, RNN was applied to deduce primitives through lexical substitution; thus, it reached 100,000 commonsense concepts.

The use of SenticNet is based on sentic patterns (Poria et al., 2015), which is the compilation of some syntax-based axioms that explain how the concepts will be interrelated in the sentences. It demonstrates the flow of sentiments (positive or negative) of the users from concept to concept based on the input sentences' dependency relationships. Human or manual assessment of languages to various alphabets, traditions, and kinds, established the robustness of the approaches and their capability of use in future research on multilingual concept-based sentiment analysis. SenticNet has been

tested through various researches (Poria et al., 2016; Ara´ujo et al., 2014; Cambria et al., 2010) for validity.

However, this resource faces the problem of unavailability of content other than English. A few attempts have been put to propose a prototype algorithm for developing concept-level knowledge bases for non-English languages. Nevertheless, they face the problem of maintenance, differences, and reproducibility due to the dependency on some heterogeneous resources. Moreover, creating a non-English edition of SenticNet from scrape also needs some resources, for example, AffectiveSpace (Cambria et al., 2018) or ConceptNet (Speer & Havasi, 2012) that are generally not available as per designated languages.

BabelSenticNet, a concept-level knowledge resource, was created by translating SenticNet (Vilares et al., 2018). This resource is considered the only multilingual concept-level knowledge resource that supports 40 languages and achieved around 64.4% polarities correctly. The development of the SenticNet for the designated language was started using statistical machine translation (SMT). The knowledge base was improved by using the synsets of WordNet.

Poria et al. (2012) have aggregated the emotion list from SenticNet (consists of numerical polarity values) and WordNet-Affect (consists of sentiment-related data) and assigned the emotion label to 2700 or more concepts. They have extended the sentiment labels stated in SemEval 2007 WordNet-Affect using SenticNet resource. This resource was greatly helpful in classification problems, such as improving the accuracy of classification. Cambria et al. (2012) developed a publicly available resource on semantics and affective information from an open mind corpus, known as SenticNet2. It is one of

the wide-ranging semantic resources for affect-sensitive appliances such as sentiment analysis and other social media mining, multilingual affective processing, etc.

Poria et al. (2013) have applied a machine-learning algorithm to automatically aggregate SenticNet with affective information from WordNet-Affect (WNA) and achieve 88.64% accuracy. To reach this solution, they have extracted many features from ISEAR (emotion dataset), used similarity measures based on the polarity value in SenticNet and WordNet, applied distance-based measures of ISEAR such as emotional affinity and point-wise mutual information. This resource is considered the most extensive dictionary (with quantitative polarity values and qualitative affective information) for sentiment analysis from single-mode or multimode, monolingual, or multilingual data.

Musto et al. (2014) used a lexicon-based approach (with exploration to most widely used lexical resources, i.e., SenticNet, SentiWordNet, MPQA, and WordNet-A) for classifying sentiments of Twitter posts from the datasets such as SemEval-2013 (Nakov et al., 2013) and Stanford Twitter sentiment (STS) data set (Go et al., 2009). SentiWordNet and MPQA were found to be well-performing than the other two resources on STS Dataset. The result from WordNet-A and SenticNet was not good at all.

A novel method was proposed that extends sentiment lexicons with their corresponding concept knowledge (Weichselbraun et al., 2014). The authors used concepts and their related polarity scores from SenticNet to spot out hazy sentiment terms, extracted context-related information to structure the knowledge sources like WordNet and ConceptNet. A quantitative assessment shows a noteworthy enhancement when using an extended version of SenticNet for sentiment classification. The evaluation

was carried on five different datasets such as electronics and software product reviews from Amazon, IMDB, comedy, crime, and drama.

SenticNet 3 uses ‘energy flows’ in place of dimensionality-reduction techniques and graph-mining to join various common-sense knowledge bases to one another. It also uses conceptual and affective data associated with multi-word expressions. Cambria et al. (2014) created SenticNet 3, having sentics and semantics to 30,000 common-sense concepts, and proved that SenticNet 3 and COGBASE could be easily merged in the application like social data mining. They have also developed tools and techniques to support SenticNet3, combining the other knowledge bases by improving the sentics and semantics from different media.

Dragoni et al. (2015) used fuzzy logic for generating concept polarities and finding uncertainty in them. The algorithm works based on the knowledge graph consists of two famous linguistics and lexical resources such as WordNet and SenticNet. This knowledge graph is then explored using a graph-propagation algorithm that predicts sentiment-related information from labeled datasets. This approach is tested using the Blitzer dataset and obtained higher accuracy than baseline datasets. Besides, found helpful during the calculation of the text sentiment.

Though the English language has the most vibrant set of polarity resources, many non-English languages such as Turkish, Bengali, etc., significantly lack such resources. To meet this gap, Dehkharghani et al. (2016) have created the first complete Turkish polarity resource based on SentiWordNet and SenticNet, and named it SentiTurkNet, having three polarities such as positivity, negativity, and neutral (objectivity). They have claimed, their methodology of creating polarity resources could be applied to any other

non-English language. Besides, the result generated with SentiTurkNet is found to be more accurate than the other related work.

Commonsense knowledge gives background information that allows machines to operate in the social media domain fruitfully. Other versions of SenticNet were lacking such knowledge for sentiment analysis and are imperfect to generalize. SenticNet 4 overcomes those limitations by influencing conceptual primitives generated via hierarchical clustering and dimensionality reduction. Cambria et al. (2016) first adopted this ensemble technique for identifying the primitives for noun and verb concepts and found outperforming previous versions of SenticNet.

AnRNN was applied by Jebbara et al. (2016) to detect the sentiments at the aspect level. For carrying out this experiment, they have used semantic knowledge from WordNet and SenticNet and found SenticNet was outperforming in extracting sentiment labels. This system is said to perform better for aspect-level SA than state-of-art systems. However, they did not test their system at the concept level.

Bajpai et al. (2016) have constructed Singlish sentiment Lexicon (knowledge base) for concept-level sentiment analysis on the top of the SenticNet framework. This resource was not created using manual labelings, like WordNet or DBPedia, instead applied automated techniques such as multi-dimensional scaling and graph-mining on commonsense knowledge. The knowledge was gathered from different sources and represented using three levels: vector spaces, semantic network, and matrix. The concepts were labeled via emotions as well as polarity.

Hassan et al. (2018) have proposed a substantial architecture for detecting Arabic sentences' sentiment polarity based on semantic features rather than syntactic features

using SenticNet. Their framework could deal with Arabic ambiguity, such as a gap of syntactic rules in slang Arabic. Their framework was tested using multi-domain dataset consists of 69K unique concepts and achieved 89% accuracy.

Detecting semantic similarity between different natural language texts (also called textual entailment) is a great challenge in computational linguistics. Many related types of research have been done so far and are mainly in English. Rudrapal, Das, and Bhattacharya (2015) have computed the Bengali text's similarity using WordNet. Bengali tweets (i.e., text) are found to be less noisy than English. They have defined partial textual entailment (PTE) on Bengali tweets as part of complete entailment for actual data.

Cambria et al. (2018) has created SenticNet 5 (a commonsense knowledge resource) that helps mingled symbolic and sub-symbolic AI to dig the conceptual primitives from textual data. This resource helped achieve 92.8% accuracy once applied to IMDB dataset. Dragoni et al. (2018) has applied ontology (OntoSenticNet) on SenticNet for sentiment analysis with some properties such as relating concepts with sentiment values, associating external data, and relating concepts with annotations. In their work, Cambria et al. (2020) has created a commonsense knowledge base, which integrated symbolic and sub-symbolic SA tools for polarity detection with the incorporation of deep learning architecture.

2.6 Sentiment Analysis Algorithms

Sentiment analysis is a form of classification problem. Many machine learning classifiers exist and are used in sentiment classification, such as linear classifiers (logistic

regression, NB classifier, fisher's linear discriminant), SVM, least-squares SVM, and quadratic classifiers kernel estimation (k-nearest neighbor). Besides, decision trees, boosted Tree, random forests, neural networks, learning vector quantization(Altrabsheh et al., 2014; Dashtipour et al., 2016; Lo et al., 2017; Al-Amin et al., 2017; Banik et al., 2018; Shirahatti et al., 2019).

One of this research's objectives is to propose and find the optimal algorithm (classifier) for multilingual sentiment classification; therefore, content analysis was conducted and presented in Table 2.3- Table 2.6. These tables show NB, SVM, and LSTM classifier gives better classification accuracy. NB is working better as the assumption of independence holds, and they scale well. Simultaneously, SVM works better as it avoids overfitting and can use different kernels especially linear kernels. Besides, LSTM is performing better as it accepts a large volume of training data. Moreover, there are some customize algorithms for dealing with multilingual sentiment analysis (MLSA). However, the existing survey in Table 2.3- Table 2.6 shows the clear gap of a customized algorithm for multilingual concept-level sentiment analysis.

Therefore, this section describes the basic concepts, related works of these three (NB, SVM, and LSTM) algorithms and shows the importance of using them in this thesis. Besides, some customized MLSA algorithms were discussed for better understanding the requirement of multilingual concept-level sentiment analysis algorithm.

2.6.1 Naive Bayes

Naive Bayes (NB) classifier is a basic probabilistic classifier dependent on the Bayesian theorem's use with self-determined presumptions. They are probabilistic, which implies that they ascertain each tag's likelihood for a given content (text) and output the tag with the most astounding one. These probabilities are calculated by utilizing Bayes' theorem. The theorem depicts the likelihood of the features based on earlier conditions of learning.

Bayes' theorem can be expressed mathematically as the following equation:

$$P(A|B)=\frac{P(B|A)\,P(A)}{P(B)} \tag{2.1}$$

Here, A and B are known as events. The probability of event A may be found only if given the value B is true. B is also known as evidence. P (A | B) is the posterior of probability. P (A) and P (B) are the prior probability (Tsao et al., 2013). The naïve Bayes assumption is that every feature in the dataset is independent and equally crucial for the outcome.

