

Article

LeSSA: A Unified Framework based on Lexicons and Semi-Supervised Learning Approaches for Textual Sentiment Classification

Jawad Khan  and Young-Koo Lee *

Department of Computer Science and Engineering, Kyung Hee University, Seochon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Korea; jkhanbk1@khu.ac.kr

* Correspondence: yklee@khu.ac.kr

Received: 21 November 2019; Accepted: 13 December 2019; Published: 17 December 2019



Abstract: Sentiment Analysis (SA) is an active research area. SA aims to classify the online unstructured user-generated contents (UUGC) into positive and negative classes. A reliable training data is vital to learn a sentiment classifier for textual sentiment classification, but due to domain heterogeneity, manually construction of reliable labeled sentiment corpora is a laborious and time-consuming task. In the absence of enough labeled data, the alternative usage of sentiment lexicons and semi-supervised learning approaches for sentiment classification have substantially attracted the attention of the research community. However, state-of-the-art techniques for semi-supervised sentiment classification present research challenges expressed in questions like the following. How to effectively utilize the concealed significant information in the unstructured data? How to learn the model while considering the most effective sentiment features? How to remove the noise and redundant features? How to refine the initial training data for initial model learning as the random selection may lead to performance degradation? Besides, mainly existing lexicons have trouble with word coverage, which may ignore key domain-specific sentiment words. Further research is required to improve the sentiment lexicons for textual sentiment classification. In order to address such research issues, in this paper, we propose a novel unified sentiment analysis framework for textual sentiment classification called LeSSA. Our main contributions are threefold. (a) lexicon construction, generating quality and wide coverage sentiment lexicon. (b) training classification models based on a high-quality training dataset generated by using k-mean clustering, active learning, self-learning, and co-training algorithms. (c) classification fusion, whereby the predictions from numerous learners are confluences to determine final sentiment polarity based on majority voting, and (d) practicality, that is, we validate our claim while applying our model on benchmark datasets. The empirical evaluation of multiple domain benchmark datasets demonstrates that the proposed framework outperforms existing semi-supervised learning techniques in terms of classification accuracy.

Keywords: textual sentiment classification; sentiment lexicons; initial training set; semi-supervised learning; self-training; active learning; co-training; classification fusion

1. Introduction

In the last two decades, the web became a primary source, where people look for information and share experiences and perceptions in the form of comments or opinions. The number of internet users is increasing quickly, and the size of the generated data on social network sites (SNSs) is very large. According to a statistical report issued in January 2017 by Hootsuite, 3.8 billion people (50% of the world's population are internet users and 2.8 billion of them are active users [1]. Another report issued in April 2016 specified the strength of most popular SNSs such as Instagram (400 million users),

LinkedIn (450 million Users) and Facebook (1.71 billion monthly active users) [2]. Similarly, monthly active users of Twitter are above 288 million, with over 500 million tweets generated every day [3].

These statistics show the fame and importance of SNSs. Resultantly, such SNSs platforms generate a gigantic amount of Unstructured User Generated Contents (UUGC). Such UUGC contains precious information and can play an important role in decision making that can be advantageous for both consumers and organizations. It is almost difficult for a human to check and categorize such a huge number of documents manually. To turn UUGC into precious information, it must be queried, analyzed and visualized. Sentiment Analysis (SA), using natural language processing and machine learning tools and techniques, is the candidate solution for analyzing such useful information automatically. SA, also known as opinion mining, is the process that extracts and analyzes people's opinions, sentiments, attitudes, judgments and emotions regarding entities, events and their attributes [4]. The field of SA has been thoroughly studied in the literature [4–6].

The objective of SA is to classify UUGC into either positive or negative classes. Most of the methods for SA work on Supervised Machine Learning (SML) [1,7–9]. The SML methods need a large amount of labeled data to train a classifier (such as Support Vector Machine (SVM) [10] or Naive Bayes (NB) [11] for sentiment classification. However, there is a shortage of labeled data which could be used to train a classifier for multiple domains. Besides, it is also challenging and time-consuming to label a large amount of data manually for various domains, as they need skilled human annotators.

In the absence of enough labeled data, the alternative usage of lexicon scoring (LS) and semi-supervised learning (SSL) approaches for sentiment classification have greatly attracted the attention of scientists. The LS addresses this problem by using existing general sentiment lexicons to calculate sentiment scores of review texts. Based on calculated sentiment scores, the review text is categorized into either positive or negative class. A sentiment lexicon is a list of words or phrases that express sentiments. Sentiment lexicon plays a vital role in textual sentiment analysis. Sentiment lexicons have been widely employed for textual sentiment classification due to the benefit that they do not need labeled data for training [12,13]. Existing sentiment lexicons also have been utilized for learning an initial classifier in order to generate a domain-specific sentiment feature for sentiment analysis [14]. A combination of lexicon and learning-based approaches for textual sentiment classification have been proposed [15,16]. In the literature, several sentiment lexicons with different sizes and formats have been created for sentiment analysis, e.g., AFFIN [17], OL [18], SO-CAL [19], WordNet-Affect [20], GI [21], SentiSense [22], MPQA Subjectivity Lexicon [23], NRC Hashtag Sentiment Lexicon [24], SenticNet5 [25] and SentiWordNet [26]. Despite wide usage for assessing sentiment in social media contexts, existing sentiment lexicons have a concern with coverage, which may ignore important domain-specific sentiment words. Besides, building sentiment lexicon manually is time-consuming and labor-intensive. In order to improve the existing general sentiment lexicons, we integrate them leveraging linguistic and semantic knowledge to expand the word coverage of individual sentiment lexicon.

Semi-supervised learning (SSL) is a machine learning technique that utilizes a large amount of unlabeled data along with a small amount of labeled data to learn classifiers and achieve a better performance [27]. In the literature, mostly learning models uses semi-supervised learning techniques, such as self-training, co-training, active learning, and a combination of these has been mostly used for textual sentiment classification [28–33].

Self-training is the most well-known and simple SSL algorithm. In self-training, first, a classifier is trained on a small number of labeled data to predict the label for the unlabeled data. Then, high confidence level unlabeled examples with predicted labels are selected to add to the training data. Further, the classifier is retrained, and the process is recurrent until a predefined threshold is satisfied [34]. Wang et al. [34] addressed a semi-supervised learning approach, self-training, for sentence subjectivity classification. He et al. [14] proposed a self-training method for automatically domain-specific features acquisition by using the pseudo-labeled examples with high confidence. Qiu et al. [28] presented the SELC Model (Self-supervised, Lexicon-based and Corpus-based model) for sentiment classification.

Co-training is another semi-supervised learning method presented by Blum et al. [35]. In co-training, two classifiers operate on two different and sufficient views for the training set expansion. In the standard co-training algorithm, each classifier classifies the unlabeled data and selects the most confident classified examples according to a threshold and adds into the training set of another classifier. The training set of each classifier is enriched, this process continues for several iterations until the condition criteria is satisfied [30,36]. Xia et al. [31] proposed co-training for semi-supervised sentiment classification based on dual view bag of words representation. Yang et al. [37] proposed a semi-supervised model for sentiment classification based on the idea of combining lexicon-based learning and corpus-based learning in a unified co-training framework.

Active learning, a type of semi-supervised learning, is used to actively select the unlabeled examples to be annotated by an oracle (e.g., a human annotator) with minimum annotation effort [38,39]. The goal of active learning is to improve the classification accuracy with fewer labeled instances [40]. Zhou et al. [41] proposed a novel Active Deep Network (ADC), to address the sentiment classification problem with active learning using a small number of labeled data. Hajmohammaadi et al. [33] proposed a novel learning model based on the combination of active learning and self-training for cross-lingual sentiment classification. They also designed and implemented some other baseline models and compared it with their proposed model to show the actual effects of different parts of their proposed model.

In semi-supervised learning, the initial training set is selected randomly from the whole unlabeled data. Random selection of the initial training set does not guarantee to be the most representative subset of the entire unlabeled data. Besides, mainly existing semi-supervised learning methods, which utilize unlabeled data, contain abundant and redundant noisy features that may affect classification accuracy. The existing semi-supervised techniques cannot handle these noisy features effectively. Moreover, existing approaches do not focus effectively on the sentiment features of the unstructured data.

In this paper, we propose a novel unified sentiment analysis framework for textual sentiment classification called LeSSA. In LeSSA, we utilize high-quality wide coverage sentiment lexicons along with other machine learning techniques, such as k-mean clustering, self-training, co-training, active learning for training sentiment learners. The full-trained sentiment learners predict the sentiment polarity class for the test review document. We employ classification fusion using majority voting to predict the final sentiment label for the unseen textual reviews.

In LeSSA, we aim to address the issues mentioned above under the umbrella of the two-step process (1) integrating several sentiment lexicons leveraging linguistic and semantic knowledge to induce high-quality wide coverage sentiment lexicon and, (2) training classification models based on high-quality training datasets for sentiment classification.

In the former case, our objective is to induce a high-quality wide coverage sentiment lexicon. The existing general sentiment lexicons have some shortcomings. (a) Many of the existing general sentiment lexicons do not cover domain sentiment words, which may ignore important domain-specific sentiment words. (b) Mostly words that are not considered sentiment words in the existing general sentiment lexicon can express sentiment in a particular domain. (c) Most of the existing sentiment lexicons cannot handle effectively context-dependent sentiment words. In order to solve these shortcomings in the existing general sentiment lexicons, we integrate various sentiment lexicons leveraging linguistic and semantic knowledge to expand the sentiment lexicons for better sentiment classification. In the latter case, we competently select high-quality data for training to minimize the data labeling exertion for better sentiment classification. In order to select the high-quality training data, first, we generate the initial training set based on k-means clustering then, we train first classifier on the initial training set and apply on unlabeled training dataset to label them based on some strategy (e.g., uncertainty sampling, high confident level). This process continues for several iterations until the termination condition is satisfied, and finally, the fully learned classifier is trained on the labeled training data. While training data we remove the noisy and redundant features and extract and select the most effective sentiment features for better sentiment classifier learning. Solutions to the mentioned

issues result in lexicon-based and semi-supervised learning-based classification models. In detail, the final output predictions of all the designed and implemented classification models on test data are combined using majority voting to make a more accurate sentiment classification.

The experimental results show that our proposed approach is suitable in predicting the true class that can improve the sentiment classification performance efficiently and outperforms baseline methods.

The key contributions of this work are as follows:

- A novel, unified sentiment analysis framework, known as LeSSA, has been proposed for textual sentiment classification.
- The confluence of various sentiment dictionaries has been employed to generate a high-quality wide coverage sentiment lexicon.
- We identify and extract context-aware domain-specific sentiment words by leveraging linguistic rules and semantic knowledge.
- We exploit the bootstrapping method to obtain the most reliable training datasets employing three well-known semi-supervised learning techniques, such as self-training, co-training, and active learning in order to select the most useful instances to enhance and update the initial training set.
- We extract sentiment bearing features and reduce the feature space while exploiting MRMR feature selection criteria.
- We design and implement multiple sentiment learners.
- The results obtained using classification fusion show higher accuracy comparable with base sentiment learners.
- We evaluate the performance of various designed classification models in order to show their effect on the proposed unified framework.
- The experiments on two well-known benchmark datasets validate that the proposed method improves the sentiment classification in terms of accuracy.

All the acronyms used in this paper are listed in Table 1. The rest of the paper is organized as follows: Section 2 is about related work on SML, SSL, and Sentiment Lexicons for textual sentiment classification. The proposed unified SA framework (LeSSA) is briefly described in Section 3. The technical and implementation details of LeSSA are discussed in Section 4. Experimental results, evaluation and discussion are presented in Section 5. Finally, in Section 6, we conclude the paper by signaling future work.

Table 1. List of acronyms being used in this paper.

Name	Acronym
Sentiment Analysis	SA
Support Vector Machine	SVM
Naïve Bayes	NB
Supervised Machine Learning	SML
Social Networks Sites	SNSs
Semi-Supervised Learning	SSL
Unstructured User-Generated Contents	UUGC
Text Pre-Processing	TPP
Sentiment Features Extraction and Selection	SFES
Multi-Model Sentiment Learning Layer	MMSLL
Wide Coverage Sentiment Lexicon-Based Sentiment Learner	WCSL-SL
Bag of Words	BOW
Minimum Redundancy Maximum Relevance	MRMR
Self-Training Based Sentiment Learner	ST-SL
Part-of-Speech Tag	POS
Feature Engineering	FE
Multi-Model Sentiment Learning	MMSL
Sentiment Classification	SC

Table 1. Cont.

High Ranking Pseudo-Labeled Based Sentiment Learner	HRPL-SL
Active-Self-Training Based Sentiment Learner	AST-SL
Active-Self Co-Training-Based Sentiment Learner	ASCT-SL
Classification Fusion	CF
Lexicon Scoring	LS
Sampling by Clustering	SBC

2. Related Work

Recently several SML, SSL and lexicon-based approaches for textual SA have been proposed [7,12,34,42,43].

