Contents lists available at ScienceDirect

Machine Learning with Applications

journal homepage: www.elsevier.com/locate/mlwa

SSentiA: A Self-supervised Sentiment Analyzer for classification from unlabeled data

Salim Sazzed [*], Sampath Jayarathna

Old Dominion University, Norfolk, VA, USA

ARTICLE INFO

Dataset link: https://ai.stanford.edu/~amaas/data/sentiment/, https://figshare.com/articles/TripAdvisor_reviews_of_hotels_and_restaurants_by_gender/6255284, http://times.cs.uiuc.edu/~wang296/Data/, https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews, https://github.com/sazzadcsedu/SSentiA

ABSTRACT

In recent years, supervised machine learning (ML) methods have realized remarkable performance gains for sentiment classification utilizing labeled data. However, labeled data are usually expensive to obtain, thus, not always achievable. When annotated data are unavailable, the unsupervised tools are exercised, which still lag behind the performance of supervised ML methods by a large margin. Therefore, in this work, we focus on improving the performance of sentiment classification from unlabeled data. We present a self-supervised hybrid methodology SSentiA (Self-supervised Sentiment Analyzer) that couples an ML classifier with a lexicon-based method for sentiment classification from unlabeled data. We first introduce LRSentiA (Lexical Rule-based Sentiment Analyzer), a lexicon-based method to predict the semantic orientation of a review along with the confidence score of prediction. Utilizing the confidence scores of LRSentiA, we generate highly accurate pseudo-labels for SSentiA that incorporates a supervised ML algorithm to improve the performance of sentiment classification for less polarized and complex reviews. We compare the performances of LRSentiA and SSSentA with the existing unsupervised, lexicon-based and self-supervised methods in multiple datasets. The LRSentiA performs similarly to the existing lexicon-based methods in both binary and 3-class sentiment analysis. By combining LRSentiA with an ML classifier, the hybrid approach SSentiA attains 10%–30% improvements in macro F1 score for both binary and 3-class sentiment analysis. The results suggest that in domains where annotated data are unavailable, SSentiA can significantly improve the performance of sentiment classification. Moreover, we demonstrate that using 30%–60% annotated training data, SSentiA delivers similar performances of the fully labeled training dataset.

1. Introduction

Sentiment Analysis, also known as opinion mining, is the process of categorizing opinions expressed (e.g., *positive* or *negative*) in a text document. With the advancement of web 3.0 and escalating popularity of social media, a vast amount of user-generated content regarding products, events, services, etc., has now become available. This paradigm shift necessitates sophisticated computational tools to extract insights from these data. Due to its wide applicability in various real-word problems, sentiment analysis is a well-suited solution for pattern mining in user-generated data. Sentiment analysis has been applied in various domains such as product reviews (Fang & Zhan, 2015; Turney, 2002), restaurant reviews (Kang, Yoo, & Han, 2012), movie recommendation (Turney, 2002), drama review (Sazzed, 2020a), micro-blogs posts (Musto, Semeraro, & Polignano, 2014), election results prediction (Tumasjan, Sprenger, Sandner, & Welpe, 2010), and stock market predictions (Mittal & Goel, 2012).

Researchers have analyzed sentiments incorporated in textual data at various levels, such as aspect, sentence, or document level. In document-level sentiment analysis, the objective is to classify the opinion expressed in a document as *positive* or *negative* by considering the whole document. Sentence-level sentiment analysis determines whether a sentence conveys a *positive* or *negative* opinion. To determine the sentiment at more granular level, such as towards an entity, aspect-based sentiment analysis is used.

Both the lexicon-based (Musto et al., 2014; Sazzed, 2020b; Taboada, Brooke, Tofiloski, Voll, & Stede, 2011) and machine learning-based approaches (Gamon, 2004; Go, Bhayani, & Huang, 2009; Liu et al., 2010; Sazzed & Jayarathna, 2019; Tripathy, Agrawal, & Rath, 2016) have been investigated in various studies. Besides, researchers proposed various hybrid approaches (Appel, Chiclana, Carter, & Fujita, 2016; Malandrakis, Kazemzadeh, Potamianos, & Narayanan, 2013; Mudinas, Zhang, & Levene, 2012) combining both. Lexicon-based methods rely on linguistic resources such as sentiment lexicons composed of words and corresponding polarity values. The opinion-conveying *positive* or *negative* terms are used to evaluate polarity in a text without using any labeled data (Turney, 2002). Opinion words can express desirable (e.g., good, awesome, etc.) or undesirable (e.g., terrible, pathetic, etc.)

[*] Corresponding author.
E-mail addresses: ssazz001@odu.edu (S. Sazzed), sampath@cs.odu.edu (S. Jayarathna).

https://doi.org/10.1016/j.mlwa.2021.100026
Received 7 October 2020; Received in revised form 15 December 2020; Accepted 1 March 2021
Available online 17 March 2021

2666-8270/© 2021 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

states. Sentiment lexicons can be binary (e.g., (Hu & Liu, 2004)) such as +1 for *positive* words and -1 for *negative* words, or it can also contain words associated with the sentiment intensity score (e.g., AFINN (Nielsen, 2011), SentiWordNet (Baccianella, Esuli, & Sebastiani, 2010), and SenticNet (Cambria, Speer, Havasi, & Hussain, 2010)).

Both approaches of sentiment analysis (i.e., lexicon-based and machine learning-based) have their strengths and weaknesses. The lexicon-based methods have an advantage over supervised ML-based methods as they do not require labeled data for predicting unseen instances. The cost associated with the data labeling process makes a fully labeled training set often infeasible, whereas unlabeled data can be obtained relatively easily. Besides, the accuracy of supervised ML methods can vary across domains and can be affected by parameter tuning. However, the lexicon-based approaches have their limitations too. The explicit lexicon-based method often cannot distinguish the classes when the margin between classes is too small. The complexity and noise present in the dataset can also affect the performance of the lexicon-based system. In contrast, the supervised ML methods learn implicit patterns from the labeled data, thus show better performance in determining the polarity of complex cases.