NB has been applied to different domains successfully and works well for NLP difficulties such as text classification. One of the fundamental reasons for working well on text data is the use of large 'vocabularies' or 'words.' The NB algorithm is broadly utilized for the document (both English, Bengali) classification (Aggarwal & Zhai, 2012; Korde & Mahender, 2012; Colas & Brazdil, 2006; Mamoun et al., 2014) due to its simplicity, accuracy, and reliability of results.

Due to its easiness and fastest performance, NB is considered the baseline in text classification (Xu, 2018). Kim et al. (2002) proposed and assessed some valuable techniques for enhancing the NB text classifier's outcome. They have suggested a

weighted mutual information text classifier to improve the consequences of very informative words. They have worked on the data set of Reuters and 20 newsgroups. For the assessment of the NB text classifier's arguments, He et al. (2007) used many smoothing techniques such as Witten-Bell, linear, absolute, and good turing on the data set of Yahoo! web scope having 3,894,900 questions. The authors claimed to achieve better performance than Laplace smoothing. Yuan (2010) has enhanced the text classification of NB by computing the posterior probability and trimming down the dimensions of feature words of the given text. They have classified 17910 documents of starter edition having nine categories and are said to achieve better results than the actual algorithm.

Gamollo et al. (2013.) did sentiment analysis in the Spanish language text of millions of tweets. Their system has found 67% accuracy using NB and classified the text into six categories: positive, negative, neutral, strong, average, and weak. Gamallo et al. (2014.) applied NB classifiers to find the sentiment from English tweets, which are doubtful human biases and too undersized to be linguistically explored. The tweet is matched with the polarity lexicon; if matched, found that corresponding polarity is considered; if not found, the polarity is assigned to be neutral. They have said to found 80% precision for the classification by the binary classifier. Some companies used the proposed system focused on natural language technology, such as Cilenis S. L., working with four languages like English, Portuguese, Spanish, and Galician.

Talbot et al. (2015) performed a nearby classification in SemEval-2015, where the text was classified using supervised classification techniques such as NB. They have claimed to improve the features of the lexicon by launching different pre-processing

techniques on Twitter data. They have used positive and negative word lexicon for classification and obtained an F-score value of 60% on the test data set.

A new NB classifier was proposed by Tsao et al. (2013) and named it semantic naïve Bayes classifier (SNBC). This classifier could obtain semantic information of the documents using planned semantic feature extraction processes and classifying algorithms. The feature extraction is done by converting each word of the document into the corresponding semantic vector and then extracting the word vector by applying principal component analysis (PCA). Finally, the original NB classifier merges with SNBC to classify the given document and outperforms traditional NB.

The main concept of naïve bays is to compute the probability of certain text documents D be a member of class C. Naïve bayes model is of two types such as multinomial model and multivariate Bernoulli model (Lewis, 1998; Vidhya et al., 2010; McCallum et al., 1998). The multinomial model is considered best in the case of a large dataset. However, it poses some problems, such as processing difficulty of data with a small number of training documents. NB Poisson model was proposed in (Kim, 2006) for text classification and handling the problem mentioned above. This model is said to improve the performance of the category mentioned above. Moreover, for multi-class text classification of the document, a two matrices model was proposed by Chen (2009), which could improve the performance of the NB by giving more emphasis on the features of higher correlation and sensitivity.

Melville et al. (2009) used a simple lexical classifier (works based on probability) to detect sentiment polarity based on the knowledge base (lexicon with positive and negative polarity developed in IBM India research labs for text mining (Ramakrishnan et

al., 2003). Sarkar et al. (2017) detected Bengali tweets' sentiment polarity using multinomial naïve Bayes with n-gram and SentiWordnet features. They have also tested different feature combinations and said to achieve 45% accuracy, which is better than similar work done in Sarkar et al. (2015).

Karim et al. (2016) used Gaussian Naïve Bayes (GND) and decision tree algorithms to classify the Bengali dataset with 2000 reviews related to the restaurant and achieved results with 65% accuracy. The authors recommended using GNB for dealing with real-valued data. Mandal et al. (2014) classified five different types of bangle news articles, namely, business, sports, health, technology, and education, using different classifier such as NB, KNN, etc. and have achieved 73%, 92%, 86%, 96%, 93% precision value using NB for business, sports, health, technology, and education, respectively.

Chy et al. (2014) classified Bengali news using NB. They have grabbed the data using their proposed data crawler. They have then tokenized the news documents with a lightweight Bangla stemmer. Also, performed necessary pre-processing for improving classification results and said to achieve a proper classification system. Kramer et al. (2014) carried a study on the IMDB dataset using the NB classifier and different feature combinations. They have achieved better accuracy than n-gram from the combination of minimal recursion semantics (MRS) and back-off replacement. Also, most of their feature combinations achieved an average accuracy of 89.09%.

Dhar et al. (2018) have successfully classified Bengali text documents (news data) based on three weighting methods such as TF, TD-IDF, TF-IDF inverse class frequency, and said to achieve 98.87% accuracy using an NB Multinomial algorithm. They have validated results from other famous classifiers such as NB, BayesNet, J48,

decision table, random tree, PART, RIPPER, and claim to found naïve Bayes multinomial as the best performer.

Al- Amin et al. (2017) used six different methods such as parts of speech ratio, cosine similarity using TFIDF, cosine similarity using custom TF-IDF, NB model with unigram & stammer, NB model with bigram, stammer & normalizer, Word embedding with Hellinger PCA to find the genuine sentiment of the Bengali text. They claimed the NB model with bigram, stammer, & normalizer outperforms others with an accuracy of 83.20%. However, it does not work well for large corpus; in this case, word embedding with Hellinger PCA works better. Haque et al. (2020) have used RF, SVM, NB, and LR for sentiment classification from the cricket dataset and have achieved 37% of classification accuracy with the adaptation of BOW and TF-IDF features.

From the above discussion, it is clear that NB is used in many multilingual sentiment analysis research; however, it is not used for assessing and finding optimal techniques (preprocessing feature, and concept extraction) and resources (lexical or knowledge resources, and datasets).

2.6.2 Support Vector Machine

A famous supervised machine learning classification algorithm was proposed by Hava Siegelmann and Vladimir Vapnik and named it support vector machine (SVM). SVM has been effectively applied in many natural language processing (NLP) tasks such as text classification (Lee et al., 2004; Lewis et al., 2004), part-of-speech (POS) tagging (Gimenez& Marquez, 2003; Nakagawa et al., 2001), terminology extraction, word sense

disambiguation (Lee et al., 2004), automatic summarization, noun phrase (NP) chunking (Kudo, T. & Y.Matsumoto. 2000), information extraction (Isozaki & Kazawa, 2002; Li et al., 2005), sentiment analysis, topic segmentation and dependency analysis (Kudoh et al., 2000; Yamada et al., 2003). The working procedure is almost the same for all these applications.

SVM is considered as one of the optimal classifiers due to some reasons such as 1) SVM offers the improved simplification capability for unseen data in classification than other similar classifiers such as decision trees (DT) or k-nearest neighbor (KNN). 2) SVM uses different kernel functions to investigate different types of feature combinations without adding computational complexity. 3) It could efficiently manage different feature combinations. 4) It could effectively classify linear and non-linear data. SVM works well for the data with no idea and semi-structured and unstructured data such as text. 6) It could be used to deal with data of higher dimensions. 6) SVM models are less prone to over-fitting problems.

In NLP tasks, instances are represented by higher dimensions and with scattered feature vectors, where positive and negative cases are distributed across the diverse parts of feature space. On the other hand, this helps the SVM find the classification hyperplane from the feature space and achieves good classification results in some NLP problems. Additionally, SVM could select helpful features efficiently and effectively from the vast number of features (as SVM learns by feature combinations with different weights) for some specific classification problems. Interestingly, numerous different algorithms require careful manual feature selection and are beneficial while applying the SVM to NLP issues. There are numerous kinds of NLP features from morphology, linguistic

structure (syntax), semantics, and knowledge sources such as thesaurus and gazetteers. In the function of SVM to NLP, those various types of features are simply assembled to shape one feature vector, and the learning would naturally figure out which features are helpful.

SVM creates a hyperplane in a high dimensional space for classification, outlier detection, and regression purposes. In the case of text classification, it creates a vector representing the text document according to distinct dimensions. In the larger text documents, the dimension may increase the computational cost; thus, features should be reduced to minimize the dimensionality (Joachims, 1998). The use of SVM in text classification was first introduced by Joachims (1998). The SVM application in text classification is separate from other classifiers because it needs a special training set for both types (positive or negative) of data.

This separate training set helps SVM to divide the data quickly to hyperplane (n-dimensional space) of the decision surface. The text documents that match closely to this surface are called support vectors. SVM also classifies the text better than other classifiers in combination with HMM (use as feature extractor) or NB (reducing features or dimensions) (Lin, 2002; Donghui, 2010; Qin et al., 2009).

Many types of research have been conducted to improve the outcomes of SVM. Ageev et al. (2003) assessed the effects of different features (such as feature space reduction and different kernel functions) for text classification by SVM. They have used 10372 documents from the university information system RUSSIA and are said to achieve improved accuracy of 1-5%.

Mahtab et al. (2018) have prepared a Bengali dataset on Bangladesh cricket and classified it into three classes using SVM and other algorithms. As per their experiments, SVM outperforms different classifiers by 4-10%. Das et al. (2010) have used SVM to classify news text and said to found the semantic orientation of the expression as positive or negative with lexicons and linguistic features. They have claimed to achieve 70.04%, 63.02% precision, and recall, respectively.

Bangla text content (Articles from online news) was categorized by Mostakim et al. (2018) using some supervised learning algorithms such as random forest (RF), SVM (rbf kernel), SVM (linear kernel), KNN, logistic regression (LR), gaussian NB. SVM works better with nearby accuracy to LR though in their study LR outperforms other classifiers. Rahman et al. (2018) used SVM, RF, KNN to classify two baseline Bengali datasets (restaurant and cricket) and found SVM outperforms other classifiers with precision values of 71% and 77% for cricket and restaurants, respectively.