The SML approach relies on the presence of labeled training data to learn a classifier like SVM, which is subsequently applied to the test data for classification. An SML approach for movie review sentiment classification was proposed by pang et al. [7]. They applied three SML algorithms (NB, Maximum Entropy and SVM) and achieved a high accuracy of 82.9% using unigrams as features. Another SML approach was presented by Saleh et al. [44], in which they applied SVM for testing diverse domains of datasets and used various weighting schemes (TFIDF, BO, TO) while adopting n-gram based approach. They obtained a high accuracy level of 91.5% using unigram features with TFIDF scheme. SML approach for sentiment classification performs well when enough amount of label data is available. However, it is difficult to get a large amount of labeled data manually for different domain. This deficiency brought an interest to lexicon based and SSL approaches for SA.

Various sentiment lexicons with prior polarity have been created recently for document classification. Sentiment lexicon assists as a vital tool for making a sentiment classification system [19]. Sentiment lexicon is a list of opinioned words or phrases. In some cases, a value is assigned to each word in the sentiment lexicon to differentiate the semantic orientation level of the word. Some of the most well-known and largely used general sentiment lexicons are AFFIN, OL, MPQA, SO-CAL, Subjectivity lexicon, NRC Hashtag Sentiment Lexicon, etc. According to the existing literature, the general sentiment lexicons have been generated manually, semi-automatically or fully automatically [45]. The manual approach of sentiment lexicon generation is very difficult due to high effort and expert time requirements. The manual approach for sentiment lexicon development is usually combined with automatic approaches. Hu et al. [18] have developed a manually compiled sentiment lexicon for sentiment analysis.

Generally, there are two popular methods for sentiment lexicon construction, dictionary-based method, and corpus-based method [4]. The dictionary-based method utilizes an initial set of seed sentiment words, which are collected and labeled manually. This set is then expanding by looking synonyms and antonyms in thesauri such as WordNet [46]. The corpus-based approach utilizes seed sentiment words to discover new domain sentiment words and their orientation [4]. Hazivassiloglou et al. [47] exploited some seed sentiment words for discovering new domain sentiment words in the corpus. Another corpus-based method relies on the word co-occurrence in the corpus [48]. Most of the existing general sentiment lexicons for sentiment classification is insufficient due to the limited word coverage that may neglect important sentiment words, which are not present in the existing lexicons.

SSL approach can take benefit of utilizing a small amount of labeled training data, together with a large amount of unlabeled data. There is a reason for using unlabeled data to boost the learning performance of a learner in a case when only a small-scale of labeled data exists [27]. In the SSL approach, an initial set is used for training the first classifier. The selection of an initial set for training first classifier is significant. In previous studies, the selection of an initial set from the whole unlabeled data is made by random sampling. However, due to the complex distribution of data, random sampling cannot ensure selection of the most representative instances. Therefore, other techniques like sampling by clustering and sentiment lexicon are used for initial training set generation [14,49,50].

In the literature, many SSL techniques such as self-training, co-training, active learning and graph-based and topic-based semi-supervised learning have been used for classification problems [27].

In self-training, the most confident unlabeled examples, together with their predicted label with a given threshold are selected to add to the training set. Wang et al. [34] addressed self-training, for sentence subjectivity classification. They adapted Value Difference Metric (VDM) as the selection metric in self-training. According to their experimental results, self-training with NBTree and VDM outperformed self-training with other combinations of classifiers selection metrics. Qiu et al. [28] presented the SELC Model for sentiment classification, which successfully integrates a corpus-based model with a lexicon-based approach.

Active learning is an encouraging process for SSL classification, which reduces the data labeling cost. Zhou et al. [41] proposed a novel Active Deep Network (ADN) method for textual sentiment classification. Li et al. [51] proposed active learning for imbalanced sentiment classification. They utilized two complementary classifiers where the first classifier was used to obtain the most certain samples and the second to obtain the most uncertain samples for manual annotation. Hajmohammadi et al. [33] proposed a novel learning model in which they combine active learning with self-learning to decrease the effort of human labeling and improve the classification performance in cross-lingual sentiment classification. In their model, firstly, unlabeled data translated from the target language into the source language, then translated data are increased into initial training data in the source language utilizing active learning and self-training. They also considered the density measure of unlabeled examples in active learning to avoid the selection of outlier examples from unlabeled data.

Co-training [27] is a bootstrapping method that assumes that feature space can be divided into two different redundant and sufficient sets (views). Initially, two different classifiers are trained with the labeled data on the two different views, respectively. Each classifier then applied to the unlabeled data, and the most confident predictions of each trained classifier on the unlabeled data are added into the training set of another classifier for label data expansion. Xia et al. [31] proposed a dual view co-training algorithm based on dual view bag-of-words (BOW) representation for semi-supervised sentiment classification. In dual view BOW, they automatically constructed antonymous reviews by a pair of bag-of-words with opposite views. They made use of the original and antonymous views pair-wisely in training, bootstrapping and testing processes, all based on a joint observation of two views. Yang et al. [37] presented an LCCT (Lexicon-based and Corpus-based, Co-Training) model for semi-supervised sentiment classification, combining the idea of lexicon-based learning and corpus-based learning in a unified co-training framework. Li et al. [29] proposed a cooperative semi-supervised learning approach based on the hybrid mechanism of active learning and self-learning for textual sentiment classification.

Recently, deep learning-based aspect extraction [52], attention-based LSTM [53] and capsule networks [54,55] have also been widely used for sentiment analysis and challenging NLP applications that yield state-of-the-art prediction results. In this work, our focus is on the design of a unified sentiment analysis framework for multi-domain textual sentiment classification.

In literature, for SA, SSL has been used by many researchers while gaining performance [29–31,41,51]. However, most of them did not notice how to utilize effectively concealed significant information, which gives more sentiment clue as compare to other words. Likewise, they did not consider the most appropriate sentiment features. The presence of redundant and noisy features in the text data is also ignored. The initial training set for first model training is randomly selected, which may affect the performance of sentiment classification. Besides, different classification models based on lexicon and semi-supervised learning techniques have been proposed, but it is hard to judge which model performs best in common.

In order to address the above issues in state-of-the-art work, in this paper, we propose a novel unified SA framework based on high-quality wide coverage sentiment lexicons and semi-supervised learning techniques in conjunction with classification fusion for textual sentiment classification.

3. Proposed LeSSA Framework

Formally, we describe the LeSSA framework in this section, the implementation and technical details in the next section. LeSSA is a novel unified SA framework, which has been designed with the intentions for multi-domain sentiment classification in a case when the labeled data is scarce or not available. LeSSA is also applicable for heterogeneous domain sentiment classification. The core architecture of the LeSSA is illustrated in Figure 1. LeSSA is composed of three main layers: Feature Engineering (FE), Multi-Model Sentiment Learning (MMSL), and Sentiment Classification (SC).

FE layer is responsible for generating effective sentiment features from unstructured data. The construction of suitable feature vector creation from the unstructured textual data is a significant task for well-learning performance in sentiment classification. In the literature, various feature generation techniques such as BOW, n-grams, POS tagged features and semantic features have been presented [7,9,56–59]. In this paper, we extract and select the most appropriate features from review texts, which are adjectives, adverbs, verbs and nouns for learning sentiment classifiers.

MMSL Layer, uses five different sentiment learners, such as wide coverage sentiment lexicon-based sentiment learner (WCSL-SL), high ranking pseudo-labeled based sentiment learner (HRPL-SL), Self-Training based Sentiment Learner (ST-SL), Active-Self-Training based Sentiment Learner (AST-SL) and Active-Self-Co-training based Sentiment Learner (ASCT-SL). For the first two sentiment learners, we utilize the induced WCSL for learning, while for the remaining learners, we use SSL techniques with the K-means clustering algorithm to generate an initial training set for first classifier learning. In the SC layer, the final fully trained sentiment learners based on WCSL-SL, HRPL-SL, ST-SL, AST-SL, and ASCT-SL are applied to the test data for sentiment polarity prediction. Finally, classification fusion is presented that determines the final sentiment polarity based on majoring voting. In the proposed framework diagram, the solid arrow line represents the model training workflow, while the dotted arrow line represents the offline workflow for real-time prediction of SA.

4. LeSSA—Implementation and Technical Details

LeSSA is a multi-domain SA framework and consists of three layers. The technical details of each layer are explained in the following subsequent sections.

4.1. Feature Engineering Layer

The construction of appropriate feature vector creation from the unstructured textual data is a significant task for well-learning performance in sentiment classification. For this purpose, we develop the FE layer, which consists of two components: Text Pre-processing (TPP) and Sentiment Features Extraction and Selection (SFES). The TPP is employed to generate appropriate feature vector for better learning performance in sentiment classification. The TPP component consists of four types of modules. These modules are sentence parser, lower case, noise remover, tokenizer.

First, the review dataset is loaded, then the review text is split into words by using sentence parser and tokenizer, respectively. In the next step, the noise remover module is invoked to remove noise from the text if any, and the same are then transformed to lower case by using case transformer module.

The TPP then inputs the data to the SFES sub-module where the POS Tagger assigns POS tags with the purpose of determining and extracting the likely words such as adjectives, adverbs, verbs and nouns. After that, these words are searched in the induced wide-coverage sentiment lexicon to select the effective sentiment words and filter-out non-sentiment words. Next, the feature vector is created and selected using term frequency-inverse document frequency scheme (TF-IDF) and pruned words below 'absolute = 3' and above 'absolute = 3000' from the feature vector space.

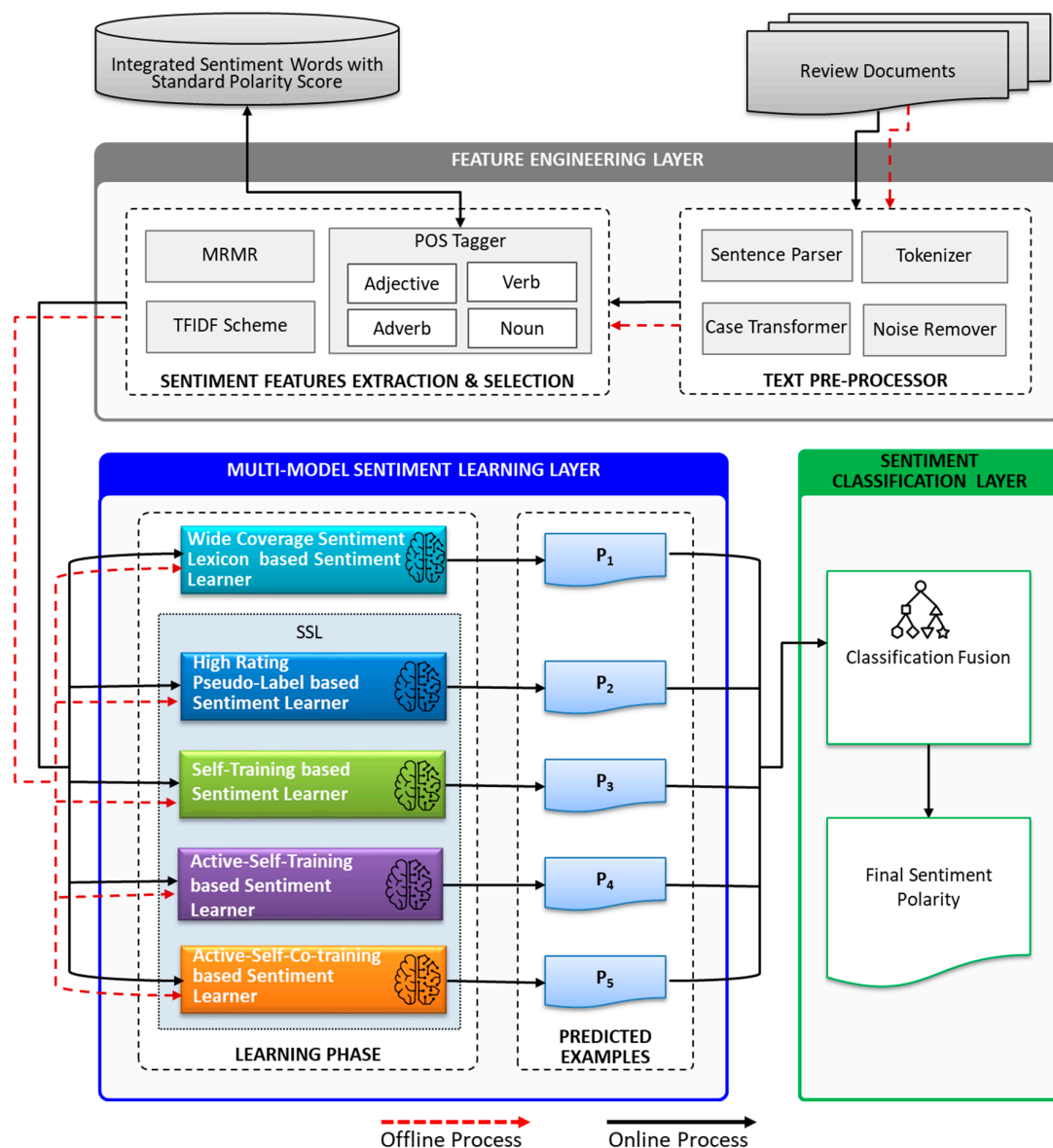


Figure 1. The architecture of the proposed framework.

Further, we selected high-quality top rank sentiment features using the Minimum Redundancy Maximum Relevance (MRMR) [60] feature selection approach. The extraction and selection criteria for high-quality sentiment features have a great impact on the performance of supervised and semi-supervised textual sentiment classification [1,61]. We employed the aforementioned text-mining techniques to all the proposed sentiment learners.