Combining both approaches to form a hybrid classifier can increase robustness, accuracy, and overall generalization capability of sentiment classification. The integration step fuses learning-based methods to rule-based methods in both the lexicon generation and the sentiment classification stage. In the lexicon generation step, machine learning-based approaches have been applied for determining the weights of opinion words (Cambria et al., 2010; Thelwall, Buckley, Paltoglou, Cai, & Kappas, 2010). In the prediction step, the hybrid method can utilize lexical features (Melville, Gryc, & Lawrence, 2009; Pandey, Rajpoot, & Saraswat, 2017) or limited labeled data for supervised training to combine with the rule-based method (Andreevskaia & Bergler, 2008).

1.1. Motivation and research objective

Although the lexicon-based methods do not require labeled data, they yield low accuracy due to various reasons such as lexicon coverage problem, considering word-level polarity without context, etc. Therefore, over the years, a plethora of research has been conducted on hybrid sentiment analysis leveraging fully or partially labeled data set along with the polarity lexicon.

However, only a few works (He & Zhou, 2011; Qiu, Zhang, Hu, & Zhao, 2009; Tan, Wang, & Cheng, 2008; Zhang, Ghosh, Dekhil, Hsu, & Liu, 2011; Zhang & He, 2013; Zhang, Zhao, Qiu, & Hu, 2009) introduced methods that work without using any labeled data (i.e., self-supervised). Even these existing self-supervised methods have several limitations such as:

- They were applied to either very few datasets or small datasets with few thousands of samples or datasets from a single domain. Therefore, their effectiveness in large datasets or multi-domain datasets is not known.
- Their applicability was only shown for binary-level sentiment analysis.
- The correlation between the chosen threshold value used for pseudo-label selection and the accuracy of the selected pseudo-labels was not provided.

Therefore, the efficacy of the self-learning approach in unlabeled data needs to be analyzed further from several perspectives. In this research, we aim to address the missing pieces which have not been investigated in the existing study. Our research objectives are as follows:

- Introduce a method that does not require manually annotated data in the learning process.
- Show the applicability of the self-supervised approach in large datasets from multiple domains.
- Investigate the efficacy of self-supervised learning for sentiment classification at a finer-level of granularity (i.e., 3-class classification).
- Infer how to select the highly accurate pseudo-labels from a lexicon-based method to minimize error propagation into ML classifiers.

We propose a self-supervised hybrid methodology, SSentiA, which functions in a fully unsupervised manner without using any labeled data. SSentiA can be applied to both sentence-level and document-level sentiment classification (i.e, review with one or multiple sentences). We evaluate SSentiA using four large review datasets, TripAdvisor, IMDB, Amazon, and Clothing and one small dataset, Cornel movie review. The datasets are selected from multiple domains to show the efficacy of SSentiA. In addition to applying SSentiA for binary classification, we explore its performance in 3-class sentiment analysis.

SSentiA operates in two steps: first, it employs a lexicon-based classifier, LRSentiA, to generate highly accurate pseudo-labels. LRSentiA predicts the semantic orientation of a review utilizing a binary-level polarity lexicon. Additionally, it provides the confidence score of the prediction. The Chi-squared test shows a correlation between the confidence score and accuracy of the predictions of LRSentiA; hence predictions with high confidence scores are highly accurate and can be used as pseudo-label. Next, SSentiA utilizes the highly confident predictions gleaned from LRSentiA. SSentiA employs these pseudo-labels to train a supervised ML classifier and predict the semantic orientation of the low-confident reviews of LRSentiA. We demonstrate that significant improvement in sentiment classification can be achieved by employing our hybrid model in the scenario when labeled data are not available. Moreover, utilizing limited labeled data, SSentiA shows a comparable performance of a fully annotated dataset, thus reduce the time and labor needed for data annotation.

1.2. Contributions

The main contributions of our work can be summarized as follows:

- We present a hybrid sentiment analysis approach, SSentiA, for sentiment classification from unlabeled data. SSentiA utilizes pseudo-labels to train a supervised ML classifier for sentiment classification.
- We introduce a lexicon-based method LRSentiA to generate highly accurate pseudo-labels. LRSentiA provides sentiment orientation of the review as well as the confidence of predictions.
- We show an effective way to reduce the error-propagation from pseudo-label using the confidence score. We demonstrate that there is a correlation between the confidence score and the accuracy of the generated pseudo-label. Utilizing this information, we can balance between the size of the training set and error-propagation to the ML classifier.
- We provide a comparative performance evaluation of SSentiA with a number of lexicon-based, unsupervised, and self-supervised methods and demonstrate that significant performance gains can be attained by employing SSentiA.
- Besides, we show that by utilizing a small number of labeled samples, SSentiA can achieve comparable performances of the fully labeled dataset.

2. Related works

Most of the existing works related to sentiment analysis have been performed in the following two settings.

- Supervised setting: This approach utilizes labeled data to build a classification model and then infers the polarity of the unseen data.

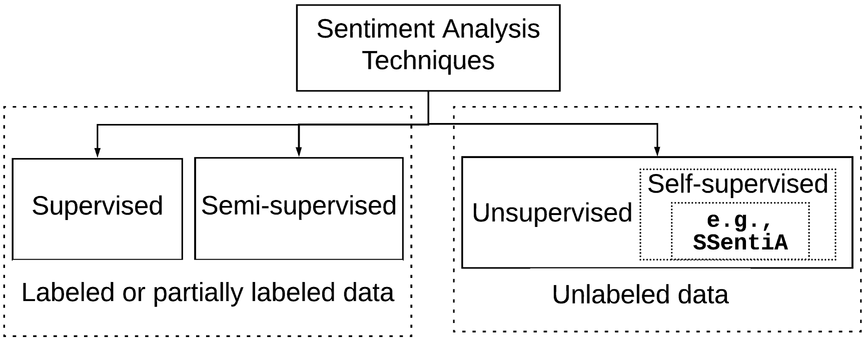


Fig. 1. Various sentiment analysis techniques.