Taher et al. (2018) have worked on Bengali text of different web sources to find the users' opinion. They have used the N-gram technique for vectorization and linear and non-linear SVM for classification. They have 92% accuracy for linear SVM and 89% accuracy for non-linear SVM. Pradhan et al. (2004) extended the shallow semantic parsing task of Jurafsky et al. (2002) and claimed that their algorithm based on SVM outperforms previous classifiers. They have validated the performance through the adaptation of new features.

Le Nguyen et al. (2005) applied an SVM to the CLANG corpus of sentences and their logical representation for semantic parsing. They said to achieve better results than a similar study. The study recommended SVM as the appropriate algorithm for semantic

parsing. Shin and Paek (2018) have shown task classification experiments using SVM on the data collected through Amazon Mechanical Turk. They have claimed the system could successfully classify little English messages to the predefined classes and achieve accuracy in the range from 82- 99% for any amount of data.

Orasan (2018) has proposed a method to detect aggression in text from social media using SVM and RF. The experiments have shown RF works well for known text, and SVM works well for hidden text. Therefore, SVM could be the best choice for text classification. Glavas et al. (2017) do cross-lingual topic classification on political texts data using linear SVM, SVM RBF, and CNN. As per their experiment results, multilingual data works better than monolingual data. Besides, linear SVM was the best performer in monolingual settings than SVM RBF and CNN. However, in a multilingual environment, CNN is the best performer (except the German language).

Chan et al. (2017) have tested different ensemble methods based on the diverse lexical feature set of native language identification. SVM and fully connected neural networks were considered base classifiers and trained them with the best performing features. Their result shows that the SVM ensemble works better, with an F1score value of 87%. Kaur et al. (2015) have used different supervised machine learning algorithms such as NB, SVM, and ANN to classify text from contents of Indian languages. They have found that all these algorithms are well suited for text classification despite many natural language processing issues. Sarkar et al. (2017) work on different feature combinations to detect the polarity of Bengali tweets' sentiment using multinomial naïve Bayes and SVM. Their study showed that SVM performs well for the SentiWordNet and unigram combination of features. Sazzed (2020) has applied SVM, LR, and RF ML

algorithms on novel datasets of multilingual nature (Bengali and English) and achieve an accuracy of 93%.

It is apparent from the above discussion, SVM is used in many MLSA research. However, these studies lack measuring and finding optimal techniques (i.e., preprocessing techniques and their combinations; feature and concept extraction techniques and their combination) and resources (i.e., lexical resources, knowledge resources, and datasets).

2.6.3 Neural Networks and Deep Learning

Eventually, a few machine learning methods, for example, NB, SVM, KNN, hidden Markov models, RF, and conditional random fields, are commonly used in NLP. However, some of these methods have been completely replaced or improved to a certain level by adapting neural models. This section has given a brief idea of all the neural models to justify the use of LSTM over other models in this study.

An artificial neural network (ANN) is a computational construct or model that resembles structurally and functionally to the human brain's neuron (vanGerven et al., 2017). A neural network consists of some interconnected nodes, or neurons, each of which could receive some inputs, deliver output through its input and output layers. In most cases, neural networks adopt an extra layer of nodes, generally called hidden layers, between the input and output layers. The layer is called a dense or fully-connected layer when the nodes in the hidden or output layers receive the input from each preceding layer. The nodes in the output layer do a weighted summation of the values it gets from the input nodes and produces an output.

There are different types of neural network classifiers based on their connection and the number of layers they adhere. The neural network with each node accepting the inputs only from the preceding nodes is called feed-forward neural networks (FFNNs). The FBNNs are where each node could accept data for themselves or its proceeding nodes. There is some confusion regarding the definition of a deep neural network (DNN). Generally, systems consist of multiple hidden layers are called DNN (Schmidhuber, 2015). Other classifiers consist of single or no hidden layer is usually called shallow. FBNN with a minimum of one hidden layer is called multilayer perceptrons (MLPs) (Rumelhart et al., 1985). Some DNNs are convolutional neural network (CNN), recursive neural network (RvNN), recurrent neural network (RNN), long short-term networks (LSTM) (Schmidhuber, 2015).

2.6.3.1 Convolutional Neural Networks

A convolutional neural network (CNN) is a kind of system that is not fully connected; that is, nodes receive the inputs from only some of the preceding layers (LeCun et al., 1989; LeCun et al. 1998). Convolutional neural networks use various functions such as filters to boost the data in changing ways and ensure concurrent analysis of the variety of features in the data (Krizhevsky, 2014; LeCun et al., 1995). The CNN structure is similar to that of biological neural networks (BNNs) (Hubel & Wiesel, 1962), where neurons receive the signals through their receptive field.

CNN has two significant advantages: the capability to share weights and accountability of weakly processed data, which minimizes the training (LeCun et al.,

1995). Secondly, the capability of reporting inconsistently and haphazardly arranged data. CNN can learn features that are residing in different areas of input data. Hence, few pre-processing required linguistic features to be recognized easily without concerning their placement in the text. Therefore, sentences with different word morphology, semantics, syntax, and other related features could be quickly processed (Kalchbrenner et al., 2014). CNN's are widely used in image, speech, video, and natural language processing (Santos & Gatti, 2014; Kalchbrenner et al., 2014; Kim, 2014; Zeng et al., 2014).

2.6.3.2 Recursive Neural Networks

One leading artificial neural network applied to date in natural language processing is a recursive neural network (RvNN) (Goller & Kuchler, 1996; Kawato et al., 1987). It is a bit similar to convolutional networks. It also has the capabilities of weight sharing that minimize the training.

Moreover, the way of weight sharing is different between these two types of ANN. CNN shares weights within a layer horizontally, and a recursive neural network shares weights between layers vertically to help NLP characterize the parse trees' structures quickly. In recursive systems, low-level and high-level trees use the single tensor of weights recursively and successively (Socher et al., 2011). A recursive neural network is considered to be feed backward as it relies highly on its previous layer results.

2.6.3.3 Recurrent Neural Networks

A simple and most widely used recursive neural network in NLP is called recurrent neural network (RNN) (Elman, 1990; Fausett, 1994). The RNN has a hidden layer with an edge on its nodes, which could feedback to themselves. Besides, it unfolds a collection of words or the elements like phonemes or sentences, which maintains the chain-like structure so that the previous inputs may be remembered in the ordered sequences in new data, which is considered exceptionally helpful (Mikolov et al., 2011; 2010; 2011b). To support backward dependencies that may exist in various cases, RNN looks at sentences in both forwards and backward directions using two cells and aggregate their outputs. This procedure is called bidirectional RNN. This RNN also solves the problems from erroneous production of higher dependent previous data. Bidirectional RNNs are most widely used in live applications, for example, speech processing programs. It is found that using two cells instead of one can create longer-term effects, which allow input to remain longer, and this arrangement is called RNN stack (Hihi & Bengio, 1996; Schmidhuber, 1992). This RNN stack could be of any number.

2.6.3.4 Long Short-Term Memory Networks

Today, Sentiment analyses are widely done using recurrent neural networks (RNN) and long short-term networks (LSTM). RNN and LSTM both introduce the memory in their network. The reason is to hold the previous results and help correlate with the present result to reach a more accurate solution. Single neurons or complex systems could be used in the internal mechanism of nodes in RvNNs or RNNs. One such

architecture of RNN is called LSTM network (Greff et al., 2017; Hochreiter & Schmidhuber, 1997). LSTM consists of numerous individual neurons in its recursive nodes that are connected and designed in a way to maintain precise information.

Conventional RNNs with single neurons sustaining back to themselves have some memory of quite long passed outcomes. However, these outcomes are weakened with consecutive iteration. Moreover, each component of the consequences is recalled similarly. As a rule, it is imperative to remember data completely from the removed past. By utilizing the LSTM structure in RNNs, this essential data can be held inconclusively by overlooking unimportant data.

On the other hand, if data are ignored, they cannot be recouped, regardless of whether its presence is needed later. Recent works addressed this issue conceivably by holding a larger quantity of information and substantially modifying specific models that are attentive to (Xu et al., 2015; Yang et al., 2016). RNN and LSTM are more helpful in text classification problems, as they highly need context knowledge. RNN could solve the problems of short-term dependencies in text. On the other hand, LSTM is more helpful for long-term dependencies in text. Moreover, a specific variant of the LSTM called the gated recurrent unit (GRU) is more efficient than standard LSTMs in performing many NLP undertakings (Cho et al., 2014; Chung et al., 2014).

Miedema et al. (2018) have used RNN and LSTM models in sentiment analysis and found the sentiments 86.74% correctly with the LSTM model in the validation set and without keywords tuning. Wang et al. (2018) have examined the consequences of LSTM and word embedding in text's sentiment classification process. They have first converted the given text into word vectors using the embedding model. They have then

imputed the word sequences to LSTM to learn the dependencies between words. The experiment concluded that, if the extensive training data given (maintaining both quality and quantity), a deep learning model such as LSTM outperforms other models.

Orabi et al. (2018) detected the depression of Twitter users (data from datasets such as CLPsych2015 and Bell Lets Talk) using two standard deep learning algorithms, such as CNN and RNN, on unstructured data. They have found CNN algorithm works well than RNN based algorithms. Sharfuddin et al. (2018) created a Bengali dataset (using different internet sources) and applied BiLSTM, one of the RNN models, to classify Bengali text's sentiment and achieved 85.67% of accuracy in classification.

A Bengali dataset with 6698 Bangla entries and 2639 Romanized Bangla text entries was created by Hassan et al. (2016) and made public for sentiment analysis and related application. They have applied a deep recurrent model, especially LSTM, with two-loss functions: binary and categorical cross-entropy. They have conducted 32 different experiments based on the dataset used and different loss functions and are said to achieve 78% accuracy in classification.

A thesis was conducted by Islam et al. (2018) on the Bangla dataset created by them. They have applied LSTM for Bengali sentence correction and completion. The encoding and decoding were done to achieve the goal. The encoder was bidirectional dynamic RNN with LSTM, and the decoder was raw RNN. They have attained 79% accuracy in the above mention job. Jebbara et al. (2016) used a neural network algorithm for the aspect-level sentiment analysis, where they have incorporated both the WordNet and SenticNet semantic knowledge bases. They have applied RNN to extract the features

and predict the sentiment and found that incorporating SenticNet improves the above study's performance.