4.2. Multi-Model Sentiment Learning Layer (MMSLL)

MMSLL is composed of five types of sentiment learners, i.e., WCSL-SL, HRPL-SL, ST-SL, AST-SL, ASCT-SL. The subsequent section explains the sentiment model learning techniques for textual sentiment classification in more detail.

4.2.1. Wide Coverage Sentiment Lexicon Based Sentiment Learner

Sentiment lexicon plays a vital role in the review document sentiment classification. A sentiment lexicon is a list of words that are used to show positive or negative sentiments [4]. In the absence of abundant training data, the alternative approach of LS has been largely employed by many researchers

in the domain of textual sentiment classification. In this approach, sentiment words in review documents are matched against sentiment lexicons, and a sentiment value is assigned to matched sentiment words. The overall sentiment orientation of a review document is then determined by using a formula (as shown in Equation (1)). In detail, the sentiment values of matched sentiment words in the review document are aggregated to determine the sentiment orientation for the review document. The architecture of induced WCSL-SL for the review document classification is shown in Figure 2. The detailed process of inducing WCSL-SL along with linguistic rules and semantic knowledge for review document sentiment classification is given in the subsection below.

$$\text{Review Sentiment Score} = \text{Sum of positive sentiment scores} - \text{Sum of negative sentiment scores.} \quad (1)$$

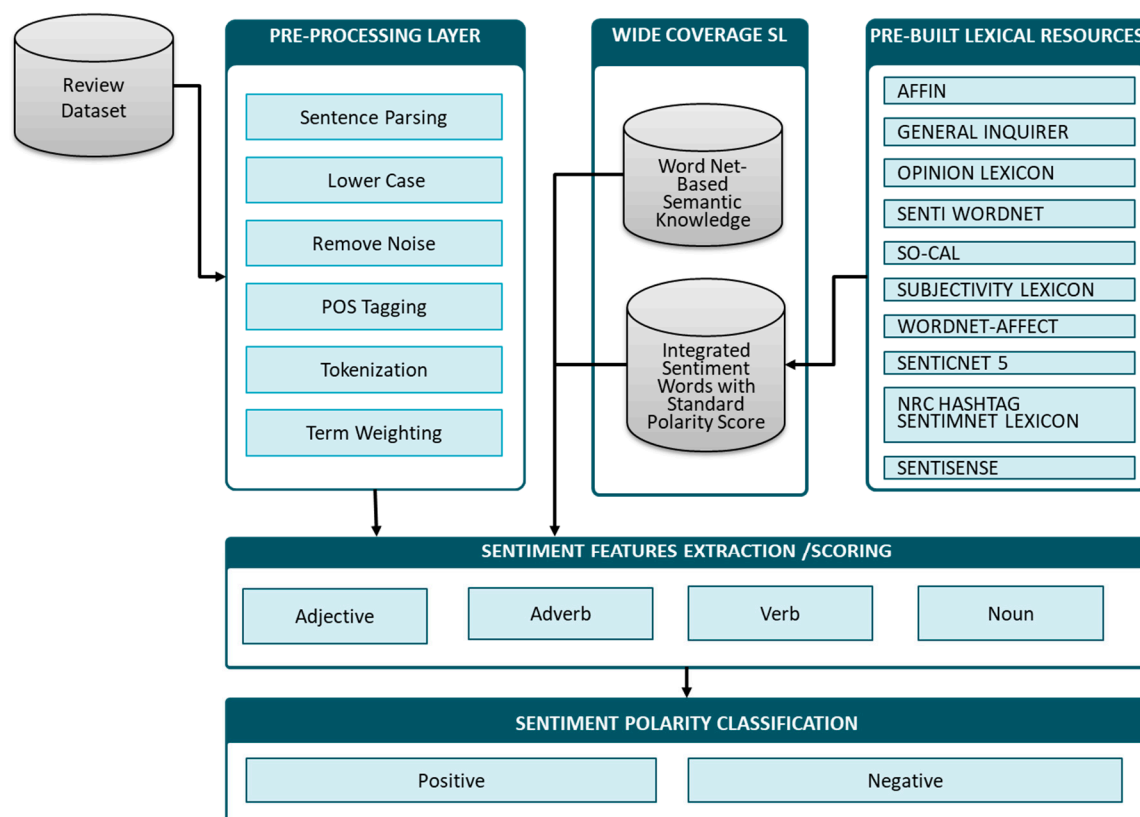


Figure 2. The architecture of WCSL based sentiment classification.

Sentiment Lexicon Integration and Standardization

In the literature, several sentiment lexicons such as SentiWordNet, Wordnet-Affect, Micro-WNOP with different sizes and formats have been constructed to classify review documents into positive and negative classes. However, there is no optimum universal/general sentiment lexicon because the semantic orientation of words is sensitive to a particular domain. Besides, words that are not considered sentiment words in the existing sentiment lexicon can express sentiment in a particular domain.

Furthermore, the existing sentiment lexicon cannot handle effectively context-dependent sentiment words. Many of the existing sentiment lexicons are beset by having limited words, which are insufficient to calculate the sentiment score of a domain-specific sentiment word. We present an integrated lexicon-based approach utilizing existing sentiment lexicons with linguistic rules and semantic knowledge. Our purpose is to integrate multiple sentiment lexicons to increase the word coverage of sentiment lexicon for sentiment classification.

Moreover, we identify and extract context-aware domain-specific sentiment words and determine their sentiment scores by leveraging linguistic rules and semantic knowledge. Here we integrate ten state-of-the-art sentiment lexicons intending to improve the limited word coverage of the individual sentiment lexicon. The summary of existing sentiment lexicons is given in Table 2, where the size and format of the sentiment lexicons are different from each other. Some sentiment lexicons contained numeric scores with different ranges, while other assigned positive, negative and neutral classes to sentiment words. Furthermore, some sentiment words were categorized into a variety of emotions such as joy, happiness and sadness. As these sentiment lexicons have distinct forms, it is important to convert them into a standard form. We standardize the sentiment lexicons to have one of three different values, +1, 0, −1. While for sentiment lexicon integration, we take the average of the sentiment values of the overlapping words, which produces a huge sentiment lexicon with more sentiment words.

The detailed process of the sentiment lexicon standardization is explained below.

- **AFINN:** AFINN is a list of English words with an integer between −5 and 5. We scaled the sentiment scores from [−5,+5] to [−1,+1] using minmax scale [62].
- **General Inquirer (GI):** The sentiment words in GI have been categorized into more than 180 classes. We considered positive, strong, pleasure, virtue, pstv, active, complete and yes as positive categories and assign +1 sentiment value to every sentiment word in the category. Furthermore, we examined negative, ngtv, fail, weak, decrease, pain, no, negate as negative categories and assigned −1 sentiment score to each word in them.
- **Opinion Lexicon (OL):** The opinion-lexicon is composed of positive and negative sentiment words. We assign +1 and −1 score to positive and negative sentiment words, respectively.
- **SentiWordNet:** Sentiwordnet is based on wordnet synsets, and each synset s is associated with positive and negative numerical scores within the range of [0,1]. We subtracted a term negative score from its positive score. We consider a term as positive if its positive sentiment score is bigger than its negative sentiment score; otherwise, it is considered as negative if its positive sentiment score was less than its negative sentiment score. If the term positive score and negative score is equal, then the term is considered as neutral. We assigned a numerical score of +1 to positive words, −1 to negative words and 0 to neutral words.
- **SO-CAL:** The sentiment scores of words are scaled from [−5,5] to [−1,+1] using the aforementioned technique
- **Subjectivity Lexicon:** Words belonging to positive, negative, both and neutral categories are assigned numeric scores +1, −1, 0 and 0, correspondingly.
- **WordNet-Affect:** Words have been grouped into different emotion categories, namely, positive, negative, ambiguous and neutral emotion. The sentiment score of +1, −1, 0 and 0, are assigned to these categories, respectively.
- **NRC Hashtag Sentiment Lexicon:** We normalized the words sentiment score from [−7,+7] to [−1,+1]
- **SenticNet5:** A sentiment score to each sentiment word has been assigned within the range of [−1,1]. We considered the sentiment score of each sentiment word.
- **SentiSense:** SentiSense comprises of synsets with a set of 14 emotional categories. We assigned score +1 to love, like, joy and hope categories and −1 to sadness, hate, despair and disgust categories, while 0 score is assigned to anticipation, surprise and ambiguous categories.

Linguistic Rules

Linguistic rules assist in identifying context-aware domain-specific sentiment words in a better way. Researchers [4] have pointed out that the usage of linguistic rules could positively affect the performance of sentiment classification. Motivated by [47], we present how to identify and extract context-aware domain-specific sentiment words using the following linguistic rules. We split the review documents into sentences and identify all those sentences from the review documents that

comprise the below conjunction clauses. We employ the Stanford Parser to parse each sentence and extract context-aware domain-specific sentiment words. Each sentence was parsed into the syntactic tree during tokenization and when applicable all the sentences that comprise conjunctions are extracted for domain-specific sentiment feature extraction.

- **Rule 1—similar sentiment:** If two words or clauses are linked by “and” conjunction in a similar sentence, then we can infer that they have a similar sentiment orientation. For example, “This phone is nice and cheap”, both “nice” and “cheap” express positive sentiment orientation. Based on this, if we are not sure about the sentiment orientation of the word “cheap”, but we can tell that the sentiment orientation of the word “nice” is positive in general sentiment lexicon, then we can deduce that “cheap” is also a positive word.
- **Rule 2—opposite sentiment:** It comprises some words like, “but”, “while”, “despite”, “unless”, “however”, “although”, “nevertheless”. If two words are connected by “but” conjunction, then we can assume that they both have opposite sentiment orientations. For example, “This laptop is nice but expensive”, both “nice” and “expensive” are sentiment words. Based on this rule, if we are not sure that the word after “but” conjunction is positive or negative, but we distinguish that the word before “but” conjunction is a sentiment word, then we can refer that the word after “but” conjunction is also a sentiment word.
- **Rule 3—negation handling:** The negation word or phrase generally alters the sentiment orientation (polarity) of the sentiment word in the sentence or clause. Negations are typical words such as, no, not, neither and do not and also pattern-based negations like “stop” vb-ing and “quit vb-ing”. We used a list of negation words to determine negation expression in the review text. We adopted a simple method for negation handling. If a negation word, e.g., “not” appears at the left side of sentiment word then, in this case, the sentiment orientation of the sentiment word will be changed. For example, “This movie is not good”, the word negation “not” alters the sentiment polarity of word “good”. Negation handling is applied at tokenization time. There are also negation words in phrases such as, “not only”, “not just”, “not all” which do not mean negation we do not consider them as a negation expression pattern.

Semantic Knowledge

The word coverage limit in the sentiment lexicons is the main challenge. If a sentiment word is not included in the existing sentiment lexicons, then it is ignored, which can affect the sentiment score of the review text. In order to address this problem, we employ semantic knowledge using WordNet to find the synonyms of the word and obtain its sentiment orientation.

Document Sentiment Score

During document sentiment score, each review text is tokenized, and POS tag is assigned to extract the likely words such as adjectives, adverbs, verbs and nouns. The sentiment score of each word is searched in the integrated wide coverage sentiment lexicon and using semantic knowledge. For each word in the review text, a sentiment score +1 and −1 is assigned to positive and negative words, respectively. The review with a score above zero is classified as positive, and the review with a score of −1 and below is classified as negative. Linguistic rules for dealing with context-dependent sentiment words are applied, and negated words are handled as mentioned before. We use the document sentiment score formula given in equation 1 to calculate the review document sentiment score. The detailed procedure for document sentiment score calculation is given in Algorithm 1.

Table 2. Pre-build lexical resources (sentiment lexicons).