- Unsupervised setting: This approach relies on the document's statistical properties such as word co-occurrence or the presence of sentiment words; therefore, annotated data are not required.

In recent years, two other approaches, semi-supervised and self-supervised learning, have achieved popularity.

- Semi-supervised learning falls between unsupervised learning and supervised learning. It utilizes both labeled and unlabeled data to build a classification model.
- Self-supervised learning is a subset of unsupervised learning where output labels are generated automatically by extracting patterns from data.

Our proposed methodology, SSentiA belongs to the self-supervised category, as shown in Fig. 1. For classifying sentiment in highly-polarized reviews (i.e., presence of intense feeling or emotion), it employs a lexicon-based method, LRSentiA. The predictions of LRSentiA are then used as pseudo-labels for an ML classifier, which classifies the remaining weakly-polarized reviews.

The next subsections discuss existing works belong to different settings.

2.1. Unsupervised (no labeled data) approaches

In (Taboada et al., 2011), the authors proposed lexicon-based Semantic Orientation CALculator (SO-CAL) that uses dictionaries of sentiment annotated words along with intensification and negation. Turney (2002) utilized a lexicon-based method to identify the sentiment polarity from the reviews of automobiles, banks, movies, and travel destinations. A rule-based model, VADER (Gilbert & Hutto, 2014), was introduced for analyzing sentiments from social media data. The applications of lexicon-based approaches span various domains such as Twitter (Jurek, Mulvenna, & Bi, 2015), blog (Melville et al., 2009), product reviews (Ding, Liu, & Yu, 2008), tourism (García, Gaines, Linaza, et al., 2012), etc.

Zhou, Zhao, and Zeng (2014) proposed a graph-based algorithm called graph co-regularized non-negative matrix tri-factorization (GNMTF). They assumed two words (or documents) sufficiently close to each other convey the same sentiment polarity. Utilizing the nearest neighbor graphs, they encoded the geometric information. Finally, they introduced an algorithm for learning the factorization, analyzed its complexity, and provided proof of convergence. Their experimental results on two datasets showed GNMTF provided higher accuracy compared to the state-of-the-art methods. In (Fernández-Gavilanes, Álvarez-López, Juncal-Martínez, Costa-Montenegro, & González-Castaño, 2016), the authors proposed an unsupervised dependency parsing-based text classification method for predicting sentiment in online textual messages. They utilized various natural language processing tools and derived sentiment features from sentiment lexicons. The applied their method to Cornell movie review, Obama-McCain debate, and SemEval-2015 datasets and achieved competitive performance.

In, the authors proposed an unsupervised dependency parsing-based text classification method for predicting sentiment in online textual messages. They utilized various natural language processing tools and derived sentiment features from sentiment lexicons. The applied their methods to Cornell Movie Review, Obama-McCain Debate, and SemEval-2015 datasets and achieved competitive performance.

SentiCircles (Saif, He, Fernandez, & Alani, 2016) is a lexicon-based method for classifying sentiment from Twitter data. SentiCircles considers the co-occurrence patterns of words in different contexts to capture their semantics and update the pre-assigned strength and polarity in sentiment lexicons accordingly. The authors evaluated SentiCircles on three Twitter datasets using three different sentiment lexicons. They found significant improvement over the baselines in terms of both accuracy and F-measure for tweet-level sentiment analysis. Among the three datasets, their approach performed better than the SentiStrength in two datasets.

SmartSA (Muhammad, Wiratunga, & Lothian, 2016) is a lexicon-based method that extracts the term polarity utilizing sentiment lexicons and aggregates such scores to predict the overall sentiment. As a general-purpose lexicon, the authors used SentiWordNet with genre-specific vocabulary and sentiment, as well as global and local context. When evaluated on the diverse social media data, their method showed improved performance compared to SentiStrength. Jiménez-Zafra, Martín-Valdivia, Martínez-Cámara, and Ureña-López (2016) proposed an unsupervised approach for aspect-based sentiment analysis (ABSA). They utilized a knowledge base to extract various aspects. Employing grammatical relationships and a lexicon-based approach, they classified the sentiments of various aspects. They presented a study of three well-known sentiment lexicons, Opinion Lexicon (Hu & Liu, 2004), MPQA (Wilson, Wiebe, & Hoffmann, 2005), and SentiWordNet (Baccianella et al., 2010) to determine the best combination for polarity classification at aspect-level.

Vilares, Gómez-Rodríguez, and Alonso (2017) proposed a framework for multilingual sentiment analysis utilizing compositional syntax-based rules. Their experiments showed improvement over both the existing unsupervised methods and state-of-the-art supervised models when evaluating outside their corpus of origin. Vashishtha and Susan (2019) determined the sentiments of social media posts exploiting a set of fuzzy rules and several lexicons. Their proposed fuzzy system integrates Word Sense Disambiguation (WSD) with nine fuzzy rule-based systems to classify posts into *positive*, *negative*, or *neutral* sentiment class. They applied their system into nine public twitter datasets, three sentiment lexicons and compared with four state-of-the-art unsupervised sentiment analysis approach, and one state-of-the-art supervised ML method.

2.2. Supervised or semi-supervised (labeled data) hybrid approaches

Supervised ML methods have been employed in many studies (Agarwal & Mittal, 2016; Liu et al., 2010). Researchers used ML classifiers in isolation using fully labeled dataset (Gamon, 2004; Go et al., 2009) or combined them with the other approaches using a partially or fully labeled set (Appel et al., 2016; Malandrakis et al., 2013; Mudinas et al., 2012).