Ma et al. (2018) have proposed an extended LSTM network and renamed it sentic LSTM for finding sentiment analysis at the aspect level by exploring commonsense knowledge. They have successfully trained recurrent encoder with commonsense knowledge and tested on two widely used datasets and said to found the proposed model (the combination of LSTM and SenticNet) outperforming the state-of-art models.

AI has achieved new force and prominence; however, its lack of expected performance in NLP due to the use of a bottom-up approach in place of the top-down approaches (Cambria et al., 2018). Cambria et al. (2018) have employed LSTM to detect the conceptual primitives from textual data by linking to the commonsense concepts and named entities automatically for sentiment analysis. They have also provided a way of extending SenticNet to form new knowledge bases that better help encoding sentiments and related semantics.

Akhtar et al. (2018) solved the data sparsity problem of aspect-level sentiment analysis by applying multi and cross-lingual word embedding learned by a parallel corpus. They have used an ensemble of handcrafted features and LSTM for sentiment classification at the aspect level. They have shown the comparative performance of the presented classifier against multi and cross-lingual setup. A deep Learning-based approach CNN was applied for sentiment analysis and aspect extraction (considered as multi-level classification problem) from multilingual data (Ruder et al., 2016). They have claimed to achieve consistent results for all domains and languages studied and suggested the future extension. Cambria et al. (2020) has used BiLSTM for sentiment classification

by adopting a new knowledge base known as SenticNet 6 on STS Dataset and has achieved 83.2% classification accuracy.

The discussion above and Table 2.3 - Table 2.6 shows that from among CNN, RvNN, RNN, LSTM, LSTM gives better performance on the scale of classification accuracy. Therefore, this study has used the LSTM classifier for all the conducted experiments (techniques and resources evaluation).

2.6.4 Multilingual Concept-level Sentiment Analysis Algorithms

One of the objectives of this thesis is to propose an algorithm for the multilingual concept-level sentiment analysis (MCSA). An overview of different algorithms used in existing monolingual and multilingual models is discussed to determine the gap in MCSA research. This section also summarizes existing resources and related performance for further investigation on monolingual and multilingual models.

Poria et al. (2014) and Bisio et al. (2017) proposed a model for finding the concept-level sentiment of the unstructured English text. Their model does semantic parsing of text and creates the bag of concepts (BOC), then the model matches the concepts with SenticNet, if found, derive sentic patterns; otherwise, classify with ELM classifiers. Cambria (2016) proposed a very similar model except introducing a dependency tree before semantic parsing. Besides, deep learning was adopted for both the feature-based and concept-based approaches. Their model was designed to address monolingual sentiment analysis only.

Tungthamthiti et al. (2014) have presented a model to address the sarcasm in concept level and supervised learning approach on monolingual (English) data. They have taken twitter data as input, pre-processed them. They then prepared those data for feature selection in three steps, such as concept-level expansion using concept-net, resolving contradictions in the sentiment score using SenticNet and SentiStrength, and finally by coherence. Finally, they have classified the sentiment score using SVM and achieved 80% accuracy. Mudinas et al. (2012) proposed the model pSenti for CLSA. They have used lexicon and machine learning-based techniques in combinations on English text and generated the sentiment score for both the approaches separately. In the end, the authors adjusted the sentiment weights and found the final sentiment with 82.30% accuracy.

A model for finding sentiment from Turkish text was proposed by Dehkharghani et al. (2016). They have created a polarity lexicon SentiTurkNet concerning WordNet for addressing Turkish text. They have tested their lexicon with three classifiers and got the highest average classification accuracy of 91.11%. An ELM-based model was proposed by Bajpai et al. (2016) to address the four dimensions (such as pleasantness, aptitude, sensitivity, and attention) of the concept level Hourglass model. In their model, the concepts were extracted first, feed to ELM classifiers, and classify them to the mentioned dimensions or classes. Finally, derive the intensity of each class with the use of SenticNet.

Poria et al. (2018) have proposed the model for SA using dynamic linguistic sentic patterns. In their model, the human lingual text is first decomposed into concepts and dynamically originated in the SenticNet. If it is not available, an SVM classifier was applied and opted the outcome by combining the output of sentic patterns. Gînscă et al.

(2011) developed a service tool (sentimatrix) for MSA. The model of sentimatrix consists of three modules such as pre-processing (with language detector, segmenter, tokenizer, and lexicon provider components), named entity recognition (with various heuristics, a custom rule engine interpreter, and in-memory storage of list entities components), sentiment fragment extraction (with locate signaling words, locate multiple modifiers, and final score calculator components).

A model proposed and classical research conducted on the Vietnamese language to find SA (Duyen et al., 2014). This model first receives hotel reviews as input, pre-processes the inputs, divides the reviews into sentences; this model then identifies the subjectivity and opinionated sentences. The model then classifies the sentiment, finds the sentences with the sentiment, and suggests hotels according to rank. Dehkharghani et al. (2017) proposed a model for SA of Turkish text. This model could address different linguistic issues at sentences, documents, and aspect levels. The working procedure of this model is as follows; it first breaks the documents into sentences, the sentences are then fed to the parser, and the parsed sentences are then matched with polarity lexicons such as SentiTurkNet and extracted the features. These features are used for sentiment classification.

Peng and Cambria (2017) developed the model of CLSA resource (CSenticNet) for the Chinese language. The procedure for implementation using this model is very similar to Poria et al. (2014). Al-Radaideh et al. (2017) proposed an Arabic sentiment analysis model. The model consists of four phases: a set of Arabic tweets collection, preprocessing, training, and testing. The Preprocessing phase consists of tokenization, normalization, stop words removal, and stemming. The training phase consists of term

weighting using TF-IDF, building a decision table, and reducting computation methods, and a rough set engine. Finally, the testing phase consists of rough set classifiers, classification process, and result evaluation of RS classifiers.

Al-Moslmi et al. (2018) have created an Arabic senti-lexicon (multi-domain Arabic sentiment corpus (MASC)) and a model to find the sentiment from the Arabic text. The lexicon consists of 8860 positive and negative feedbacks from different domains. They have used the lexicon as feature vectors and classified using various machine learning algorithms such as NB, KNN, SVM, LR, and NN classifiers. Their model consists of preprocessing, feature extraction (Arabic senti-lexicon), and classification evaluation units.

A model for predicting product helpfulness from multilingual product reviews was developed by Zhang and Lin (2018). Their model consists of different stages such as review acquisition, filtering, processing, creation of the database, predicting using statistical models. They have claimed to achieve 85.19% accuracy with their proposed model.

Asgarian et al. (2018) have proposed the model for Persian sentiment classification. Their proposed model is very similar to the work of Al-Moslmi et al. (2018), except the Persian sentiment lexicon was used in place of the Arabic senti-lexicon. Al-Saffar et al. (2018) proposed a model for Malay sentiment analysis using the Senti-lexicon algorithm, the steps in their model are very similar to the standard process of sentiment analysis, except they have ensemble the result of classification algorithms. They have tested their model with many features and algorithms and achieved a classification accuracy of more than 90%.

There is a model that classifies a sentence as out-of-vocabulary (OOV) or in-vocabulary (IV) using binary classifiers. This model then feeds the OOV sentences to the concept parser, and the concepts are then converted to International Phonetic Alphabet (IPA) using epitran (A python module for transliterating orthographic text as IPA). The model then matches the IPA of the OOV concepts to the PhonSenticNet (It is formed using SenticNet and its phonetics). Finally, in-vocabulary concepts are retrieved with the corresponding polarity from SenticNet (Satapathy et al., 2019).

Dashtipour et al. (2017) proposed a model to find the sentiments from a Persian movie review written in Persian text. The model consists of preprocessing, feature extraction, and classification section. The part-of-speech (POS) tags (such as verb and adverbs, adjectives, and nouns) and n-gram were used as features. The polarity was determined using a Persian lexicon known as PerSent lexicon, and finally evaluated the result of feature and their combinations using the SVM algorithm.

The discussed research work has used classic algorithms for evaluating the sentiment. However, some of the research work has used customized algorithms for finding sentiment polarity from monolingual data ignoring multilingual context. Few of them are discussed here.

Hassan et al. (2018) proposed a model of CLSA for processing Arabic text using SenticNet. Their model has checked the sentences for unigram and bigram; if found, they directly translated it to English; if not, they have normalized and transformed into unigram or bigram first then translated it. They then matched the translated words with the SenticNet, and if matched found, choose the corresponding polarity; otherwise, in case of a bigram, find the first word in the SenticNet; if found, take the corresponding

polarity value; otherwise, report not found. They have also followed the same procedure for matching with a lexicon dataset of SenticNet.

Xie et al. (2019) has proposed an improved SA algorithm to deal with English text. The algorithm has been tested on two datasets and has achieved a recall value of 90.5% by applying it on word and sentence levels. Bhowmik et al. (2021) has proposed an algorithm for sentiment score calculation from Bengali text that can generate the score using parts of speech and special character. The author has claimed to achieve 82.21% accuracy using the TF-IDF feature.

Table 2.3 shows the work summary of monolingual and multilingual models, algorithms, resources, and related performance. The Table reveals that the study done so far have mainly used standard algorithms like LSTM, CNN, NB, SVM, etc., and only a few works have used customized algorithms for dealing with multilingual data. However, those works have ignored the issue of concept-level sentiment analysis. Moreover, the studies used SenticNet (a concept-level knowledge base) primarily for English, keeping aside the language like Bengali. The dataset explored so far with the use of knowledge base, and multilingual data has ignored student feedback data.