Lexical Resources (Sentiment Lexicon)	Size	Category	Sentiment Score
AFFIN	Words: 2477	N/A	[−5, +5]
GI	Words: 11,789	Positive, negative, pstv, ngtv, strong, weak etc.	N/A
OL	Words: 6786	Positive, negative	N/A
SentiWordNet	117,659 Synsets, Words: 155,287	Positive, negative and objective	[0,1]
SO-CAL	Words: 6306	N/A	[−5, +5]
Subjectivity Lexicons	Words: 8221	Positive, negative and neutral	N/A
WordNet-Affect	2872 Synsets, Words: 4785	Positive, negative, ambiguous, neutral	N/A
NRC Hashtag Sentiment lexicon	Words: 54,129	Positive, negative	[−7, +7]
SenticNet5	Concepts: 100,000	Positive, negative	[−1, +1]
SentiSense	2190 Synsets, Words: 5496	Like, love, hope, hate, etc.	N/A

Algorithm 1. Document sentiment score calculation

inputs: $D = \{s_1, s_2, \dots, s_n\}$; /* Review documents with n sentences */
 $S_i = \{w_1, w_2, \dots, w_n\}$; /* Occurrence of n words in a sentence I */
 $S_w = \{adjective, adverb, verb, noun\}$; /* Denotes sentiment words */
output: $D_O \{ssw_1, ssw_2, \dots, ssw_n\}$; /* Denotes the n words sentiment score for review document classification */

```

1  Begin
2  foreach  $S_i$  in  $D$  {
3     $PS_i \leftarrow \text{POSTagger}(S_i)$  /* Assign POS Tag to each sentence */
4    foreach  $W_i$  in  $PS_i$  {
56     if  $W_i \in S_w$  {
7       //WCSL: Wide coverage sentiment lexicon
8       if  $W_i$  exists in WCSL {
9          $SS_i W_i \leftarrow \text{Search } W_i \text{ in WCSL}$ 
10        }
11       else {
12          $Syn\_W_i \leftarrow \text{fetch synonyms of } W_i \text{ from Wordnet}$ 
13         // Sum score of all synonyms
14          $SS_i W_i \leftarrow \text{Sum WCSL} (Syn\_PS_i W_i)$ 
15       }
16     }
17   }
18   Negation handling
19   Looking conjunctions for domain-specific words extractions
20   Determine the sentiment score of the  $D$  using Equation (1)
21 End

```

4.2.2. Semi-Supervised Learning for Textual Sentiment Classification

Semi-supervised learning is a popular technique that utilizes large amounts of unlabeled data together with a small amount of labeled data to make better classifiers [27]. Semi-supervised learning needs low human effort and grants high accuracy. Self-training, co-training and many other methods are used for semi-supervised learning. In this work, we investigated four different semi-supervised learning techniques for textual sentiment classification, which are mentioned in the below subsections.

4.2.3. High Ranking Pseudo-Label-Based Sentiment Learner

In this approach, we utilize the unlabeled training review documents to generate pseudo label examples and it is a two-phase bootstrapping process. First, the quality wide coverage sentiment lexicon is used to calculate sentiment scores for known review documents. Then the scored review documents are classified into positive and negative classes by the sentiment scores. After, a portion of the classified review documents as a reliable pseudo label example is selected according to high absolute

sentiment score. Further, high ranking pseudo-label examples based on high absolute sentiment scores are selected for training, and a supervised sentiment learner such as SVM is trained on these examples. Finally, the learning classifier is applied to the unseen review documents, and the sentiment class is determined for them.

While learning the sentiment classifier, we utilized the preprocessed textual reviews documents using FE layer. The sentiment score of each review document is calculated using the review sentiment score formula mentioned in Equation (1). The reviews are ranked in decreasing order based on sentiment score, and the top positive and negative review documents are selected as pseudo labeled examples. The workflow for HRPL is shown in Figure 3.

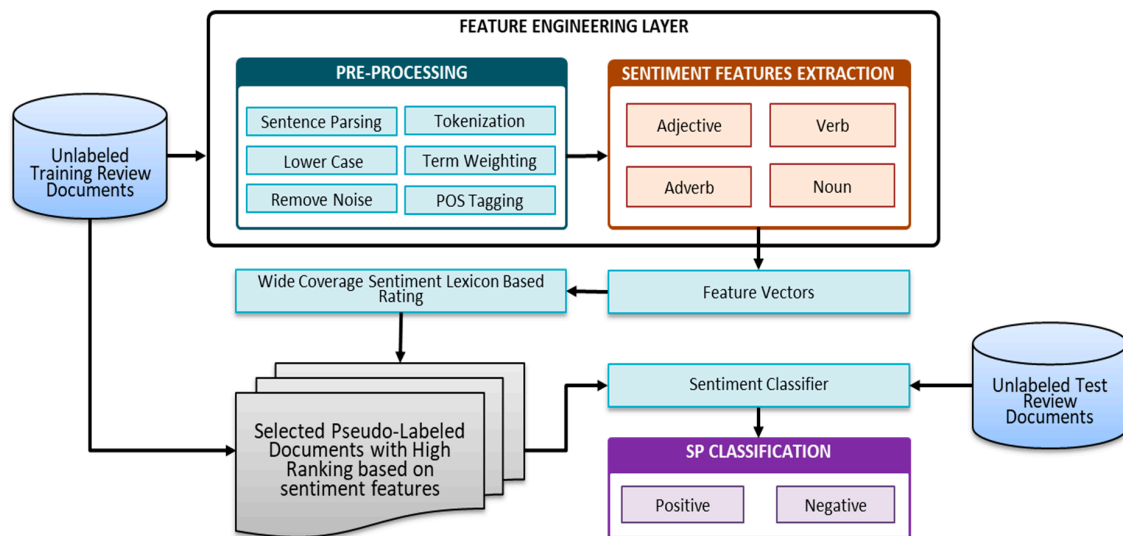


Figure 3. Sentiment classification based on HRPL documents.

4.2.4. Initial Training Set Selection for Self-Training, Active Learning and Co-Training

In semi-supervised learning, the selection of an initial training set for first classifier learning is very important. This initial training set may affect the performance of semi-supervised learning. In previous studies on semi-supervised learning, the initial training set is usually induced at random. However, due to the complexity of unstructured unlabeled data distribution, it is hard to say that the randomly selected subset for initial training set will be the most representative samples to the unstructured unlabeled data. There have been many works on how to select the initial training set such as employing clustering and sentiment lexicons [14,49,50]. Motivated by the method proposed in [40], sampling by clustering (SBC) scheme is adopted for initial training set generation. In this scheme, the unlabeled textual data is divided into known number of k clusters utilizing k -means clustering algorithm. The k -means algorithm groups the unlabeled examples into k -clusters of similar ones. In order to obtain the most representative initial training set, first we cluster the documents into k -clusters, and then, a single document from each cluster closest to the respective centroids based on cosine similarity is selected, which is considered the most representative sample from each cluster. The selected documents are annotated, and an initial classifier is trained on them to predict the labels for unlabeled data. From the unlabeled data the most reliable examples with their predicted labels are selected and added into the initial training set based on some strategy (high confidence, high uncertainty).

4.2.5. Self-Training-Based Sentiment Learner

Self-training, also known as self-learning, is one of the well-known semi-supervised learning methods. In self-training, first, a classifier is trained using a small amount of initial labeled data. Then the classifier is employed to classify the unlabeled data. After that, the high confident unlabeled examples

with their predicted labels are selected to add to the training data. The classifier is retrained, and this process is repeated many times to label unlabeled data and enrich the training data. In this paper, first, the classifier is trained using the initial set which was obtained with SBC scheme and annotated manually. Then the classifier is applied to the unlabeled text data for class prediction; thereafter, high-confidence level classified examples by the classifier, according to a specific threshold are selected and added to the labeled training text set. Furthermore, the classifier is retrained based on the enriched labeled training text set and predicts labels for the leftover unlabeled texts. This process is continued for a certain number of iterations until a termination criterion is satisfied. After a complete training process, the test data is given to the final trained classifier for sentiment classification. The general procedure used for self-training is described in Algorithm 2. The workflow for self-training is shown in Figure 4.

Algorithm 2. Self-training (ST)-based learning

inputs: $X = \{x_1, x_2, \dots, x_m\}$; /*unlabeled data of m instances*/

$Y = \{y_1, y_2, \dots, y_m\}$; /*labeled data of m instances*/

S /*classifier*/

λ /*Threshold*/

output: /*labeled data, $|Y_0| \leq |Y| + |X|$, leq: less than or equal to */

$Y_0 = \{y_1, y_2, \dots, y_n\}$

```

1  Begin
2    WHILE X contain instances {
3       $S \leftarrow$  train classifier on Y;
4       $X^1 \leftarrow$  pick a portion of the unlabeled data
5      if (confidence of  $S = \lambda$ ) {
6         $X_p^1 \leftarrow$  classify  $X^1$  by  $S$  and select high confident examples
7        Add  $X_p^1$  to Y
8        Remove  $X_p^1$ , from X
9      }
10   Return  $Y_0$  /*Final trained dataset*/
11 End
  
```

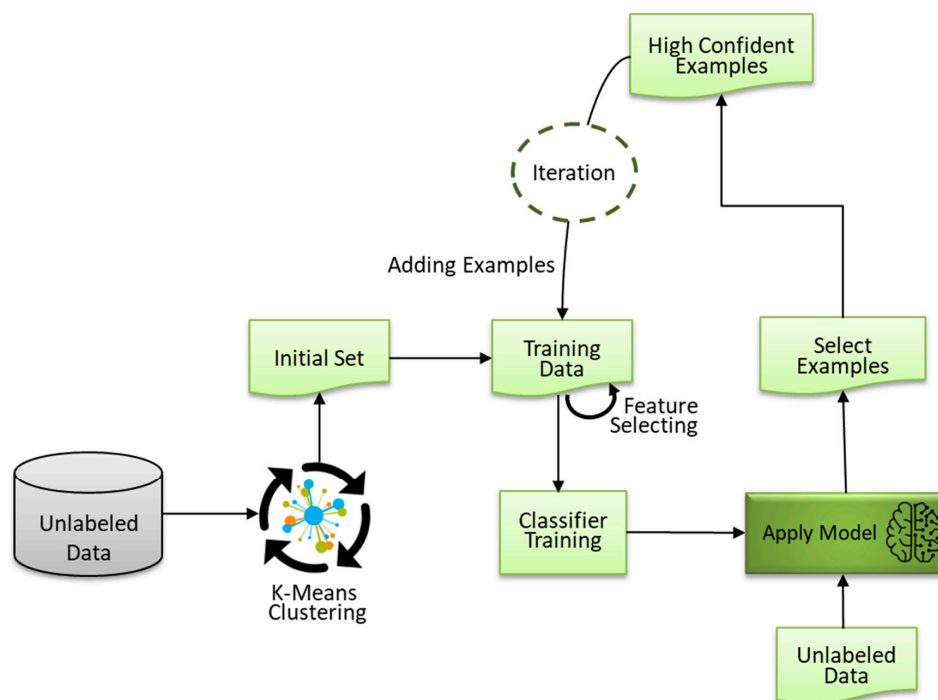


Figure 4. Self-training for learning sentiment classifier.

4.2.6. Active Self-Training Based Sentiment Learner

Active learning techniques select the most informative unlabeled examples to predict their label and include them in the training set [63]. Many researchers have successfully combined active learning with self-training to decrease the human labeling struggle and enhance the classification performance [33,64]. Motivated by the existing research [33], we integrate active learning with self-training to select the most informative and high confident level examples. In this method, first, an initial classifier is trained using the initial training set created by SBC technique and then applied to the unlabeled data. Subsequently, some of the best informative examples from unlabeled text data in each iteration are selected by active learning using uncertainty sampling and a class labeled is assigned to them by a domain expert. In uncertainty sampling, the unlabeled examples with maximum uncertainty are considered as the best informative case [49]. Maximum uncertainty refers that the learners have less certainty to classify the unlabeled examples. In this case, labeling these examples to add to the training set is useful, as it furnishes unrevealed sentiment clues that can better sentiment classifier learning. At the same time, some of the high confidence level classified examples are selected in every iteration by self-training with a specific threshold. Further, these informative and high confident level classified examples are included in the training text set for the next iteration. In the next iteration, the classifier is retrained on the enrich training text set. This process is iterated several times until a stopping criterion is obtained. When the training process is converged, then the final trained sentiment classifier for test review classification is obtained. The procedure used for AST based sentiment learning is described in Algorithm 3. The work flow for AST-SL is shown in Figure 5.

Algorithm 3 Active self-training (AST) based learning

inputs: $X = \{x_1, x_2, \dots, x_m\}$; /* unlabeled data of m instances */

$Y = \{y_1, y_2, \dots, y_m\}$; /* labeled data of m instances */

S /* classifier */

$\lambda = \{\lambda_1, \lambda_2\}$; /* Thresholds */

output: /* labeled data $|Y_O|$, $leq |Y| + |X|$ leq : less than or equal to */

$Y_O = \{y_1, y_2, \dots, y_n\}$

```

1  Begin
2    WHILE  $X$  contain instances {
3       $S \leftarrow$  train classifier on  $Y$ ;
4       $X^1 \leftarrow$  pick a portion of the unlabeled data
5       $X_{1p}^1, X_{2p}^1 \leftarrow$  classify  $X^1$  by  $S$  and select high confident examples ( $X_{1p}^1$ ) and high
informative examples ( $X_{2p}^1$ ) respectively, use  $\lambda_1$  and  $\lambda_2$ 
6      If(confidence =  $\lambda_1$ ){
7         $X_{1p}^1 \leftarrow$  classify  $X^1$  by  $S$  and select high confident examples
8        Add  $X_{1p}^1$  to  $Y$ 
9      }
10     If(confidence =  $\lambda_2$ ){
11        $X_{2p}^1 \leftarrow$  classify  $X^1$  by  $S$  and select high informative examples
12       Add  $X_{2p}^1$  to Pool /* for human annotation */
13     }
14      $H_x \leftarrow$  Annotate instances from Pool /*Annotate by Human*/
15     Add  $H_x$  to  $Y$ 
16     Remove  $X_{1p}^1, X_{2p}^1$  from  $X$ 
17   }
18   Return  $Y_O$ 
19 End