Appel et al. (2016) used a sentiment lexicon enhanced with SentiWordNet (Baccianella et al., 2010) to classify sentiment in sentence-level. In (Ghiassi, Skinner, & Zimbra, 2013), the authors utilized fuzzy sets to assess both the semantic orientation and intensity to develop a Twitter-specific lexicon for sentiment analysis. SentBuk (Ortigosa, Martín, & Carro, 2014), an application for hybrid sentiment analysis in Facebook, reported accuracy of 83.27%. pSenti (Mudinas et al., 2012) is a concept-level sentiment analysis system that integrated both the lexicon-based and learning-based approaches for opinion mining. They achieved good accuracy in both sentiment polarity classification and strength detection for CNET and IMDB movie review datasets.

In (Malandrakis et al., 2013), the authors proposed a hybrid approach for sentiment analysis in Twitter data. Their model used a lexicon generated from a large corpus. They consolidated the lexicon-based model with a maximum entropy-based classifier trained on a large dataset. The two models are combined at the posterior level to generate the final output. Similarly, Xiang and Zhou (2014) proposed improvement over the Twitter sentiment analysis with the help of a topic-based mixture modeling approach with semi-supervised training. In (Prabowo & Thelwall, 2009), the authors introduced an approach that combines rule-based classification, supervised learning, and machine learning into a new hybrid model, and tested it on movie reviews, product reviews, and MySpace comments. Becker, Erhart, Skiba, and Matula (2013) proposed linear classifiers with a combination of lexical and syntactic features and automatically labeled a large Twitter dataset. They utilized the automatically labeled tweet to discover prior polarities of words and to provide additional training examples for self-training. Their found expanding the polarity lexicon and augmenting the training data with unlabeled tweets can yield performance improvement.

ALDONAr, a hybrid solution for sentence-level aspect-based sentiment analysis was proposed by Meškelė and Frasincar (2020). They used a lexicalized domain ontology and a neural attention model. Their manually created lexicalized domain ontology is integrated to utilize the domain-specific knowledge. ALDONAr uses BERT word embeddings, regularization, Adam optimizer, and different model initialization.

Schouten and Frasincar (2018) presented a hybrid approach for aspect-based sentiment analysis. They proposed a knowledge-driven approach that complements traditional machine learning methods. By using domain knowledge encoded in an ontology, they improved the sentiment analysis of a given aspect. Cai et al. (2019) constructed a domain-specific three-layered sentiment dictionary with entities, aspects and sentiment words. They employed a stacking approach to combine SVM and GBDT and achieved a better performance than baseline single models. da Silva, Coletta, Hruschka, and Hruschka (2016) combined Support Vector Machines (SVM), constructed from labeled data, with the information provided by the pair-wise similarities between unlabeled data points. Their proposed framework is based on an iterative self-training approach. Their results show that the use of unlabeled tweets improves classification performance when a few labeled tweets are available.

Lee and Kim (2017) proposed a sentiment labeling approach with a joint sentiment/topic model (JST). Their semi-supervised sentiment classification framework adds pseudo-labeled instances to the training corpus by filtering confidently predicted instances. To exploit the sufficient number of unlabeled instances, they conducted self-training with a concatenated vector that complements the document and polarity vectors.

SAIL (Malandrakis et al., 2013) model uses a lexicon automatically generated from a very large web corpus. The word and bigram affective ratings were calculated and used as features of a Naive Bayes (NB) tree model. In the unconstrained scenario, the authors combined the lexicon-based model with a classifier built on maximum entropy language models and trained on a large external dataset. The two models were fused at the posterior level to produce a final output. Their approach performed well in Twitter sentiment analysis in both constrained and unconstrained scenarios.

Giatsoglou et al. (2017) proposed a hybrid approach for the prediction of sentiment by combining the context-sensitive Word2Vec with the sentiment lexicon. The resulting hybrid representations are then used as inputs for the supervised training of a classifier. They tested their approach in four different text corpora in Greek and English, along with different coding schemes and different classifier models. They found the SVM model with a linear kernel achieved the best results in terms of efficiency in accuracy and the process times.

Araque, Corcuera-Platas, Sánchez-Rada, and Iglesias (2017) presented a methodology for sentiment analysis that performs surface and deep features integration. Their ensemble techniques combined several sentiment classifiers trained with different kinds of features. They utilized six datasets from two domains: Twitter and movie reviews.

Yu, Wang, Lai, and Zhang (2017) proposed a word vector refinement model to improve the existing word embeddings such as Word2vec and GloVe. Their proposed approach adjusts the vector representations of words so that semantically and sentimentally similar words come closer. They applied the refined word-embedding with CNN, LSTM, and DNN in Stanford Sentiment Treebank (SST) and achieved improvement over conventional word-embeddings in both binary and fine-grained sentiment classification.

2.3. *Self-supervised approaches (pseudo-labeled data)*

Zhang et al. (2011) presented an entity-level sentiment analysis method for Twitter. Their method first adopt a lexicon-based approach to perform sentiment analysis. They found that the lexicon-based method provided high precision, but low recall. To improve recall, they identified additional tweets that are likely to be opinionated by the lexicon-based method. A classifier is then trained to assign polarities to the entities in the newly identified tweets. Instead of being labeled manually, the training examples are given by the lexicon-based approach. Their experimental results showed that the proposed method improved the F-score and outperformed the state-of-the-art baselines.

He and Zhou (2011) proposed a framework where an initial classifier is learned by incorporating a sentiment lexicon and using generalized expectation criteria. They utilized documents classified with high confidence as pseudo-labeled examples for automatical domain-specific feature acquisition. The word-class distributions of self-learned features are estimated from the pseudo-labeled examples. They trained another classifier by constraining the model's predictions on unlabeled instances. Their framework was evaluated on two small datasets, the movie-review (MR) dataset and the multi-domain sentiment (MDS) dataset (each contains 2000 reviews), and attained comparable performance with other weakly-supervised sentiment classification methods.