Table 2.3: Monolingual and Multilingual Models, Algorithms, Resources and Related Performance

Reference	Language	Dataset	Lexical resources	Algorithm (Accuracy /Precision%)
Das et al., 2010	Bangla	Bengali news corpus	SentiWordNet	SVM (70.04)
Gînscă et al., 2011	English and Romanian	Internet sources	Sentimatrix	proposed (90)
Poria et al., 2013	English	Emotion Antecedents and Reactions (ISEAR)	Enhanced SenticNet	Naïve Bayes (71.20), MLP (74.12), SVM (88.64)
Poria et al., 2014	English	Semeval 2014 data, Blitzer-derived Dataset Stanford Sentiment Dataset (SSD)	SenticNet	Proposed parser results Semeval 2014 (91.25) Blitzer-derived Dataset (87.00) SSD (92.01)
Duyen et al., 2014	Vietnamese	Hotel Reviews		SVM (76.8) MEM (75.3), NB (70.4)
Musto et al., 2014	English	SemEval-2013 and Stanford Twitter Sentiment (STS)	SentiWordNet , WordNet-Affect , MPQA , SenticNet	Proposed Basic (54.47), normalized (54.59), emphasized (54.78) and emphasized-normalized (55.06)
Dashtipour et al., 2017	Persian	Persian Movie reviews	PerSent lexicon (Persian lexicon)	proposed (88.36)
Lu & Mori, 2017	English, Japanese, and Chinese	MDSU corpus		Parameter-sharing CNN (57.3)
Al-Radaideh et al., 2017	Arabic	Arabic tweets		SVM (60.30) KNN (KStar) (55.09) Decision Tree (52.63) Naïve Bayes (57.75)

Reference	Language	Dataset	Lexical resources	Algorithm (Accuracy /Precision%)
Peng & Cambria, 2017	English and Chinese	Chn sentiment corpus 2000, It168 and Weibo dataset from NLP&CC	CSenticNet	Proposed algorithm Chn2000 (54.85), It168 (59.04), Weibo (55.90)
Dehkharghani et al., 2017	Turkish	Turkish movie reviews	SentiTurkNet	proposed algorithm (79.56)
Hassan et al., 2018	Arabic	Hotels (HTL), Movies (MOV), Restaurants (RES#2) and Products (PROD) reviews	SenticNet	Proposed algorithm HTL(73), MOV(70), RES (85), PROD(73)
Al-Saffar et al., 2018	Malay	Malay Reviews Corpus (MRC)	Malay sentiment lexicon	NB (88.81), SVM(88.70), DBN (88.88)
Vilares et al., 2018	Spanish, Portuguese, Italian, Hindi and Chinese.		BabelSenticNet.	proposed algorithm Spanish (69), Chinese (70.7), Hindi (68.4)
Al-Moslmi et al., 2018	Arabic	MASC corpus	Arabic senti-lexicon	KNN(77.97), SVM (82.07), LLR (97.8), NB (96), NN (97.6)
Poria et al., 2018	English	Amazon Product Review Dataset Blitzer Dataset Movie Review Dataset	SenticNet and SentiWordNet	Movie review-SVM (74.59), ELM Classifier (74.27) Blitzer- SVM (76.00), ELM Classifier (75.46) Amazon dataset- SVM (72.63), ELM Classifier (71.40)
Mahtab et al., 2018	Bangla	Bangladesh Cricket, ABSA		SVM (64.59), Decision Tree (43.47) and Multinomial NB(58.38)

Reference	Language	Dataset	Lexical resources	Algorithm (Accuracy /Precision%)
Bhargava et al., 2018	Bangla, Hindi, Tamil	SAIL 2015 data	SentiWordNet	CNN (77.63%) and RNN–LSTM (77.63%) on the Hindi dataset, LSTM–CNN (57.37%) on the Bengali dataset, RNN–LSTM–RNN (69.19%) on the Tamil dataset.
Rahman et al., 2018	Bangla	ABSA (Cricket, Restaurant)		Cricket SVM (71), RF (60), KNN (45) Restaurant SVM (77), RF (64), KNN (54)
Taher et al., 2018	Bangla	News documents from social media		Linear SVM (91.68) Non-linear SVM (89.271)
Cambria et al., 2018	English	IMDB	SenticNet 5	LSTM (92%)
Satapathy et al., 2019	English	NUS SMS Dataset	SenticNet	Logistic -Regression (92.2), SGDC (89.8), SVC(91) Multinomial-NB (92)
Sazzed, 2020	Bangla	Bangla Drama Dataset	VADER,TextBlob, and SentiStrength	SVM(78%)
Cambria et al., 2020	English	STS dataset	SenticNet 6	biLSTM(83.82)
Bhowmik et al., 2021	Bangla	Cricket dataset	Domain-based categorical weighted lexicon	novel bangla text sentiment score algorithm (82.2%)
Sharmin et al., 2021	Bangla	Newspaper dataset		CNN (71.71%)

2.7 Sentiment Analysis of Student Feedback Datasets

Datasets are considered as the key resource for any SA research work. In search of resources, this research has found a gap of not having appropriate student feedback datasets both in English and Bengali. This section justified this claim through a comprehensive review of related works. Student feedback is the comments or post students' places in social media regarding the experience of their educational institutions, tutors, and services. Very few researches were conducted on student feedback, and some of them are discussed below:

Altrabsheh et al. (2014) have applied different methods for collecting and creating a dataset. They have collected a total of 1036 real-time student feedback from the University of Portsmouth. The author applied NB and SVM on the dataset and got 95% accuracy with SVM linear kernel. Welch et al. (2016) find the sentiment using student comments. The comments were extracted from Facebook and form a dataset of 1042 utterances. With entity-based evaluation, the SA result with this dataset outperforms the state-of-art datasets. Nguyen et al. (2018) have developed a dataset on Vietnamese students' feedback from 2014 to 2017. The dataset is in English language and included 16000-labeled sentences, and achieved the highest accuracy of 87.94% with maximum entropy algorithm.

Lalata et al. (2019) has created an English lingual student feedback dataset with 1822 comments where 1413 are positive, 327 are negative, and 82 are neutral. The authors have achieved the highest accuracy of 90.26% with NB by applying NB, SVM, DT, RF, and LR algorithms. Hujala et al. (2020) has created a student feedback dataset of

6087 responses in Finnish. The data were collected in the year 2016 to 2018 from Finnish universities. The authors have created the topic model by Latent Dirichlet Allocation and validated the outcome using the qualitative evaluation method. Hynninen et al. (2020) developed an English lingual dataset with 4990 student feedback collected from 2016 to 2018 and evaluated the emotion values using the Syuzhet library and the NRC lexicon.

An English dataset by Katragadda et al. (2020) has 30,000 text reviews along with student information. With three algorithms, such as NB, SVM, and ANN, the ANN algorithm showed better SA results with 88% accuracy. Neumann et al. (2021) have collected data from 99 students of undergraduate level to develop a dataset. However, though the participants are few, they have managed to collect a handsome amount of data (300-500 text documents) due to the broad coverage of topics. The authors have claimed to obtain efficient SA results with this dataset.

A comprehensive summary of student feedback datasets, lexicons, knowledge bases, algorithms, or other measures adopted for analysis and the performance of those researches in the scale of accuracy, precision, and recall are shown in Table 2.4. It is evident from the table and the above discussion that there is an insufficient number of student feedback datasets, especially in Bengali. The table also shows that SenticNet was not applied in any existing research (MLSA from student feedback), and the research was mainly supervised. Besides, the studies in the table used many algorithms; it is not clear which algorithm is better, and there seem to be no customize algorithms. Also, it is not evident that any knowledge sources (such as SenticNet) or polarity lexicons were tested for sentiment analysis from student feedback in the Bengali language.

Table 2.4: Works on Student Feedback Datasets

References	Dataset	Lexicon/ Knowledge base	Learning	Algorithms/ Measures	Result (Accuracy/ Precision/ Recall)
Altrabsheh et al., 2014	real-time feedback from lectures		Supervised	NB, CNB, ME, SVM	95%
Altrabsheh et al., 2014	real-time collection of feedback in lectures and end of unit feedback		Supervised	NB, CNB, SVM	94%
Welch et al., 2016	sentences from Facebook student group		Supervised	SVM	69.5%
Rajput et al., 2016	Student feedback	Sentiment dictionary	Unsupervised	Sentiment score	97%
Dhanalakshmi et al., 2016	student feedback from six programs		Supervised	NB, SVM, ANN, KNN	97.07%
Esparza et al., 2017	1040 comments of systems engineering students		Supervised	SVM	80%
Rani et al., 2017	Student feedback	NRC lexicon	Unsupervised	proposed	90%
Sivakumar et al., 2017	Twitter data		Supervised	NB, CNB	85%
Aung et al., 2017	student feedback	Lexicon	Unsupervised	Opinion score	Positive
Gottipati et al., 2017	student feedback	Lexicon (Senti WordNet)	Unsupervised	NB	82%
Nasim et al., 2017	1230 comments	Lexicon	Supervised	Opinion score	92%
Cabada et al., 2018	Student feedback	SentiText and EduERAS	Supervised	CNN, LSTM	88.26% and 90.30%
Sultana et al., 2018	Educational dataset		Supervised	SVM, ANN, NB	78%
Nguyen et al., 2018	Students' Feedback Corpus (UIT-VSFC)		Supervised	SVM, LSTM	82.23% and 90.2%
Yu et al., 2018	undergraduate students		Supervised	SVM, CNN	66%
Nguyen et al., 2018	UIT-VSFC		Supervised	ME, NB	87.94%

References	Dataset	Lexicon/ Knowledge base	Learning	Algorithms/ Measures	Result (Accuracy/ Precision/ Recall)
Ibrahim et al., 2018	collected from students and consists of 979 instances		Supervised	SVM, NB, DT, RF	76%
Atif, 2018	Survey		Supervised	proposed	80%
Öhman et al., 2018	approximately 9,000 English annotated sentences		Supervised	NB, MLP	50%
Tripto et al., 2018	Bangla, English and Romanized Bangla comments from YouTube videos		Supervised	NB, SVM, CNN, LSTM	65%
Dsouza et al., 2019	Students' Feedback		Supervised	NB, SVM, RF	81%
Sengkey et al., 2019	Students' Feedback		Supervised	SVM	74%
Kandhro et al., 2019	student's feedback feedback/comments database was built through Google forms		Supervised	LSTM	85%
Sindhu et al., 2019	Students' feedback		Supervised	LSTM	91%
Bhargava et al., 2019	SAIL 2015 tweet data		Supervised	LSTM, CNN	57.37%
Lwin et al., 2020	Students' feedback		supervised	SVM, NB	97%
Sangeetha et al., 2020	Vietnamese student feedback		supervised	LSTM	86%

2.8 Feature Extraction

The feature is the information that could be extracted from any data set (Avinash et al., 2019). Feature selection is the process of choosing a subset from the original set of features. At the same time, feature extraction is getting useful features from existing or raw data. Sentiment analysis requires using NLP and text mining tools and techniques for finding the polarities. These tools could find the actual polarities (positive, negative, neutral) provided that appropriate features or features are ensured (Meyer-Baese et al., 2014; Avinash et al., 2019). Therefore, feature extraction plays a vital role in sentiment analysis. Feature extraction is also a dimensionality (feature) reduction (select or combine variables into features) method that could still describe the original data set entirely and accurately. It could reduce the redundant data from the analysis, provide more accurate performance measures, and speed up the machine learning process (Guyon et al., 2006; Hira et al., 2015; Avinash et al., 2019).