```

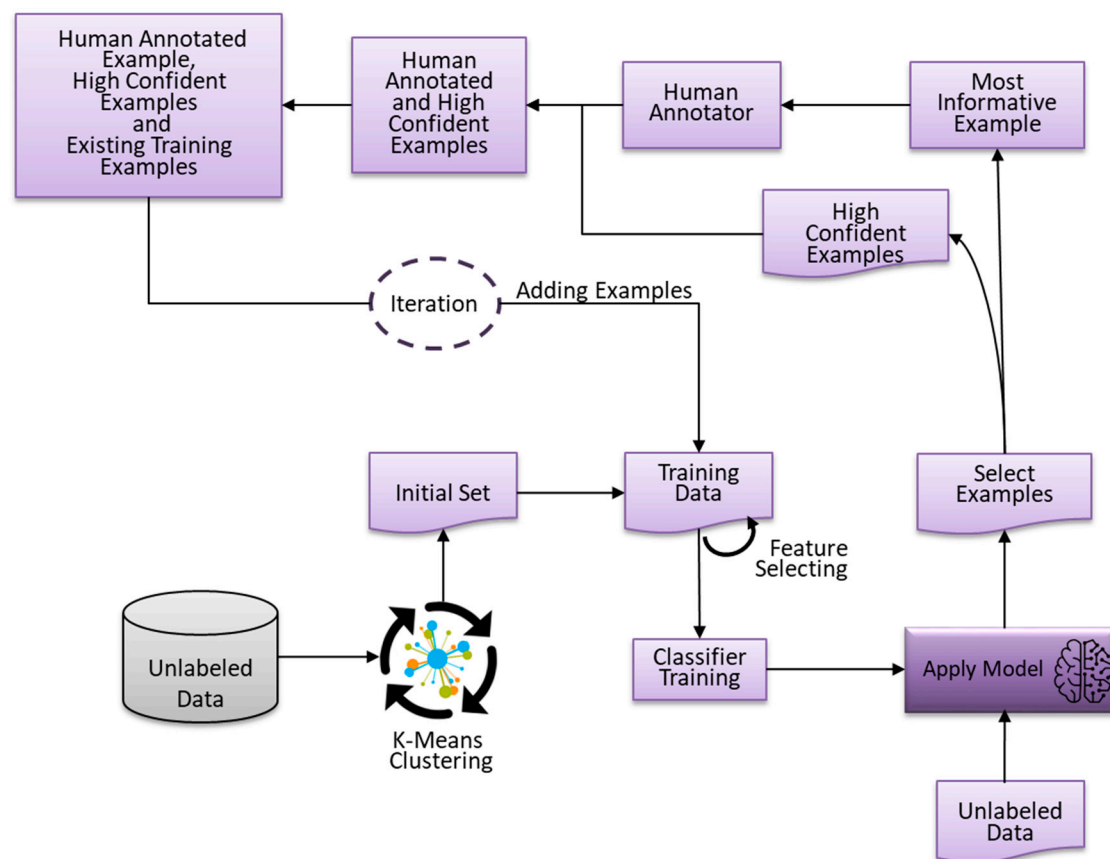


Figure 5. Active self-training for learning sentiment classifier.

4.2.7. Active Self-Co-Training (ASCT) for Sentiment Classification

In this method, we combined co-training with self-training and active learning. Given a set of labeled data (Y) and Unlabeled data (X). First, the feature space of the labeled data Y which is created with SBC technique is divided into two views, $Y = (Y^1, Y^2)$. Then a pool of unlabeled data, let us represent it by X^1 is also divided into two views (X_1^1, X_2^1). After then, separate classifiers $S1$ and $S2$ are trained on one view, Y^1, Y^2 respectively.

Further, the classifier $S1$ is applied to the unlabeled view X_1^1 to select some of the high confident level classified examples according to a specific threshold predicted by the classifier $S1$ and added into the training data. Then applying classifier $S2$ to the unlabeled view X_2^1 . Some of the best informative examples are selected by active learning (utilizing uncertainty sampling scheme) and assigned to a domain expert for manual annotation that is added into the training data. This process continues for several iterations until the termination criteria is satisfied. After several numbers of iterations, we obtain a set of labeled data from the co-training components of classifier $S1$ and $S2$. The labeled data obtained from two co-training components are integrated into a final classifier. The final trained sentiment classifier is then used for test review classification. The procedure used for ASCT based sentiment learning is described in Algorithm 4. The workflow for ASCT-SL is shown in Figure 6.

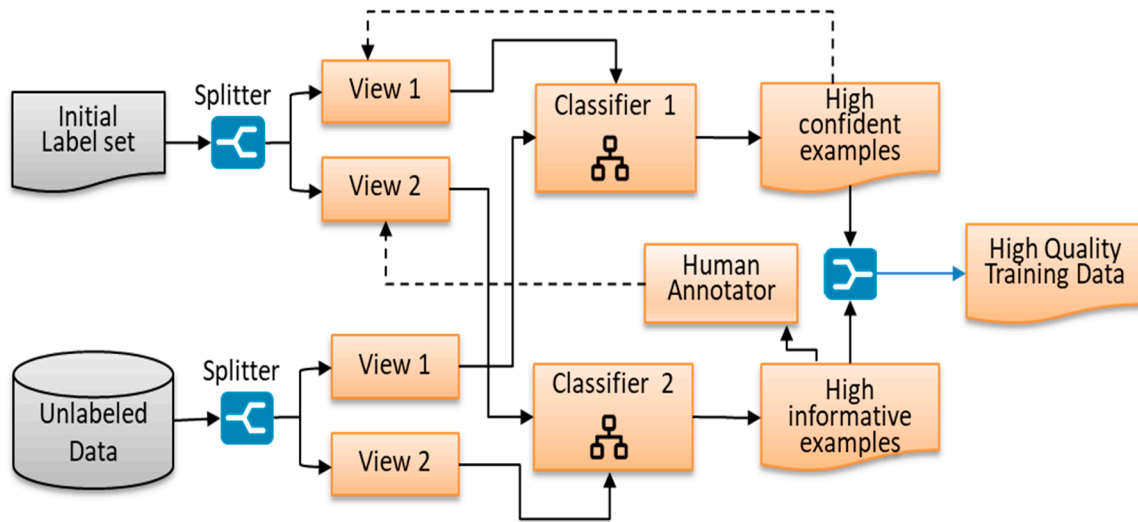


Figure 6. Active Self-Co-Training for learning Sentiment classifier.

Algorithm 4 Active self co-training (ASCT) based learning

inputs: $X = \{x_1, x_2, \dots, x_m\}$; /*unlabeled data of n instances*/

$Y = \{y_1, y_2, \dots, y_m\}$; /*labeled data of n instances*/

$S = \{s_1, s_2\}$; /*classifiers*/

$\lambda = \{\lambda_1, \lambda_2\}$; /*Thresholds*/

output: /* labeled data, $|Y_O| \leq |Y| + |X|$, \leq : less than or equal to */

$Y_O = \{y_1, y_2, \dots, y_n\}$

```

1  BEGIN
2      WHILE X contain instances {
3          /* divides labeled data into two views*/
4           $Y^1, Y^2 \leftarrow \text{Split } Y$ 
5           $s_1, s_2 \leftarrow \text{train classifiers on each view } Y^1, Y^2 \text{ respectively}$ 
6           $X^1 \leftarrow \text{pick a certain number of the unlabeled data}$ 
7          /* divides unlabeled data into two views */
8           $X_1^1, X_2^1 \leftarrow \text{Split } X^1$ 
9          If(confidence =  $\lambda_1$ ){
10              $X_{1p}^1 \leftarrow \text{classify } X_1^1 \text{ by } s_1 \text{ and pick high confident examples}$ 
11             Add  $X_{1p}^1$  to Y
12         }
13         If(confidence =  $\lambda_2$ ){
14              $X_{2p}^1 \leftarrow \text{classify } X_2^1 \text{ by } s_2 \text{ and pick high informative examples}$ 
15             Add  $X_{2p}^1$  to Pool /* for human annotation*/
16         }
17          $H_x \leftarrow \text{Annotate instances from Pool /* Annotate by Human */}$ 
18         Add  $H_x$  to Y
19         Remove  $X_{1p}^1, X_{2p}^1$  from X,
20     }
21     Return  $Y_O$ 
22 End

```

4.3. Sentiment Analysis with Classification Fusion

The idea of classification fusion is to utilize multiple classification models and combine their predictions in some way, such as voting. The primary aim of classification fusion is to decrease the error rate in classification tasks, which may occur in single classifiers [65], and also to improve generalization

ability/robustness over a single classifier and enhance the classification performance of the system. Classifier fusion produces better results when there is an obvious diversity among classifiers. In this paper, the predictions (votes) of all the designed and implemented sentiment classification models on test review are combined using majority voting to determine the final sentiment orientation and produce improved results. The details of classification fusion for designed sentiment classification models are given in the below case study.

Case Study—Classification Fusion

In this case, each sentiment learner predicts the sentiment label (positive or negative) of test textual review document. The predictions from each sentiment learner on test review are then combined and provided to a fusion module that determines the final sentiment polarity of test review based on a majority voting scheme. For example, we have a test textual review document from movie review dataset given in Figure 7. The five different designed sentiment learners, WCSL-SL, HRPL-SL, ST-SL, AST-SL, ASCT-SL predict the sentiment polarity for the input textual review document. The predicted polarities of test textual review document are combined based on majority voting using a threshold t (i.e., $t \geq c$, $c = 3$, where c is the value of threshold) to determine the same polarity (final sentiment polarity) predicted by at least three sentiment learners.

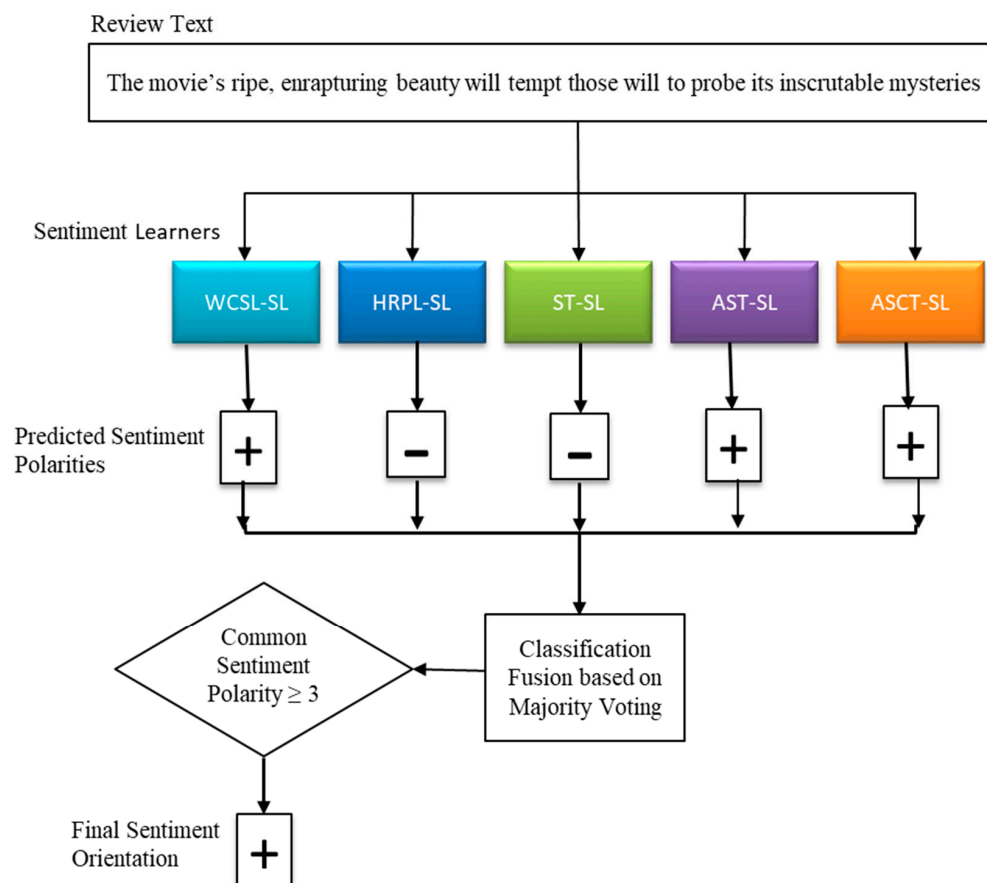


Figure 7. Prediction of final sentiment polarity based on classification fusion.

5. Results, Evaluation and Discussion

5.1. Datasets

We evaluate our proposed unified framework on the two different domain datasets, the Cornell movie review dataset [7] and Amazon product review datasets [66]. The Cornell movie review dataset

consists of 2000 reviews contain 1000 positive and 1000 negative reviews. The Amazon product reviews datasets contain four domains, i.e., Book, DVD, Electronics, and Kitchen; each domain consists of 1000 positive and 1000 negative reviews. For each domain dataset, we used 80 percent instances as the training set and the remaining 20 percent instances as the test set. We select 10 percent instances of the training set as the initial labeled data with reinforcement of the labeled data using the SBC technique and used the remaining instances as the unlabeled data. The classification performance of final trained sentiment learners is evaluated on test documents.