SESS (SElf-Supervised and Syntax-Based method) was proposed by (Zhang et al., 2009) that consists of three phases. In the first phase, some documents are classified based on a sentiment dictionary, and then the sentiments of phrases and documents are iteratively revised. In the second phase, a machine learning model is trained with the labeled data from the first phase. In the third phase, for the final classification, the learned model is applied to the whole data set. Their datasets span four domains, where each domain contains 1000 *positive* and 1000 *negative* documents.

Zhang and He (2013) introduced a weakly-supervised approach for Chinese sentiment classification. They applied a variant of a self-training algorithm to train an initial classifier. Later they utilized a pseudo-labeled training set and adopted a standard self-learning cycle to obtain the overall classification results.

Qiu et al. (2009) proposed SELC Model for the sentiment classification in Chinese. The SELC Model is comprised of two phases; In the first

Table 1
The description of existing self-supervised hybrid approaches.

Method	Language	#Class	Granularity	# Samples	Lexicon	ML Classifier
(Zhang et al., 2011)	English	2	Entity-level	2500	Opinion Lexicon	SVM
(He & Zhou, 2011)	English	2	Document	4000	MPQA	GE
(Zhang et al., 2009)	English	2	Document	8000	MPQA	NB
(Tan et al., 2008)	Chinese	2	Document	4356	NTUSD	Centroid Classifier
(Zhang & He, 2013)	Chinese	2	Document	23203	HowNet	SVM
(Qiu et al., 2009)	Chinese	2	Document	7,779	HowNet	SVM
SSentiA	English	2/3	Document/Sentence	95150/54800	Opinion Lexicon	SVM/LR

phase, a sentiment dictionary is used to classify reviews. In the second phase, a supervised classifier is trained by utilizing some of the reviews predicted in the first phase. Then the supervised classifier predicts other reviews and improves the results produced in the first phase. They applied the SELC model to a dataset of 7779 Chinese product reviews and achieved an improvement of 6.63% in F1 score over the previous best result.

2.4. Comparison of SSentiA and existing self-supervised methods

Table 1 shows the details of the existing self-supervised methods and SSentiA. SSentiA differs from the existing self-supervised methods in several perspectives. The key methodological differences exist in the following aspects: pseudo-label generation and selection, training–testing data splitting, the granularity of classification, and sentiment lexicon used. In the evaluation phase, the differences come from the assessment of SSentiA on large datasets and classification at both binary and ternary levels.

2.4.1. Selection of pseudo-label and training–testing data

He and Zhou (2011) inferred the review class labels using polarity lexicon and generalized expectation criteria. Using the threshold of class prediction probability and information gain, they extracted top features for training the ML classifier. They used different testing set for the evaluation. In SELC, Qiu et al. (2009) utilized a sentiment dictionary for iteratively classifying unlabeled data and extracting sentiment features. The vocabulary and classified reviews are updated and enlarged gradually in the next steps. They adopted a balanced subset of top k *positive* and *negative* reviews as pseudo-labeled training data for ML algorithms. The reviews having zero polarity scores are used as testing data. Zhang et al. (2011) first extracted opinion indicators from the tweets. Then they determine whether a tweet is opinionated based on the indicators in the context. They used all *positive* and *negative* opinion tweets as training data. Zhang et al. (2009) used a predefined mean value for pseudo-label selection in SESS. They used 61.8% of the classified data from the lexicon-based method (first phase) as training data for phase-2 ML algorithms. Tan et al. (2008) used top n/2 examples of *positive* and *negative* predictions of the lexicon-based classifier, where n is a predefined number. They employed a very simple approach such as counting *positive* and *negative* words for the initial prediction.

Our proposed methodology, SSentiA, utilizes the a lexicon-based classifier LRSentiA to generate accurate pseudo-labels. We compute the confidence score of predictions of LRSentiA and categorize them into multiple groups. We find that the prediction accuracy of various confidence groups differs. We utilize the predicted reviews of the highly accurate groups as pseudo-labeled training data for ML classifiers while the remaining are used as testing data.

3. Dataset

We use several publicly available review datasets: TripAdvisor,[1] (Thelwall, 2018) IMDB,[2] (Maas et al., 2011) Amazon,[3] (Wang, Lu, & Zhai, 2011), Clothing[4] and Cornel movie review.[5] The TripAdvisor hotel dataset consists of 23600 reviews with 5 different ratings (−4, −2, 0, 2, 4). We consider reviews with a rating below 0 as *negative*, above 0 as *positive*, and 0 as *neutral*. At binary-level, the dataset uses *positive* and *negative* reviews, while the 3-class dataset includes *neutral* reviews as well. The IMDB dataset is originally a binary-level dataset with 12500 *positive* and 12500 *negative* reviews; therefore, we use it only for binary-level classification. The Amazon MP3 dataset consists of user reviews with ratings between 1 to 5. To categorize the reviews into the binary level, we consider rating 1–2 as a *negative* and 4–5 as *positive* (Maas et al., 2011; Pang & Lee, 2004, 2005). This binary dataset contains a total of 28469 customer reviews, 21987 *positive* and 6482 *negative*. The 3-class dataset includes additional 2531 *neutral* reviews with ratings of 3 (Mudinas et al., 2012; Pang & Lee, 2004, 2005). The binary-level Clothing dataset comprised of 18539 *positive* reviews and 4101 *negative* reviews, totaling 22640 reviews. Although The Cornel movie review dataset is very small, consists of only 1000 *positive* and 1000 *negative* reviews, we use it as some of the state-of-the-art methods reported their results on this dataset, therefore, allow us to compare the results. Among the five binary datasets, IMDB, TripAdvisor, and Cornel datasets are class-balanced, while other datasets contain mostly *positive* reviews. None of the 3-class datasets is class-balanced (see Table 2 and Table 3).