The feature extraction techniques or their combinations applied successfully on English data are negative word, stemming, cluster functional word, parts of speech, and chunk. Also, dependency tree feature, n-gram (unigram, bigram, trigram), the position of words, bag-of-words features, aspects (Das et al., 2010; Altrabsheh et al., 2014; Mahendran et al., 2018; Yu et al., 2018; Nguyen et al., 2018; Cabada et al., 2018; Shirahatti et al., 2019; Sindhu et al., 2019; Jamatia et al., 2019; Sazzed et al., 2019). The feature extraction techniques or their combination applied successfully on Bengali data are negative word, stemming cluster functional word, parts of speech, and chunk (Ray et al., 2015; Islam et al., 2016; Sarkar et al., 2017; Eshan et al., 2017; Islam et al., 2017; Alam et al., 2017; Rani et al., 2017). Additionally, dependency tree

feature, words, characters, n-gram (unigram, bigram, trigram), TF, IDF, rule-based, content-based, and context-based features (Mahtab et al., 2018; Taher et al., 2018; Gope et al., 2018; Banik et al., 2018; Dhar et al., 2018; Sharfuddin et al., 2018; Rahman et al., 2019; Milu et al., 2020). The feature extraction techniques or their combination successfully applied on both English, and Bengali data are negative word, stemming cluster functional word, parts of speech, chunk, dependency tree feature, words, characters, n-gram, bag-of-words features (Das et al., 2010; Ray et al., 2015; Nguyen et al., 2018; Bhargava et al., 2019; Jamatia et al., 2019; Sazzed et al., 2019). Some of the works that have adopted the feature extraction techniques mentioned above are :

Jamatia et al. (2019) have extracted the n-gram features from English and Bengali lingual data for classifying by CRF, LSTM, and Bi-LSTM algorithms and achieved an overall accuracy 88.27%. Sindhu et al. (2019) have extracted aspects from English student feedback data, applied LSTM for SA, and achieved 91% accuracy. Rahman et al. (2019) have performed SA on Bengali lingual Facebook data and achieved 52.98% accuracy using n-gram, TF-IDF, and POS features with NB, SVM, KNN, DT, and K-means algorithm. Sazzed et al. (2019) have attained 70% accuracy using unigram and bigram features on English and Bengali lingual Facebook data. The authors have used four classic algorithms as LR, RF, SVM, and LSTM for SA.

Milu et al. (2020) have extracted TF, TF-IDF, applied NB, SVM, LR, RF algorithm on Bengali data for SA, and got 88.05% accuracy. Basarslan et al. (2020) have used TF-IDF and word2vector feature techniques on the IMDB English dataset. The study has attained 90% overall SA accuracy using NB, SVM, and ANN. Sangeetha et al. (2020) have applied n-gram and dimensionality reduction extracting

techniques on Vietnamese student feedback for SA and achieved 86% accuracy with the LSTM algorithm. A study used IMDB English lingual dataset for SA using Word count and encoding feature extraction techniques. The study has attained 90.5% accuracy using LSTM (Hameed et al., 2020). Hasan et al. (2020) have used the n-gram feature extraction technique on the Bengali Cricket dataset and achieved an overall accuracy of 69% using RF, SVM, and CNN algorithms. Bhowmik et al. (2021) have extracted TF-IDF and bi-gram features from the Bengali Cricket dataset, applied SVM and novel Bengali text sentiment score algorithm, and achieved an accuracy of 82.2%.

The state-of-the-art literature shows that almost every sentiment classification task requires feature extraction; some are shown in Table 2.5. This table represents the summary of sentiment classification research that has adopted feature extraction techniques and the performance of those researches. The table also represents information such as investigated feature or feature set, domains, and languages of data explored, lexicon or knowledge base used, and machine learning algorithms or measures applied and their performance. As we know, the more the classification accuracy, the better the classifier is.

Thus, we can conclude from the above discussion and the summaries in Table 2.5 that the solely used feature extraction technique can not be considered only criteria for performance enhancement. However, adopting a proper combination of feature extraction techniques is the key criteria for performance enhancement. The table also shows that the studies are mainly supervised, and NB, SVM, and LSTM are better than other algorithms. Besides, the analysis was mostly done on n-gram features by ignoring other feature combinations and concepts.

Table 2.5: Summary on Feature Extraction Task in State-of-Art Research

References	Language	Data	Features techniques	Lexicon/ Knowledge base	Algorithms/ Measure	Performance (Accuracy)
Das et al., 2010	English, Bangla	MPQA, IMDB, NEWS, BLOG	Negative Word, Stemming Cluster Functional Word, Parts Of Speech, Chunk, Dependency tree feature	Senti WordNet	Sentiment score	70.04%
Altrabsheh et al., 2014	English	real-time feedback from lectures	Unigram, bigram, trigram		NB, CNB, ME, SVM	95%
Ray et al., 2015	Oriya, Bangla	printed Oriya text	Words, characters		BLSTM	95.82%
Islam et al., 2016	Bangla	Facebook comments	Unigram, bigram		NB	72%
Sarkar et al., 2017	Bangla	Bengali tweets	Unigram, bigram, trigram	SentiWordnet	NB, SVM	45%
Eshan et al., 2017	Bangla	Facebook post	Unigram, bigram, trigram		RF, MNB, SVM	90%
Islam et al., 2017	Bangla	Bengali document corpus	TF-IDF		NB, SVM, SGD	92.56%
Alam et al., 2017	Bangla	Bangla comments	Unigram, bigram	Sentiwordnet	SVM, CNN	99%
Rani et al., 2017	English	Student feedback	n-gram	NRC lexicon	Proposed	90%
Mahtab et al., 2018	Bangla	ABSA	TF-IDF		NB, SVM	73%
Taher et al., 2018	Bangla	web based diverse data	n-gram		NON LINEAR SVM, LINEAR SVM	91.684%
Gope et al., 2018	Bangla	Bangla PDFs	Rule-Based, content-based, and context-based Features		Proposed	86%
Banik et al., 2018	Bangla	Bangla Movie Database	Unigram, bigram		NB, SVM	86%
Dhar et al., 2018	Bangla	Text documents of Business, Medical, State, Sports, Technology domain	TF, TF-IDF		LIBLINEAR, Multinomial NB, NB, RF	97%

References	Language	Data	Features techniques	Lexicon/ Knowledge base	Algorithms/ Measure	Performance (Accuracy)
Mahendran et al., 2018	English	User feedback	Bigrams, Unigrams, position of words and Parts of Speech (POS)		NB, SVM, ME	94%
Sharfuddin et al., 2018	Bangla	Facebook comments	n-gram		SVM, DT, LLR, BiLSTM	85.67%
Yu et al., 2018	English	undergraduate students feedback	n-gram		SVM, CNN	66%
Nguyen et al., 2018	English, Vietnamese	Students' Feedback Corpus (UIT-VSFC)	bag-of-words features with unigram and bigram		SVM, LSTM	82.23% and 90.2%
Cabada et al., 2018	English	SentiText and EduERAS	n-gram		CNN, LSTM	88.26% and 90.30%
Gamal et al., 2019	Arabic	Twitter posts	Unigram, bigram, trigram		NB, SVM	85.97%, 85.32%
Rahman et al., 2019	Bangla	Facebook groups	n-gram, TF-IDF, POS		NB,SVM, KNN, DT, K-means	52.98%
Shirahatti et al., 2019	English	Social Media	Unigram, bigram		GloVe –DCNN	88%
Sindhu et al., 2019	English	Students' feedback	Aspects		LSTM	91%
Bhargava et al., 2019	Hindi, Tamil, Bangla	SAIL 2015 tweet data	n-gram		LSTM, CNN	57.37%
Jamatia et al., 2019	English, Bangla, Hindi	Facebook, Twitter and WhatsApp	n-gram		CRF, LSTM, Bi-LSTM	88.27%
Sazzed et al., 2019	English, Bangla	Social Media	Unigram, bigram		LR, RF, SVM, LSTM	65% and 70%
Milu et al., 2020	Bangla	Comments from social sites and online resources	TF, TF-IDF		NB, SVM, LR, RF	88.05%
Basarslan et al., 2020	English	IMDB	TF-IDF, word2vector		NB, SVM, ANN	90%
Sangeetha et al., 2020	English	Vietnamese student feedback	n-gram, dimensionality reduction		LSTM	86%

References	Language	Data	Features techniques	Lexicon/ Knowledge base	Algorithms/ Measure	Performance (Accuracy)
Hameed et al., 2020	English	IMDB	Word count, encoding		LSTM	90.5%
Haque et al., 2020	Bangla	Cricket dataset	BOW, TF-IDF		RF, SVM, NB, LR	37%
Hasan et al., 2020	Bangla	Cricket dataset	n-gram		RF, SVM, CNN	69%
Bhowmik et al., 2021	Bangla	Cricket dataset	TF-IDF, bi-gram	Domain-based categorical weighted lexicon	SVM, novel bangla text sentiment score algorithm	82.2%

2.9 Preprocessing Techniques

Pre-processing is an essential step of data preparation for sentiment classification (Haddi et al., 2013; Krouska et al., 2016). Pre-processing is the cleaning, normalizing, transforming, and preparing of the data for sentiment classification (Turban et al., 2010; Haddi et al., 2013). It is also helpful in dimensionality reduction (Ghag et al., 2015). The data in social media are noisy and contain some unnecessary parts such as advertisements, scripts, HTML tags, irrelevant words known as stop words etc. these data create some additional dimensions in classification and make the task more difficult (Kumar et al., 2019). The hypothesis is to process the data correctly, which will minimize the problems and help improve the sentiment classification (Turban et al., 2010; Haddi et al., 2013; Ghag et al., 2015; Krouska et al., 2016).