5.2. Experimental Setting

Similar to state-of-the-art techniques for multi-domain datasets, we used 5-fold cross-validation for all the experiments in multi-domain data sets. Each dataset was split randomly into a training set and testing set in 5-fold cross-validation, one of which is used for testing, and the remaining including initial set are used for training. The classification performance is estimated by the averaged results of 5-fold cross-validation. For the implementation, we exploited Rapidminer Studio [67], which provides a complete environment for machine learning, text mining, deep learning, data processing and predictive analytics. We used SVM (Linear) for HRPL-SL. Multinomial Naive Bayes (MNBs) is utilized as the base classifier of ST-SL, AST-SL, ASCT-SL [68,69] with all parameters set at their default values. SVM (Linear) is also used for the final trained sentiment learners to classify test textual reviews.

We set the thresholds of text samples as $p = 3$, $n = 2$ (p = positive class, n = negative class) for ST-SL, AST-SL, ASCT-SL. The total number of iterations is set to 90 for Algorithm 2, 3 and 4. Sentiment features were extracted and selected on the training dataset by using MRMR selection method. The dimension for appropriate sentiment feature is 800. We utilized the paired t-test ($P < 0.05$) to calculate the accuracies of all techniques.

5.3. Evaluation Metrics

We used accuracy as an evaluation criterion to calculate the overall sentiment classification performance. The classification accuracy of a method is measured based on the testing texts.

$$\text{Accuracy} = \text{correctly classified reviews} / \text{total number of review} \quad (2)$$

5.4. Experimental Results and Discussion

In this section, we evaluate and show the effects of various parts of the proposed approach on multi-domain datasets. We also evaluate and discuss how the proposed approach could overcome the sentiment analysis problem.

Experiment 1. *Evaluation of induced WCSL for review document scoring and classification.*

The induced high-quality WCSL in conjunction with linguistic and semantic knowledge achieved good performance on Book, DVD, Electronics and Kitchen datasets and best classification performance on Movie domain dataset. The effective performance of induced high-quality WCSL is due to the wide coverage of general sentiment words including context-aware domain-specific sentiment words which matched many sentiment words in the review documents. The classification accuracy achieved by WCSL on multi-domain datasets is shown in Figure 8.

Experiment 2. *Evaluation of high ranking pseudo-labeled examples for review sentiment classification.*

The selection of high ranking pseudo-labeled example by WCSL for sentiment classification achieved good performance on Book, DVD and Electronics domain datasets and best performance on Kitchen and Movie domains. The influential performance of this approach is due to the selection of

high absolute score sentiment examples, which comprise effective sentiment words. The classification performance achieved by HRPL-SL is shown in Figure 9.

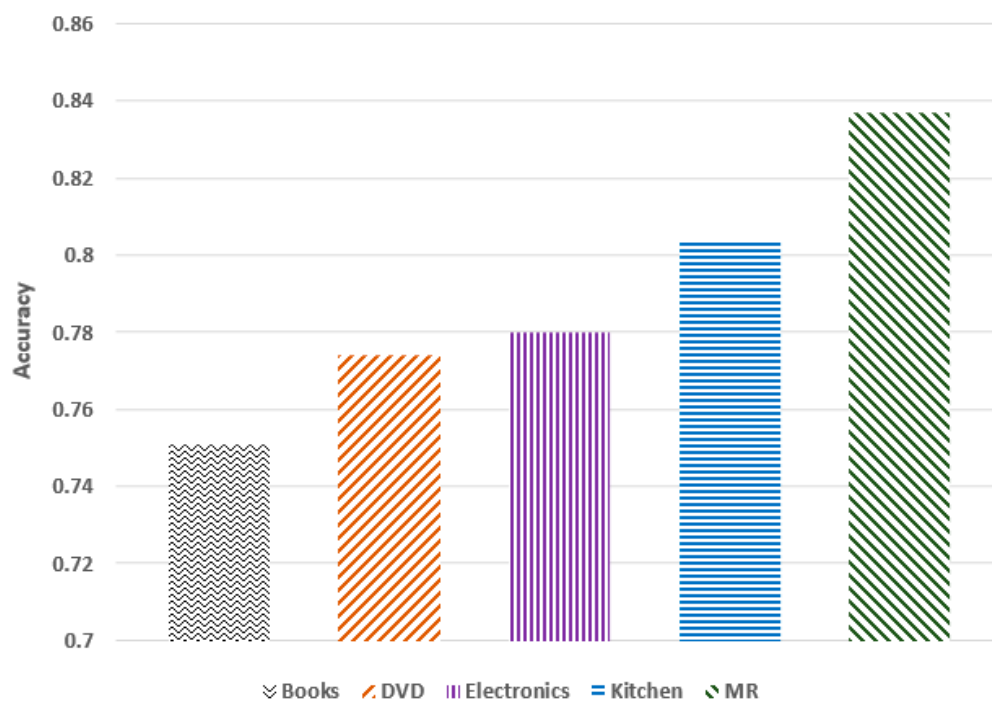


Figure 8. This figure shows the classification performance achieved by WCSL-SL on multi-domain datasets.

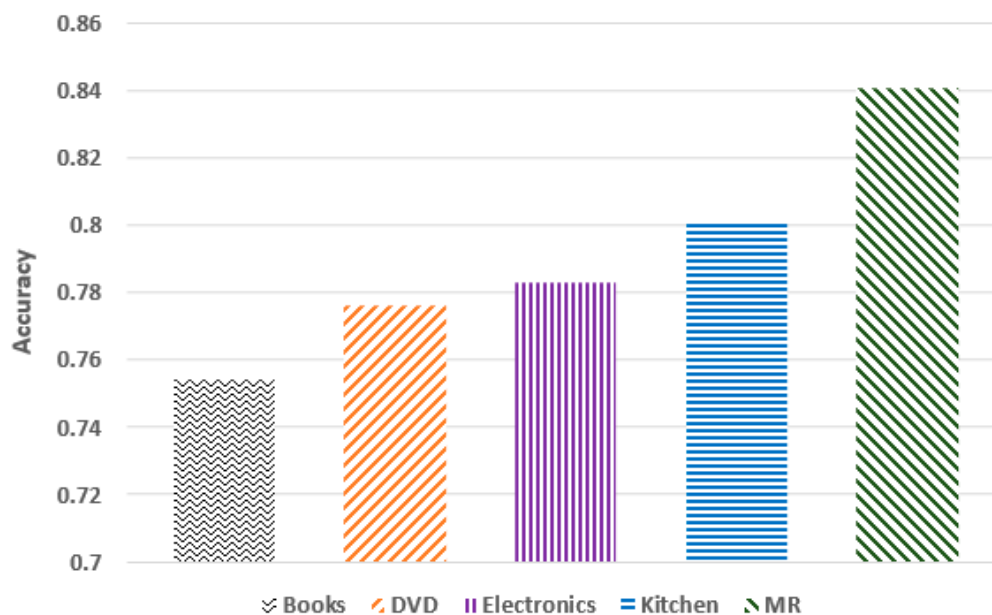


Figure 9. This figure shows the classification performance achieved by HRPL-SL on multi-domain datasets.

Experiment 3. Evaluation of sentiment learner based on self-training.

In this experiment, we used specific confidence threshold values while predicting the labels of unlabeled data. The confidence values evaluated as thresholds are 0.5, 0.6, 0.7 and 0.8. Those classified examples whose confident score is high are selected and added in the label training set. The selection of classified examples based on low confidence threshold values, i.e., 0.5 and 0.6 decline the learning

process due to the misclassified examples. Therefore, we adopted the high confident threshold value 0.8 to select more valuable p positive and n negative examples for the complete learning process. Figure 10 shows the performance of the final trained sentiment learner on multi-domain datasets based on self-training with four different confidence threshold values, 0.5, 0.6, 0.7 and 0.8. From Figure 10, it is clear that the final trained sentiment learner based on ST-SL with confident threshold 0.8 achieved good classification performance.

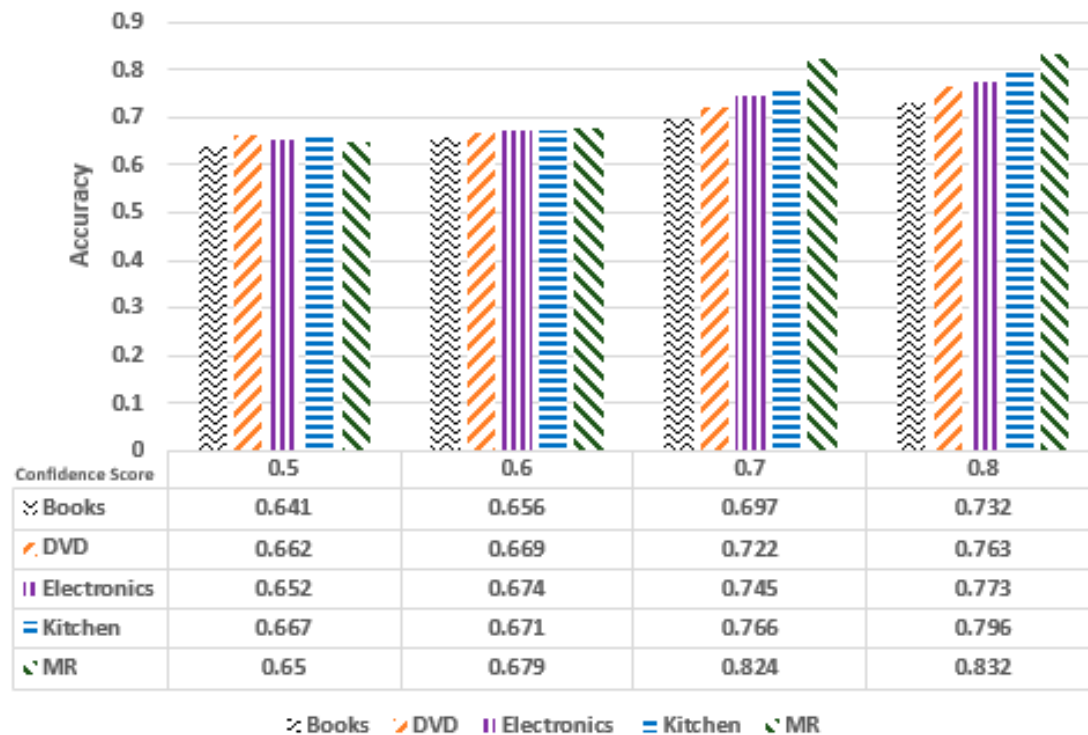


Figure 10. This figure shows the classification performance based on ST-SL with 4 different confidence threshold values on multi-domain datasets.

Experiment 4. Evaluation of sentiment learner based on active self-training.

In this experiment, some of the highly-uncertain, most-informative examples (misclassified unlabeled examples) whose confidence score is close to 0.5 are actively selected for manual annotation. According to the method proposed in [70], the misclassified unlabeled examples in active learning may carry more information than correctly classified unlabeled examples. Simultaneously, some of the high-confidence level classified examples with threshold value 0.8 are selected by self-training. The good performance of this approach on multi-domain datasets is due to the selection of classified examples with active learning together with self-training. The classification performance obtained by final sentiment learner based on AST is shown in Figure 11.

Experiment 5. Evaluation of sentiment learner based on active self-co-training.

In this experiment, most informative and high confident level classified examples are selected by active learning and self-training, respectively. In details, two different classifiers (S_1 , S_2) with different labeled (Y^1 , Y^2) and unlabeled views (X_1^1 , X_2^1) were hired for active learning (AL) and self-training (ST) respectively. The most informative examples predicted by classifier S_1 whose confidence score is closest to 0.5 are selected by active learning for human annotation, more especially uncertainty sampling [70]. Further, the most confident level examples with threshold value 0.8 are selected by self-training. The final trained sentiment learner based on Active-Self-Co-Training achieved good

performance on Book, DVD, Electronics and Kitchen domain dataset, and best performance on Movie domain dataset. Figure 12 shows the performance of the final sentiment learner based on ASCT on multi-domain datasets.

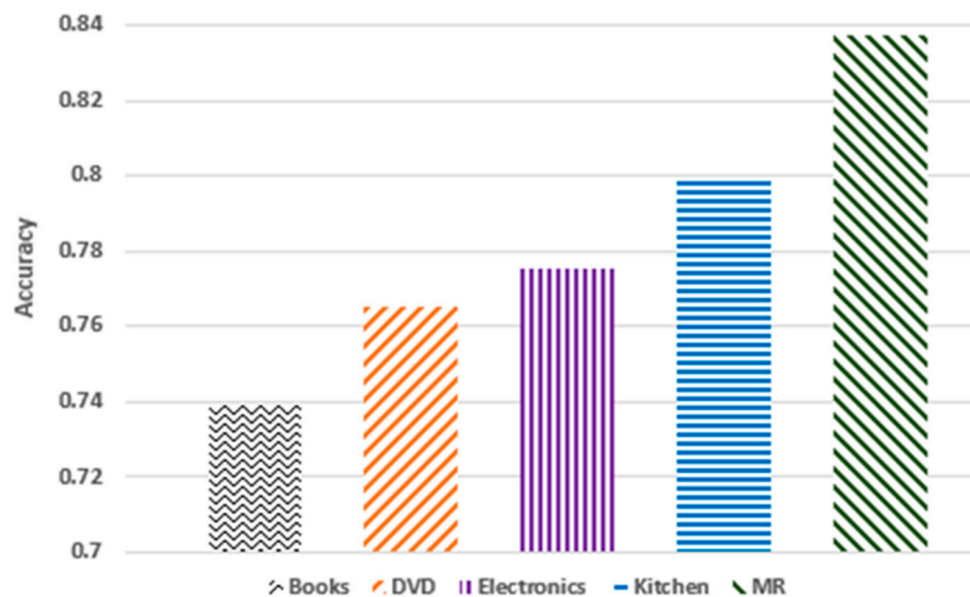


Figure 11. This figure shows the classification performance achieved by AST-SL on multi-domain datasets.