Table 2
The description of 2-class datasets.

	Domain	Negative	Positive	Total
TripAdvisor	Hotel	9520	9520	19040
IMDB	Movie	12500	12500	25000
Amazon	MP3	6482	21987	28469
Clothing	Garment	4101	18540	22641
Cornel	Movie	1000	1000	2000

Table 3
The description of 3-class datasets.

	Negative	Neutral	Positive	Total
TripAdvisor	9520	4760	9520	23800
Amazon	6482	2531	21987	31000

To convert the 5-point scale system (i.e., reviews with a rating between 1 and 5) to a ternary dataset, we follow the class-labeling procedure of existing literature. Maas et al. (2011), Mudinas et al. (2012), Pang and Lee (2004). Based on Mudinas et al. (2012), Pang and Lee (2004), in a 5-points scale system, a rating of 1 or 2 is considered as *negative*, 3 as *neutral*, and 4 or 5 as *positive*.

4. Baseline comparison

We compare the performance of SSentiA with the existing tools and classifiers in two different settings.

1. Without using any labeled data (which is the main focus of this work).

[1] https://figshare.com/articles/TripAdvisor_reviews_of_hotels_and_restaurants_by_gender/6255284.
[2] https://ai.stanford.edu/~amaas/data/sentiment/.
[3] http://times.cs.uiuc.edu/~wang296/Data/.
[4] https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews.
[5] http://www.cs.cornell.edu/people/pabo/movie-review-data/.

2. Utilizing limited labeled data

Although we show that SSentiA can be utilized as a semi-supervised tool using limited labeled data, the main focus of this study is to show the efficacy of SSentiA for classifying sentiment in unlabeled data.

4.1. Setting 1: Without using any labeled data

We compare both LRSentiA and SSentiA with a number of state-of-the-art lexicon-based, unsupervised and self-supervised methodologies that do not use any labeled data.

4.1.1. Comparison with fully lexicon-based methods

Some of the state-of-the-art lexicon-based sentiment analysis tools and lexicons: AFINN (Nielsen, 2011), SentiStrength (Thelwall et al., 2010) , VADER (Gilbert & Hutto, 2014), TextBlob[6] and Opinion-Finder (Wilson et al., 2005) are utilized. These lexicon-based tools are capable of classifying sentiments from unlabeled data. They are selected based on their performances on a number of benchmark comparisons (Ahmed Abbasi & Dhar, 2014; Ribeiro, Araújo, Gonçalves, Gonçalves, & Benevenuto, 2016).

The AFINN (Nielsen, 2011) lexicon consists of a list of English terms manually rated for valence by Finn Årup Nielsen. Each term is represented by an integer between −5 (strongly *negative*) and +5 (strongly *positive*). We utilize the PyPI implementation of AFINN[7] to compute the polarity of the review. For binary classification, a *non-negative* polarity score indicates a *positive* class. For 3-class, a polarity score above 0 refers to *positive* prediction, below 0 refers to *negative* prediction, and a 0 polarity score means *neutral* class prediction.

SentiStrength (Thelwall et al., 2010) is a sentiment analysis tool that employs multiple methods to simultaneously extract *positive* and *negative* sentiment strength from the short informal text. It uses a dictionary of sentiment words with associated polarity strength. SentiStrength utilizes a dual 5-point system for *positive* and *negative* sentiment and leverages machine learning methods to optimize sentiment term weightings. The predicted class of a review is determined following the same procedure of AFINN. We utilize the PyPI SentiStrength package.[8]

VADER (Valence Aware Dictionary and sEntiment Reasoner) (Gilbert & Hutto, 2014) is a lexicon and rule-based sentiment analysis tool specifically attuned to sentiments expressed in social media. VADER combines the lexical features with five generalizable rules to classify reviews at the document level. For the binary classification, a *non-negative* compound score refers to a *positive* prediction. For 3-class prediction, we consider a compound score greater than 0.05 as *positive*, a score less than −0.05 as *negative*, and a score between −0.05 and 0.05 as neutral. The ranges of the compound score for different classes are selected based on the original VADER paper. We utilize the PyPI VADER package.[9]

TextBlob[10] is a Python library for processing textual data. It provides an API for various NLP tasks such as part-of-speech tagging, noun phrase extraction, and sentiment analysis. TextBlob calculates both the polarity and subjectivity score from the text reviews utilizing opinion word's weighted scores. At the binary-level, a *non-negative* polarity score refers to *positive* prediction. When 3-class is considered, a polarity score above 0.05 indicates the *positive* class, below −0.05 refers to the *negative* class, and remaining attributes to the *neutral* class.

OpinionFinder is a framework that (Wilson et al., 2005) can identify the subjectivity and polarity of a text. The OpinionFinder polarity classifier utilizes a prior word-level polarity score of *positive*, *negative*, or *neutral* to classify text. The class prediction of a review is determined following the same procedure of AFINN and SentiStrength.

4.1.2. Unsupervised/self-supervised methods

Besides, we provide comparative performances with some of the state-of-the-art unsupervised methodologies Fernández-Gavilanes et al. (2016), Vashishtha and Susan (2020), Vilares et al. (2017), Zhou et al. (2014) and self-supervised He and Zhou (2011) in a benchmark dataset (i.e., Cornel movie review). Few other existing self-supervised methodologies do not have their source code or datasets publicly available Zhang et al. (2011), Zhang and He (2013); Besides, some of them has been applied to a non-English language (i.e., Chinese) Qiu et al. (2009), Tan et al. (2008), Zhang and He (2013). Therefore, we could not compare their performances with SSentiA.

4.2. Setting 2: With limited labeled data

In setting 2, we show that SSentiA utilizing limited labeled data can yield similar performance to a fully labeled training dataset. This comparison is performed to exhibit the efficacy of SSentiA for minimizing the necessity of labeled data. Two supervised ML classifiers, Logistic Regression (LR) and Support Vector Machine (SVM) are used for the comparison.