Preprocessing for English data involves the following tasks: tokenization, annotation, POS tagging, stop-words removal, punctuations, numbers, symbols, stickers, emojis, and other special characters character repetition and communication abbreviations removal. Besides, normalization, URL and user tags removal, stemming, segmentation, replacing abbreviations, merging stem words, symbolic emoji replacement, and negation resolution. Additionally, other preprocessing task are lemmatization, case conversion, multiword grouping, hashtag removal, noun phrase removal, labeling, white space removal, handling words with apostrophe (Camacho-Collados et al., 2017; Esparza et al., 2017; Nasim et al., 2017; Jain et al., 2017; Sivakumar et al., 2017; Nguyen et al., 2018; Raiyani et al., 2018; Zobeidi et al., 2019; Sindhu et al., 2019; Jamatia et al., 2019).

Bengali is an inflected language facing major challenges like insufficiency of a labeled dataset and lack of NLP tools (Karim et al., 2013). Negation handling is a big issue in Bengali SA (Hasan et al., 2015; Paul et al., 2016). Amazon watch dataset was converted to Bengali datasets by applying different standard preprocessing techniques and achieved better accuracy once applied negation (Hasan et al., 2015; Paul et al., 2016). Islam et al. (2016) tackled the negations through antonym substitution along with basic preprocessing. Another issue with Bengali data is handling complex multi-lines in a document, where Al-Amin et al. (2017) have used bigram, stemmer, and normalize to handle it technically. A general issue with Bengali lingual data is the format of web or Facebook reviews. There is no hard and fast rule regarding this issue. Therefore, users post the comments as per their will. This issue is handled by Akter et al. (2016) with a dictionary-based approach. Some other issues with Bengali SA are handling inflected Bengali word, and spelling error and tried to resolve by applying to small dataset (Patra et al., 2015).

A more complicated issue with Bengali lingual data is the colloquialisms that humans and machines are very hard to process. This issue creates dimensionality that makes SA more complex (Kumar et al., 2019). A more concerning issue in preprocessing Bengali text is removing special characters and punctuations. It creates complexity in text classification. Notable that, full stop "." in English is represented by the "|" character in Bengali text (Haque et al., 2020). A very sophisticated issue with Bengali text is its verb, which has multiple forms and is used in different places in a sentence for a different meaning. Haque et al. (2019) have dealt with this matter by building bag-of-words of multiform Bengali verbs. Another significant issue is the text canonicalization. This issue is seen to be not dealt with in current research. However, a very similar form is adopted in some research (Bhowmik et al., 2021).

In general, preprocessing for Bengali data involves the following issues such as tokenization, normalization, negation resolution, stemming, punctuation and number removal, stop words removal, emoticons and noun phrase removal, Hashtag, URL removal, symbols, numbers, and stickers removal (Chowdhury et al., 2014; Ghosal et al., 2015; Hassan et al. 2016; Islam et al., 2016; Paul et al., 2016; Alam et al., 2017). Besides, annotation, removed the Dari (j), which is equivalent to English Full Stop (.), convert the comments to one line, removed slangs and corrected the misspelled Bengali words (Sumit et al., 2018; Wahid et al., 2019; Sharfuddin et al., 2018; Mahtab et al., 2018; Banik et al., 2018). Additionally, Non-Bengali character and symbol removal, user tags removal, labeling, filtering non-Bengali words (Taher et al., 2018; Bhargava et al., 2019; Jamatia et al., 2019; Zobeidi et al., 2019; Sarkar, 2019; Milu et al., 2020). Selecting an appropriate preprocessing method or combination of methods is crucial in improving sentiment classification accuracy (Haddi et al., 2013; Ghag et al., 2015).

As per the discussion in the previous paragraphs, many issues with Bengali text preprocessing are similar to English. However, it still faces huge problems in dealing with those issues due to the unavailability of tools and the proper combinations of techniques (Karim et al., 2013; Al-Amin et al., 2017; Rahman et al., 2019; Sazzed et al., 2020; Bhowmik et al., 2021).

The state-of-the-art literature shows almost every sentiment classification task requires preprocessing; some are shown in Table 2.6 and Table 2.7. The table represents the summary of sentiment classification research that has adopted preprocessing techniques along with the performance of those researches. The table also represents information such as investigated level of sentiment analysis, domains

and languages of data explored, learning method adopted, and machine learning algorithms used in those researches.

From the above discussion and the summaries in the Table 2.6 and Table 2.7 we can conclude that the overall performance of the investigations indicates, applying preprocessing techniques solely is not only the criteria for performance enhancement but also adopting a proper combination of preprocessing techniques is the critical criteria for performance enhancement. The table also shows that the studies were mainly supervised, and NB, SVM, and LSTM are better than other algorithms. Moreover, the studies emphasized SA at the sentence level by ignoring the feature and concept level.

Table 2.6: Summary of Bengali SA that Applied Preprocessing Techniques

Reference	Analysis level	Language	Data (domain)	Preprocessing	Learning method	Algorithms	performance (Accuracy)
Chowdhury et al., 2014	Sentence	Bangla	Social media Reviews	Tokenization, normalization	Supervised	SVM, ME	93%
Ghosal et al., 2015	Sentence	Bangla	Horoscope	Punctuations, stop words removal	Supervised	NB,SVM, KNN, DT, RF	98.7%
Paul et al., 2016	Document	Bangla	Product reviews	Stop words removal, negation resolution	Supervised	NB	85%
Hassan et al. 2016	Sentence	Bangla	Social media, news, product reviews	Emoticons, noun phrase removal	Supervised	LSTM	70%
Islam et al., 2016	Sentence	Bangla	Social media Reviews	Hash tag, url removal, stemming	Supervised	NB	77%
Alam et al., 2017	Feature	Bangla	Bangla comments	removed the URLs, symbols, Dari (j) which is equivalent of English Full Stop (.), converted the comments to one line, Removed slangs and corrected the misspelled Bangla words.	Semi supervised	SVM, CNN	99%
Taher et al., 2018	Sentence	Bangla	Social media Reviews	negation resolution, emoticons and punctuation removal, stemming	Supervised	SVM	91.68%
Banik et al., 2018	Sentence	Bangla	Social media Reviews	URL, emoticon, punctuation, and stop words removal	Supervised	NB, SVM	86%
Mahtab et al., 2018	Sentence	Bangla	Social Media Reviews	punctuation and number removal, Tokenization, and stop words removal	Supervised	SVM	64%
Sharfuddin et al., 2018	Sentence	Bangla	Social Media Reviews	Emojis, symbols, numbers, stickers, English letters removal	Supervised	RNN	85.67%
Sumit et al., 2018	Sentence	Bangla	Social Media Reviews	Non Bengali character and symbol removal	Supervised	Word2Vec	83.79%
Wahid et al., 2019	Sentence	Bangla	Social Media Reviews	Tokenization, URL, punctuation, user tags, stop words removal	Supervised	NB, SVM, LSTM, CNN,	95%

Reference	Analysis level	Language	Data (domain)	Preprocessing	Learning method	Algorithms	performance (Accuracy)
Sarkar, 2019	Sentence	Bangla	Social media Reviews	Special characters removal, Stop word removal	Supervised	CNN	46.80%
Bhargava et al., 2019	Feature	Hindi, Tamil, Bangla	SAIL 2015 tweet data	Tokenization, labeling	Supervised	LSTM, CNN	57.37%
Jamatia et al., 2019	Document	English, Bangla, Hindi	Facebook, Twitter and WhatsApp	Tokenizing, annotation	Supervised	CRF, LSTM, BiLSTM	88.27%
Milu et al., 2020	Feature	Bangla	Comments from social sites and online resources	Filtering non-bengali words, Punctuations, numbers and other special characters removal, stemming	Supervised	NB, LR, SVM, RF	88.05%
Haque et al., 2020	Feature	Bangla	Cricket dataset	Removing punctuation and special character	Supervised	RF, SVM, NB, LR	37%
Hasan et al., 2020	Feature	Bangla	Cricket dataset	Removing stop word, invisible characters, URL, punctuation and hash tags, removing Romanized text	Supervised	RF, SVM, CNN	69%
Bhowmik et al., 2021	Feature	Bangla	Cricket dataset	Normalization, tokenizing, stemming	Supervised, unsupervised	SVM, novel bangla text sentiment score algorithm	82.2%

Table 2.7: Summary of SA Research that Applied Preprocessing Techniques

Reference	Analysis level	Language	Data (domain)	Preprocessing	Learning method	Algorithms	performance (Accuracy)
Camacho-Collados et al., 2017	Document	English	Social media Reviews	Vanilla text tokenization, Lowercasing Lemmatizing, Multiword grouping	Supervised	CNN, CNN+LSTM.	91.2%, 88.9%
Esparza et al., 2017	Sentence	Spanish	Student Feedback	stop words and nouns deleted, punctuation removal, case conversion	Supervised	SVM	80%
Nasim et al., 2017	Feature	English	Student Feedback	Punctuations, numbers and other special characters removal, Tokenization, Case Conversion, Stop words	Supervised	RF, SVM	93.4%, 92.9%
Jain et al., 2017	Feature	English	Social media Reviews	tokenization, stop-words removal, stemming, lemmatization	Supervised	NB, SVM	72.81%
Sivakumar et al., 2017	Feature	English	Student Feedback	tokenization, stemming, stop word removal	Supervised	NB	97.0%
Yenter et al., 2017	Feature	English	IMDB	Punctuation and other symbols removal	supervised	CNN-LSTM	89%
Mathapati et al., 2018	Feature	English	IMDB	Encode the reviews	supervised	NB, LSTM	91.8%
Raiyani et al., 2018	Sentence	English, Spanish, and Arabic	Social media Reviews	Merging of Stem Words, Users in Tweets.convert to 'user', Symbolic Emoji Replacement. Segmentation. Character Repetition, Stopwords, and Communication Abbreviations removal	Supervised	SVM	English (72.79%) Spanish (72.20%) and Arabic (64.36%)
Nguyen et al., 2018	Document	English	Vietnamese student feedback	Normalization, Segmentation, replacing abbreviations	Supervised	NB, ME	86.1%, 87.9%