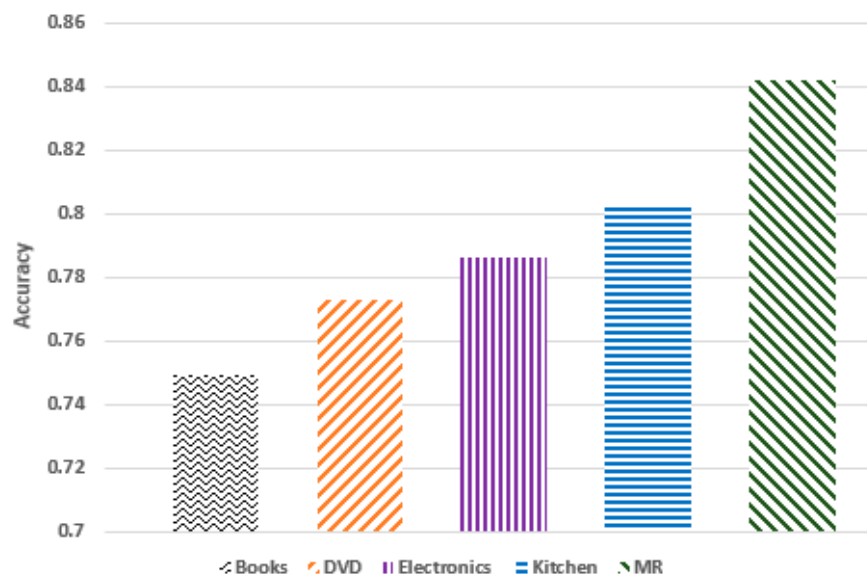


Figure 12. This figure shows the classification performance achieved by ASCT-SL on multi-domain test datasets.

Experiment 6. Evaluation of final sentiment orientation based on classification fusion.

In this experiment, the predictions of final trained sentiment learners on test reviews are combined based on majority voting exploiting a threshold t (i.e., $t \geq c$, $c = 3$, where c is the value of threshold). In detail, the same sentiment orientation predicted by at least three sentiment learners is considered as the final sentiment polarity of the class. The approach of classification fusion based on majority voting for multi-domain sentiment classification achieved high performance. The classification performance obtained by CF is shown in Figure 13.

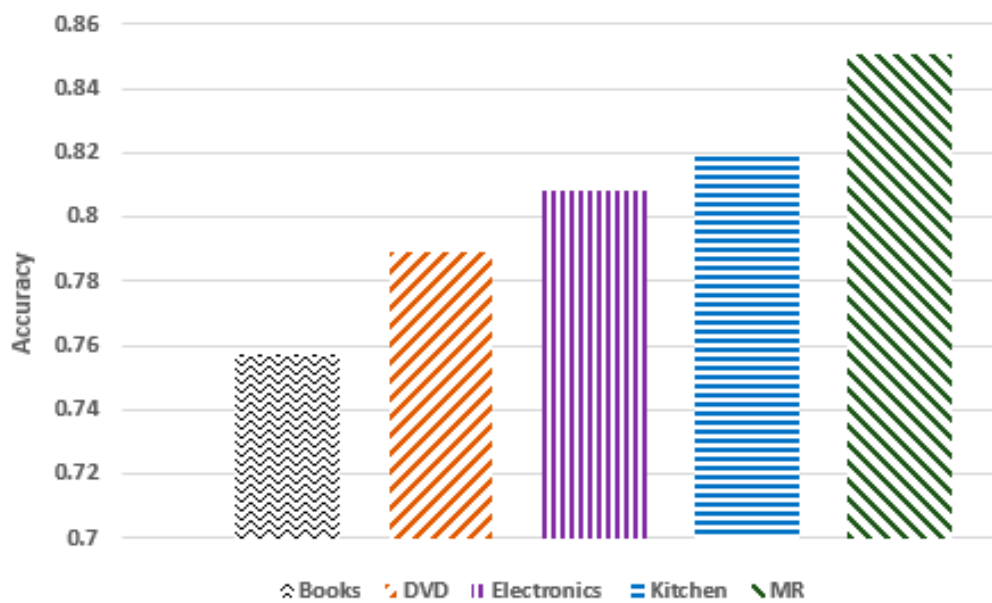


Figure 13. This figure shows the classification performance achieved by CF on multi-domain test datasets.

5.5. Result Discussion

The performance of all the designed and implemented sentiment analysis techniques on multi-domain datasets is shown in Figure 14 along with the Table. As shown in Figure 14, the WCSL based sentiment classification obtained the best performance on Kitchen review dataset; the HRPL-SL achieved the best performance on Books and DVD datasets; and the ASCT-SL achieved the best performance on Electronics and Movie review datasets, while the performances of ST-SL and AST-SL on multiple domain datasets were satisfactory. Further, the CF-based on simple majority voting improved the classification performance. In this approach (CF), we combined the classification predictions from all the sentiment learners on test datasets using majority voting and determined the final sentiment orientation for each review text in the test datasets.

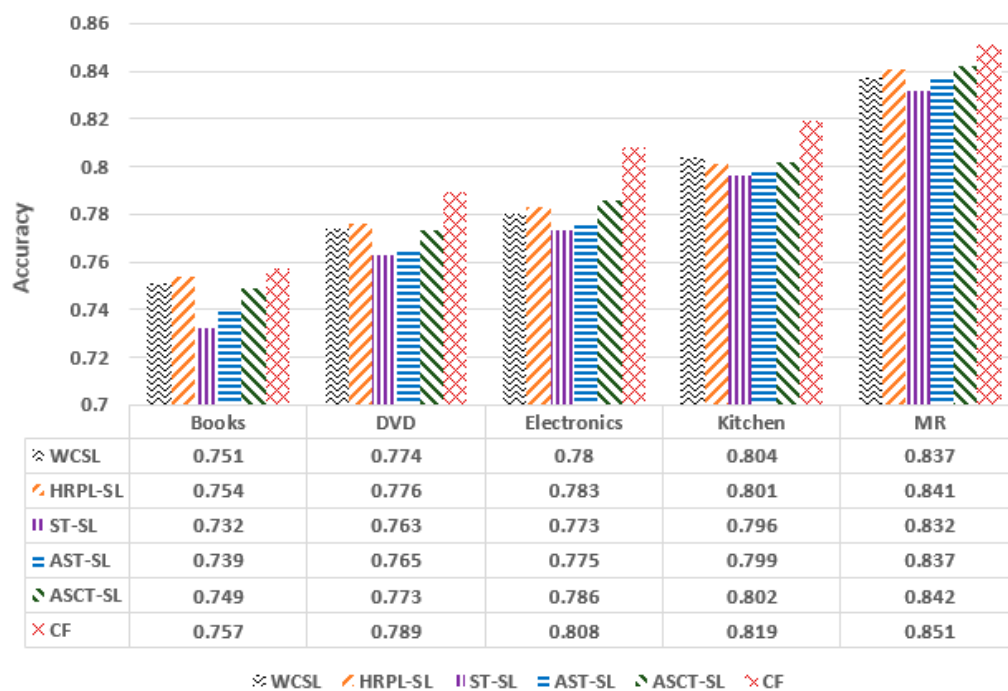


Figure 14. This figure shows the classification performance achieved by different designed sentiment classification models.

5.6. Performance Comparison with State-of-the-Art Methods

In order to show the effectiveness of our proposed approach, we compared it with the following state-of-the-art methods for textual sentiment classification.

Active deep network (ADN) [41]: ADN is a semi-supervised learning method which exploits the embedding information from the huge amount of unlabeled data together with a small number of labeled data for review sentiment classification.

Dual-view co-training [31]: A dual-view co-training method, which makes use of the original and antonymous views in pairs, in the training, bootstrapping and testing process, all based on a joint observation of two views for semi-supervised sentiment classification.

Self-training-S [61]: A self-training approach in which multiple feature subspace-based classifiers are used to explore a set of good features and select informative samples for automatic labeling. In this approach, the top two informative samples are selected for manual annotation in each iteration.

LCCT [37]: A semi-supervised model based on lexicon and corpus-based co-training for sentiment classification.

CASCT [29]: A co-operative hybrid semi-supervised learning for text sentiment classification

Figure 15 along with table shows sentiment classification performance of LESSA against state-of-the-art approaches. It is clear that our proposed approach LESSA achieved the highest performance in terms of accuracy on the multi-domain datasets. There are many reasons for the high performance of the proposed approach as compared to other approaches. The first reason for achieving the highest performance of the proposed SA framework is the construction of high-quality wide-coverage sentiment lexicons, leveraging linguistic and semantic knowledge. The second reason is the selection of most representative instances for the initial training set. The third reason is training high-quality data with semi-supervised techniques, considering both the most informative and high-confident examples. The fourth reason is the removal of noisy and redundant features from the text data during pre-processing. The fifth reason is the extraction and selection of sentiment bearing words from textual data in each iterative round during sentiment classifier learning. The sixth reason is the prediction of final sentiment orientation based on classification fusion.

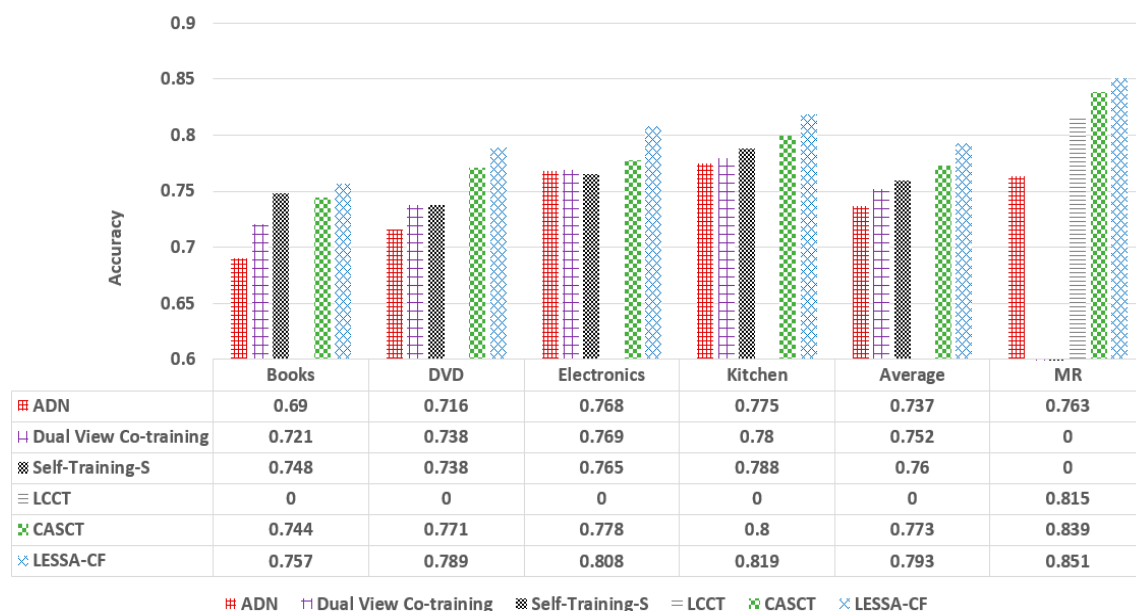


Figure 15. Comparison of LESSA-CF with state-of-the-art approaches for textual sentiment classification.

6. Conclusions

A massive amount of unstructured textual data in the form of users' reviews is available online. This unstructured data contains beneficial information for business analysis and decision making. SA is

an influential research area, which aims to classify online user's reviews into positive and negative classes. In this context, diverse techniques such as SML, SSL and LS for textual sentiment classification have been proposed. The main difficulty in SML approach is the absence of enough labeled training data for heterogeneous domains. SSL approaches for textual sentiment classification have improved their performance while suffering from research challenges, whereas, most of the SSL approaches could not identify efficaciously important concealed information in the unstructured data. Moreover, the most significant sentiment feature extraction and selection are not handled successfully. Noisy and redundant features are also ignored in the textual data. Many of them randomly selected initial training set for first classifier training. Besides, the lexicon-based approaches for textual sentiment classification need improvement. In order to tackle and investigate such concerns, we propose a novel unified sentiment analysis framework, which we call "LeSSA" for textual sentiment classification. LeSSA is a novel framework, which has been designed for multi-domain sentiment classification. LeSSA is applicable to heterogeneous domain sentiment classification. LeSSA predicts the accurate sentiment label for the unseen textual reviews. A number of experiments were performed to validate the usefulness of LeSSA. The experimental results on multi-domain benchmark datasets show that LeSSA outperforms existing SSL, lexicon-based and hybrid approaches. In future work, we will incorporate other techniques, such as deep learning and linear discriminate analysis for testing LeSSA.