5. Methodology

5.1. LRSentiA: Proposed Lexical Rule-based Approach

LRSentiA is a lexicon and rule-based method that can classify sentiment without using any labeled data. The main purpose of introducing LRSentiA is to generate accurate pseudo-labels so that supervised ML classifiers can be incorporated into SSentiA. LRSentiA utilizes a binary-level sentiment lexicon and set of rules to predict the polarity of a review. Besides predicting the class, LRSentiA also provides the confidence score of the prediction. The steps of LRSentiA are shown in Fig. 2.

5.1.1. Opinion words extraction

Not every word of a sentence constitutes an opinion. It is imperative to use only the subjective words which convey sentiment. Identifying and excluding objective text from reviews could significantly improve sentiment detection performance (Liu, 2012). In our work, we exclude words that do not convey any opinion or do not have any influence on the polarity of the sentence. We apply a dependency parser to extract grammatical structure and to identify the relationships among words. In addition to the POS tagger, a dependency parser can be used to eliminate words that do not affect sentence polarity. We use the spaCy library (Honnibal & Montani, 2017) to determine both the POS tag and the relationship among words in a sentence.

5.1.2. Sentiment lexicon

To determine the polarity of a review, a sentiment lexicon composed of lexicon units such as words and their sentiment orientations is required. The efficiency of a lexicon-based method largely depends on the coverage of opinion words in the lexicon. Some of the popular lexica for sentiment analysis are opinion lexicon (Hu & Liu, 2004), MPQA (Wilson, Wiebe, & Hoffmann, 2009), SentiWordNet (Baccianella et al., 2010), and SenticNet (Cambria et al., 2010).

We utilize the opinion lexicon curated by Hu and Liu (Hu & Liu, 2004), which consists of 4783 *negative* and 2006 *positive* words with the polarity score of either −1(*negative*) or +1(*positive*). This lexicon contains the following POS: nouns, verbs, adjectives, and adverbs. It also includes misspellings, morphological variants, slang, and social-media mark-up.

[6] https://textblob.readthedocs.io/en/dev/.

[7] https://pypi.org/project/afinn/.

[8] https://pypi.org/project/sentistrength/.

[9] https://pypi.org/project/vaderSentiment/.

[10] https://textblob.readthedocs.io/en/dev/.

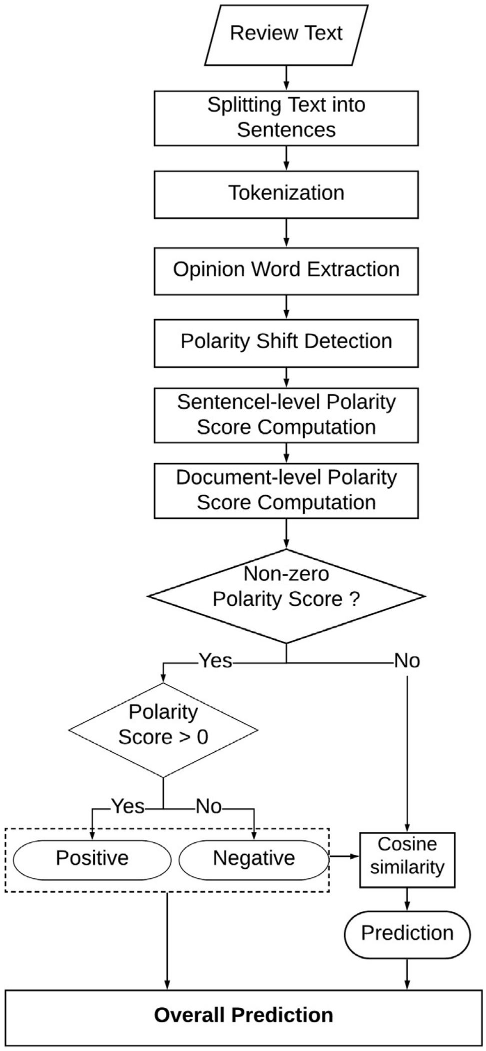


Fig. 2. Steps of LRSentiA for binary-level sentiment classification.

5.1.3. Negation and polarity shifter

Identifying negation (e.g., not, no, never) in a sentence is essential for sentiment analysis since it changes the sentiment orientations. Using spaCy (Honnibal & Montani, 2017) dependency parser, we detect the negation word and successive opinion conveying term. We alternate the polarity of the opinion word that follows the negation term. Modal verbs are auxiliary verbs that express necessity or possibility. Modal verbs in English include words such as 'must', 'should', 'would', 'can', 'could', 'may', 'might', etc. As a modal verb can change the polarity of an opinion conveying word in a sentence, it is necessary to identify them. For example- '*The food could have been better*' conveys a *negative* opinion. Though the modal verb 'could' does not convey any opinion, it changes the sentence polarity.

5.1.4. Sentence-level polarity calculation

After identifying opinion words, negations, and polarity shifters in a sentence, we calculate the *positive* and *negative* polarity score of each sentence by adding up the corresponding word-level polarity score defined by the sentiment lexicon (i.e., +1 for *positive* term and −1 for *negative* term). In the presence of the negation or polarity shifter, the word-level polarity is reversed. For a sentence s with *positive* polarity score of $P_{pos}(s)$ and *negative* polarity score of $P_{neg}(s)$, the overall polarity score $P(s)$ is determined by $P_{pos}(s) + P_{neg}(s)$.