Reference	Analysis level	Language	Data (domain)	Preprocessing	Learning method	Algorithms	performance (Accuracy)
Zobeidi et al., 2019	Feature	Persian	Social media Reviews	Removal of punctuation marks, Normalization, Stemming, Stop-words removal, Tokenization	Supervised	Word2vec + CNN + BLSTM	95.0%
Sindhu et al., 2019	Feature	English	Students' feedback	Stop-words removal, Tokenization, POS tagging	Supervised	LSTM	91%
Bhargava et al., 2019	Feature	Hindi, Tamil, Bangla	SAIL 2015 tweet data	Tokenization, labeling	Supervised	LSTM, CNN	57.37%
Lwin et al., 2020	Feature	English	Students' feedback	Tokenization, Stop-words removal, punctuation, extra whitespaces, and other symbols removal,	Supervised	SVM, NB	97%
Sangeetha et al., 2020	Sentence	English	Vietnamese student feedback	tokenization, stop-words removal, removing repeated letters and unwanted symbols	Supervised	LSTM	86%
Basarslan et al., 2020	Feature	English	IMDB	Stop word removal, stemming, converted to lower case	Supervised	NB, SVM, ANN	90%
Hameed et al., 2020	Feature	English	IMDB	Punctuation and stop word removal, converted to lower case, omitted lemmatization	Supervised, unsupervised	LSTM	90.5%

2.10 Natural Language Processing Tools

There are several tools freely accessible to natural language processing tasks. These toolkits could be applied with distinct programming languages. Hence, users can select any suitable toolkits for their NLP-related applications. NLP toolkits like NLTK (Loper et al., 2002), spacy (Hannibal, 2017), Stanford NLP (Manning et al., 2014), etc., are made easy to operate and open-sourced (Krithika et al., 2014). The toolkits were mostly developed for the English language; even then, some toolkits were designed for non-English languages due to the availability of a considerable number of non-English contents. 'Apertium' was developed for Spanish, French, Italian, and Danish text and consists of sentence splitter, POS tagger, and tokenizer (Gralinski et al., 2013; Brandt et al., 2011) whereas, 'IceNLP' was developed for Icelandic language texts and consists of POS tagger, Pre-processor, and finite-state parser (Brandt et al., 2011; Loftsson et al., 2007).

Some popular toolkits are the carabao language toolkit- which supports any natural language objects; Ellogon consists of a multilingual graphical user interface for end-user (Petasis et al., 2002). Besides, monty lingua- was developed in Python language and includes all the features of text processing and summary generation in English (Ling, 2006); Stanford CoreNLP consists of several tools that facilitate different NLP functionalities related to statistics, deep learning, and rule-base. In general, most of the NLP toolkits were designed with parts of speech (POS) tagger, tokenizer, sentence splitter, and lexica (Krithika et al., 2014).

Other renowned toolkits are spaCy (Honnibal, 2017), AllenNLP (Gardner et al., 2018), PSI-Toolkit (Jassem, 2012), General architecture for text engineering

(GATE) (Cunningham et al., 2011), Intel NLP Architect (Mamou), Flair (Chinkina et al., 2016), Gensim (Orasmaa et al., 2016), TextBlob (Orasmaa et al., 2016), Textacy (Anju et al., 2018), OpenNLP (Baldridge, 2005), CogCompNLP(Khashabi et al., 2018).

Natural language toolkit (NLTK) is considered the most favorite toolkits due to its features, multi-language and algorithm support, component availability for classification, semantic reasoning, tagging, and tokenization, parsing, and stemming. In general, it is an excellent toolkit for the experiments and applications that require the use of specific collections of algorithms (Bird et al., 2009; Loper et al., 2002). Therefore, this study used the NLKT tool for experiments. This thesis used google translator for translating SenticNet to BanglaSenticNet as well as for translating text from corpora. Some other tools used to date for multilingual sentiment analysis are shown in Table 2.8.

Table 2.8: Tools for multilingual Sentiment analysis

References	Name	Type	Language
Denecke, 2008	PROMT eXcellent Translation Technology	Translator	German,French, Portuguese, English, Spanish, Italian and Russian
Bautin et al., 2008	WebSphere Translation Server of IBM	Translator	Many
Wan, 2008, 2009; NTCIR8, 2015	Google Translate	Translator and Mapping	Many
Wan, 2008	Yahoo Babel Fish	Translator	Many
Meng et al., 2012; Lu et al., 2011	Parallel Corpus of ISI Chinese-English	Unlabelled Parallel Corpora	English, Chinese
Balahur & Turchi, 2014	Bing Translator	Translator	Many
Bautin et al., 2008; IBM, 2015	Parallel Corpus of 23 EU Languages	Unlabelled Parallel Corpora	Many
Mamou et al., 2018	Intel NLP Architect	NLP tool	Many
Yao et al., 2019	NLKT	NLP tool	Many

2.11 Research Gap

The detailed discussion of this chapter helps to figure out following problems and gaps in the existing research with respect to Bengali knowledge bases, lexicons and datasets, multilingual concept level SA algorithms, preprocessing, feature and concept extraction techniques:

1. **Bengali knowledge bases and lexicons:** From the discussion in Section 2.5 and summaries in Table 2.2, it could be concluded that, existing lexical and knowledge resources are primarily in English, ignored resource-poor language like Bengali and are not concept-based. This gap initiates the possibility to create concept-level knowledge base and polarity lexicon in Bengali.
2. **Multilingual concept level SA algorithms:** It can be concluded from the discussion in Section 2.6 and Table 2.3 - Table 2.6 that the study done so far have mainly used standard algorithms like LSTM, CNN, NB, and SVM, etc., and only a few works have used customized algorithms for dealing with multilingual data. However, those works have ignored the issue of concept-level multilingual SA. These gaps create the necessity of proposing and deploying multilingual concept level SA algorithm. Moreover, existing knowledge bases, lexicons, and datasets are tested using mostly NB, SVM, and LSTM algorithms for finding effectiveness of the resources. This gap creates the requirement of testing new resources of this research with those algorithms.
3. **Student feedback datasets:** From a comprehensive summary of datasets in Table 2.4 and related discussion in Section 2.7, this research has found an insufficient number of student feedback datasets, especially in Bengali. Moreover, there is rare

application of concept-level knowledge bases and polarity lexicon in evaluating SA from student feedback. It is also not evident that any knowledge sources are applicable in the Bengali student feedback datasets.

4. **Feature and concept extraction techniques:** The state-of-the-art literature in Section 2.8 and summary in Table 2.5 presents many issues that need addressing such as i) applying different feature extraction techniques solely or in combinations. ii) Test the impact of feature extraction techniques with respect to different algorithms. iii) Adopt a comparative evaluation of feature and concept extraction techniques. iv) Validating the effect of feature extraction techniques with respect different datasets and lexical resources.
5. **Preprocessing techniques:** The state-of-the-art literature in Section 2.9 and summaries in Table 2.6 and Table 2.7 represents many issues that need proper addressing for performance enhancement such as i) applying different preprocessing techniques solely or in combinations. ii) Test the impact of preprocessing techniques with respect to different algorithms. iii) Validating the effect of preprocessing techniques with respect different datasets and lexical resources.

.

2.12 Summary

The problems figure out or dealt by this research are all about natural language processing (NLP), specially sentiment analysis (SA). Therefore, this chapter presented an in-depth review of NLP and illustrates its relation to SA. Moreover, the chapter tried to overview all NLP levels and tasks and presented the relationship between them. This chapter presented through discussion on background of SA, such as basic concept of SA, the necessity of research on SA, and different SA tasks. The chapter also includes a detailed review of varying levels (such as document-level, sentence-level, comparative, aspect or feature-level, and concept-level) of SA, emphasizing its importance in this research. The concept-level SA was discussed in details with related works as the main concern of this research is concept-level knowledge resources, algorithms, and SA. Finally, a comparative study discusses the differences between the concept-based approach and other approaches.

Another important issue address by this research is SA on Bengali lingual data. Therefore, a thorough evaluation is conducted on Bengali language, problems of Bengali language related to SA, and related work on Bengali SA. Besides, English lingual data is also explored. SA in multiple languages requires transferring the knowledge from resource-rich languages because of resource unavailability in resource-poor langauges; this chapter highlights the resource-rich languages.

This research aims to create a Bengali knowledge base and Bengali polarity lexicon that may resemble or be related to English or Bengali knowledge bases and polarity lexicon to help the researcher in this field in the expansion of research in the

future. Related literature was studied to find the resources and application of those resources. This chapter provides a detailed overview of lexicons and knowledge bases along with work details done so far using different lexicons and knowledge bases, especially SenticNet. This section also highlights the reason for creating BanglaSecticNet (a Bangla knowledgebase) in this thesis.

In order to find out the gap in the existing concept level multilingual SA (MCSA) algorithms, this chapter then presented a thorough discussion on existing algorithms and related works on MCSA. This research aims to create the datasets on students' feedback from social media. Therefore, this chapter presented a comprehensive work review on students' feedback datasets and SA. Pre-processing and feature extraction techniques are crucial steps of data preparation and selection for sentiment classification. A comprehensive review of preprocessing and feature extraction techniques on both lingual SA are presented in this chapter to find out the gap and needful on those techniques.

There are several tools freely accessible to natural language processing tasks. These toolkits could be applied with distinct programming languages. Hence, users can select to work with any toolkits suitable for their NLP applications. This chapter gives a brief idea of such tools. Natural language toolkit (NLTK) is considered the most favorite toolkits due to its features, multi-language and algorithm support, component availability for classification, semantic reasoning, tagging, and tokenization, parsing, and stemming. The next chapter described the detailed methodology of this research along with the resources and algorithm proposed.