Author Contributions: Y.-K.L. provided supervision, remarks and financial support for the paper; J.K. presented the idea; devised, designed and accomplished all the experiments and wrote the paper.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ICT Consilience Creative program (IITP-2019-2015-0-00742) supervised by the IITP (Institute for Information and communications Technology Planning and Evaluation).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, J.; Alam, A.; Hussain, J.; Lee, Y.-K. EnSWF: Effective features extraction and selection in conjunction with ensemble learning methods for document sentiment classification. *Appl. Intell.* **2019**, *49*, 1–23. [\[CrossRef\]](#)
2. Meghanathan, N. *Graph Theoretic Approaches for Analyzing Large-Scale Social Networks*, 1st ed.; IGI Global: Hershey, PA, USA, 2017.
3. Xue, J.; Macropol, K.; Jia, Y.; Zhu, T.; Gelles, R.J. Harnessing big data for social justice: An exploration of violence against women-related conversations on Twitter. *Human Behav. Emerg. Technol.* **2019**, *1*, 269–279. [\[CrossRef\]](#)
4. Liu, B. Sentiment analysis and opinion mining. *Synth. Lect. Hum. Lang. Technol.* **2012**, *5*, 1–167. [\[CrossRef\]](#)
5. Pang, B.; Lee, L. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.* **2008**, *2*, 1–135. [\[CrossRef\]](#)
6. Cambria, E. Affective computing and sentiment analysis. *IEEE Intell. Syst.* **2016**, *31*, 102–107. [\[CrossRef\]](#)
7. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; pp. 79–86.
8. Ye, Q.; Zhang, Z.; Law, R. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Syst. Appl.* **2009**, *36*, 6527–6535. [\[CrossRef\]](#)
9. Tripathy, A.; Agrawal, A.; Rath, S.K. Classification of sentiment reviews using n-gram machine learning approach. *Expert Syst. Appl.* **2016**, *57*, 117–126. [\[CrossRef\]](#)
10. Aggarwal, C.C.; Zhai, C. An introduction to text mining. In *Mining Text Data*; Springer: Berlin, Germany, 2012; pp. 1–10.
11. Domingos, P.; Pazzani, M. On the optimality of the simple Bayesian classifier under zero-one loss. *Mach. Learn.* **1997**, *29*, 103–130. [\[CrossRef\]](#)
12. Khoo, C.S.; Johnkhan, S.B. Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons. *J. Inf. Sci.* **2018**, *44*, 491–511. [\[CrossRef\]](#)
13. Hutto, C.J.; Gilbert, E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, Ann Arbor, MI, USA, 1–4 June 2014.

14. He, Y.; Zhou, D. Self-training from labeled features for sentiment analysis. *Inf. Process. Manag.* **2011**, *47*, 606–616. [[CrossRef](#)]
15. Tan, S.; Wang, Y.; Cheng, X. Combining learn-based and lexicon-based techniques for sentiment detection without using labeled examples. In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, 20–24 July 2008; pp. 743–744.
16. Mudinas, A.; Zhang, D.; Levene, M. Combining lexicon and learning based approaches for concept-level sentiment analysis. In Proceedings of the First International Workshop on Issues of sentiment Discovery and Opinion Mining, Beijing, China, 12–16 August 2012; p. 5.
17. Nielsen, F.Å. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *arXiv* **2011**, arXiv:1103.2903.
18. Hu, M.; Liu, B. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004; pp. 168–177.
19. Taboada, M.; Brooke, J.; Tofiloski, M.; Voll, K.; Stede, M. Lexicon-based methods for sentiment analysis. *Comput. Linguist.* **2011**, *37*, 267–307. [[CrossRef](#)]
20. Strapparava, C.; Valitutti, A. Wordnet affect: An affective extension of wordnet. In Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04), Lisbon, Portugal, 26–28 May 2004; p. 40.
21. Stone, P.J.; Hunt, E.B. A computer approach to content analysis: Studies using the general inquirer system. In Proceedings of the Spring Joint Computer Conference, Detroit, Michigan, 21–23 May 1963; pp. 241–256.
22. de Albornoz, J.C.; Plaza, L.; Gervás, P. SentiSense: An easily scalable concept-based affective lexicon for sentiment analysis. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, 23–25 May 2012.
23. Riloff, E.; Wiebe, J. Learning extraction patterns for subjective expressions. In Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, Sapporo, Japan, 11–12 July 2003; pp. 105–112.
24. Mohammad, S.M.; Kiritchenko, S.; Zhu, X. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv* **2013**, arXiv:1308.6242.
25. Cambria, E.; Poria, S.; Hazarika, D.; Kwok, K. SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
26. Baccianella, S.; Esuli, A.; Sebastiani, F. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta, 17–23 May 2010; pp. 2200–2204.
27. Xiaojin, Z. *Semi-Supervised Learning Literature Survey*; Computer Sciences TR; University of Wisconsin-Madison Department of Computer Sciences: Madison, WI, USA, 2008; Volume 1530.
28. Qiu, L.; Zhang, W.; Hu, C.; Zhao, K. Selc: A self-supervised model for sentiment classification. In Proceedings of the 18th ACM Conference on Information and Knowledge Management, Hong Kong, China, 2–6 November 2009; pp. 929–936.
29. Li, Y.; Lv, Y.; Wang, S.; Liang, J.; Li, J.; Li, X. Cooperative Hybrid Semi-Supervised Learning for Text Sentiment Classification. *Symmetry* **2019**, *11*, 133. [[CrossRef](#)]
30. Wan, X. Co-training for cross-lingual sentiment classification. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1, Singapore, 2–7 August 2009; pp. 235–243.
31. Xia, R.; Wang, C.; Dai, X.-Y.; Li, T. Co-training for semi-supervised sentiment classification based on dual-view bags-of-words representation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2015; pp. 1054–1063.
32. Li, S.; Xue, Y.; Wang, Z.; Zhou, G. Active learning for cross-domain sentiment classification. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China, 3–9 August 2013.
33. Hajmohammadi, M.S.; Ibrahim, R.; Selamat, A.; Fujita, H. Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples. *Inf. Sci.* **2015**, *317*, 67–77. [[CrossRef](#)]

34. Wang, B.; Spencer, B.; Ling, C.X.; Zhang, H. Semi-supervised self-training for sentence subjectivity classification. In Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence, Windsor, ON, Canada, 28–30 May 2008; pp. 344–355.
35. Blum, A.; Mitchell, T. Combining labeled and unlabeled data with co-training. In Proceedings of the Eleventh Annual Conference on Computational Learning Theory, Madison, WI, USA, 24–26 July 1998; pp. 92–100.
36. Zhang, M.-L.; Zhou, Z.-H. CoTrade: Confident co-training with data editing. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *41*, 1612–1626. [[CrossRef](#)]
37. Yang, M.; Tu, W.; Lu, Z.; Yin, W.; Chow, K.-P. LCCT: A semi-supervised model for sentiment classification. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–5 June 2015; pp. 546–555.
38. Hu, R.; Namee, B.M.; Delany, S.J. Active learning for text classification with reusability. *Expert Syst. Appl.* **2016**, *45*, 438–449. [[CrossRef](#)]
39. Olsson, F. *A Literature Survey of Active Machine Learning in the Context of Natural Language Processing*; Swedish Institute of Computer Science: Stockholm, Sweden, 2009.
40. Baldrige, J.; Osborne, M. Active learning and the total cost of annotation. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 25–26 July 2004; pp. 9–16.
41. Zhou, S.; Chen, Q.; Wang, X. Active deep networks for semi-supervised sentiment classification. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Beijing, China, 23–27 August 2010; pp. 1515–1523.
42. Lin, Y.; Zhang, J.; Wang, X.; Zhou, A. An information theoretic approach to sentiment polarity classification. In Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality, Lyon, France, 16 April 2012; pp. 35–40.
43. Xu, T.; Peng, Q.; Cheng, Y. Identifying the semantic orientation of terms using S-HAL for sentiment analysis. *Knowl. Based Syst.* **2012**, *35*, 279–289. [[CrossRef](#)]
44. Saleh, M.R.; Martín-Valdivia, M.T.; Montejo-Ráez, A.; Ureña-López, L. Experiments with SVM to classify opinions in different domains. *Expert Syst. Appl.* **2011**, *38*, 14799–14804. [[CrossRef](#)]
45. Dey, A.; Jenamani, M.; Thakkar, J.J. Senti-N-Gram: An n-gram lexicon for sentiment analysis. *Expert Syst. Appl.* **2018**, *103*, 92–105. [[CrossRef](#)]
46. Miller, G.A. WordNet: A lexical database for English. *Commun. ACM* **1995**, *38*, 39–41. [[CrossRef](#)]
47. Hatzivassiloglou, V.; McKeown, K.R. Predicting the semantic orientation of adjectives. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, Madrid, Spain, 7–12 July 1997; pp. 174–181.
48. Igo, S.P.; Riloff, E. Corpus-based semantic lexicon induction with web-based corroboration. In Proceedings of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics, Boulder, CO, USA, 5 June 2009; pp. 18–26.
49. Zhu, J.; Wang, H.; Yao, T.; Tsou, B.K. Active learning with sampling by uncertainty and density for word sense disambiguation and text classification. In Proceedings of the 22nd International Conference on Computational Linguistics—Volume 1, Manchester, UK, 18–22 August 2008; pp. 1137–1144.
50. Kang, J.; Ryu, K.R.; Kwon, H.-C. Using cluster-based sampling to select initial training set for active learning in text classification. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Sydney, Australia, 26–28 May 2004; pp. 384–388.
51. Li, S.; Ju, S.; Zhou, G.; Li, X. Active learning for imbalanced sentiment classification. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, 12–14 July 2012; pp. 139–148.
52. Poria, S.; Cambria, E.; Gelbukh, A. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowl. Based Syst.* **2016**, *108*, 42–49. [[CrossRef](#)]
53. Wang, Y.; Huang, M.; Zhao, L. Attention-based LSTM for aspect-level sentiment classification. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 606–615.
54. Zhao, W.; Ye, J.; Yang, M.; Lei, Z.; Zhang, S.; Zhao, Z. Investigating capsule networks with dynamic routing for text classification. *arXiv* **2018**, arXiv:1804.00538.

55. Zhao, W.; Peng, H.; Eger, S.; Cambria, E.; Yang, M. Towards Scalable and Reliable Capsule Networks for Challenging NLP Applications. *arXiv* **2019**, arXiv:1906.02829.
56. Khan, J.; Jeong, B.S. Summarizing customer review based on product feature and opinion. In Proceedings of the 2016 international Conference on Machine Learning and Cybernetics (ICMLC), Jeju Island, South Korea, 10–13 July 2016; pp. 158–165.
57. Khan, J.; Jeong, B.S.; Lee, Y.-K.; Alam, A. Sentiment analysis at sentence level for heterogeneous datasets. In Proceedings of the Sixth International Conference on Emerging Databases: Technologies, Applications, and Theory, Jeju Island, Korea, 17–19 October 2016; pp. 159–163.
58. Turney, P.D. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 417–424.
59. Abbasi, A.; Chen, H.; Salem, A. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Trans. Inf. Syst. (TOIS)* **2008**, *26*, 12. [[CrossRef](#)]
60. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [[CrossRef](#)]
61. Gao, W.; Li, S.; Xue, Y.; Wang, M.; Zhou, G. Semi-supervised sentiment classification with self-training on feature subspaces. In Proceedings of the Workshop on Chinese Lexical Semantics, Macao, China, 9–12 June 2014; pp. 231–239.
62. Bisong, E. Introduction to Scikit-learn. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*; Springer: Berlin, Germany, 2019; pp. 215–229.
63. Settles, B. *Active Learning Literature Survey*; University of Wisconsin-Madison Department of Computer Sciences: Madison, WI, USA, 2009.
64. Borg, M.; Lennerstad, I.; Ros, R.; Bjarnason, E. On Using Active Learning and Self-training when Mining Performance Discussions on Stack Overflow. In Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, Karlskrona, Sweden, 15–16 June 2017; pp. 308–313.
65. Moreno-Seco, F.; Inesta, J.M.; de León, P.J.P.; Micó, L. Comparison of classifier fusion methods for classification in pattern recognition tasks. In Proceedings of the Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Hong Kong, China, 17–19 August 2006; pp. 705–713.
66. Blitzer, J.; Dredze, M.; Pereira, F. Biographies bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th annual meeting of the association of computational linguistics, Prague, Czech Republic, 23–30 June 2007; pp. 440–447.
67. Hofmann, M.; Klinkenberg, R. *RapidMiner: Data Mining Use Cases and Business Analytics Applications*; CRC Press: Boca Raton, FL, USA, 2013.
68. McCallum, A.; Nigam, K. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization (Vol. 752, No. 1, pp. 41–48)*; Available online: https://www.researchgate.net/publication/2408220_A_Comparison_of_Event_Models_for_Naive_Bayes_Text_Classification (accessed on 1 November 2019).
69. Lewis, D.D. Naive (Bayes) at forty: The independence assumption in information retrieval. In Proceedings of the European Conference on Machine Learning, Chemnitz, Germany, 21–23 April 1998; pp. 4–15.
70. Zhu, J.; Ma, M. Uncertainty-based active learning with instability estimation for text classification. *ACM Trans. Speech Lang. Process. (TSLP)* **2012**, *8*, 5. [[CrossRef](#)]