5.1.5. Aggregation and document-level class assignment

2-class dataset: As most of the reviews are comprised of multiple sentences, it is necessary to combine the polarity scores obtained from individual sentences. In the aggregation step, we combine the polarity scores of each sentence to determine the overall sentiment polarity of a review. The overall polarity of a review is determined by the cumulative sum of the polarity scores of each sentence. If the review r consists of n sentences, $s_1, s_2, \ldots, s_n$ and corresponding polarity scores are $P(s_1), P(s_2), \ldots, P(s_n)$, then the overall polarity score is calculated as $P(r) = \sum_{i=1}^{n} P(s_i)$. The semantic orientation of review r is determined by,

$$Sentiment\ of\ review(r) = \begin{cases} positive, & \text{if } P(r) > 0 \\ negative, & \text{if } P(r) < 0 \\ tie, & Otherwise \end{cases}$$

If the overall polarity score of a review is greater than 0, it is considered as a *positive* review. if it is less than 0, then it is considered as a *negative* review. If the review score is 0, we consider it as a tie.

To determine the labels of tie cases, we utilize the cosine similarity score. To find the most similar reviews of review i, We first compute its average similarity score $avgSim(i)$ and standard deviation $std(i)$ considering all the other reviews in the same dataset. Then the threshold similarity value is calculated by $simThresh(i) = avgSim(i) + a * std(i)$, where $avgSim(i)$ represents the average similarity score of review i and $std(i)$ represents standard deviation. The value of a is set heuristically, we find $a = 3$ yields good results in all the datasets. We assign the class label of review i based on the similar reviews having a similarity score above the $simThresh(i)$. If there are m similar reviews, where j = $1, 2, \ldots, m$ and similarity score of between i and j is $sim(i, j)$ and class label of j is $c(j)$, then class score of i, $classScore(i)$ is obtained using, $classScore(i) = \sum_{j=1}^{m} sim(i, j) * c(j)$. if the $classScore(i)$ is non-*negative*, then review i is predicted as *positive*, else it is assigned to the *negative* class.

3-class dataset: For 3-class datasets, we follow a very similar aggregation step of binary classification; only the class assignment procedure differs. If the polarity score of a review is positive with a confidence score (defined in the next section) above 0.10, we assign it to the *positive* class, while a negative polarity score with a confidence score above 0.10 is assigned to the *negative* class.

As stated above, in addition to the polarity score, the confidence score is utilized for a 3-class label assignment. A value of 0.10 is used as a threshold for the confidence score to distinguish between the *positive* or *negative* class and the *neutral* class. The threshold is selected based on the following assumptions — in the 3-class dataset, in a perfect case, reviews belong to the *neutral* class should have a polarity score of 0, a positive polarity score for the positive class, and a negative polarity score for reviews belong to *negative* class. However, this assumption is too stringent and not practical due to various reasons, such as lexicon coverage, binary polarity score (i.e., −1, +1) of opinion words, etc. Therefore, an additional constrain of confidence score (which actually refers to the polarity strength) of 0.10 is imposed to address the above-mentioned issue. We mark prediction as *neutral* when either $P(r)$ is 0 or confidence score is less than 0.10.

5.1.6. Prediction confidence

In addition of predicting the class of a review, LRSentiA provides the confidence score of the prediction. We utilize the ratio of *positive* and *negative* polarity scores obtained from a review to determine the confidence score of the prediction. If the review r consists of n sentences, $s_1, s_2, \ldots, s_n$ with *positive* polarity scores of $P_{pos}(s_1), P_{pos}(s_2), \ldots, P_{pos}(s_n)$ and *negative* polarity scores of $P_{neg}(s_1)$, $P_{neg}(s_2), \ldots, P_{neg}(s_n)$, then overall *positive* polarity score of review r is calculated as $P_{pos}(r) = \sum_{i=1}^{n} P_{pos}(s_i)$ and *negative* polarity score is calculated as $P_{neg}(r) = \sum_{i=1}^{n} P_{neg}(s_i)$. The confidence score of the review r is determined by -

$$ConfScore(r) = \frac{abs(P_{pos}(r) + P_{neg}(r))}{abs(P_{pos}(r)) + abs(P_{neg}(r))}$$

5.2. SSentiA: Proposed self-supervised hybrid approach

The proposed self-supervised methodology, SSentiA, which is the main focus of this work, integrates supervised ML classifiers with the lexicon-based method LRSentiA. ML classifiers are capable of capturing the implicit pattern of data; therefore, they can enhance the classification performance on the complex and less polarized reviews that the lexicon-based methods like LRSentiA cannot distinguish well. However, the supervised ML classifier requires annotated reviews, which are not always available. Hence, we utilize LRSentiA to automatically generate pseudo-labels for the ML classifier of SSentiA.

SSentiA utilizes the confidence scores and predictions of LRSentiA to generate highly accurate pseudo-labels. By leveraging these pseudo-labels, a supervised ML algorithm is incorporated into SSentiA. We employ several ML classifiers: Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and Extra Trees (ET) on reviews conveying weak or ambivalent opinion. We select the best performing ML classifiers and integrate them into SSentiA.

We utilize the scikit-learn (Pedregosa et al., 2011) implementation (ver.0.23.1) of the above-mentioned classifiers. For all the classifiers, the default parameter settings with the class-balanced weights are used. As an input feature for the ML classifiers, unigram, and bigram-based tf–idf (term frequency–inverse document frequency) scores are used. Fig. 3 shows various steps of SSentiA for sentiment classification.

5.2.1. Generation of pseudo-labels

We categorize the predictions of LRSentiA into multiple confidence groups based on their confidence scores *ConfScore*.

2-class dataset: In each dataset, we calculate the mean confidence score *(mcs)* and standard deviation (*std*) across all the predictions to find the threshold *thr* value. The *thr* value is used to determine various confidence groups, which is calculated as, *thr* = *mcs* + 0.5 * *std*. If the *thr* value is less than 0.5, we use 0.5 as a *thr* value.

The confidence group of review r, *confGroup(r)* is determined as follows,

$$confGroup(r) = \begin{cases} very\text{-}high, & \text{if } ConfScore(r) \geq thr \\ high, & \text{if } ConfScore(r) \geq thr - 0.5 * std \\ low, & \text{if } ConfScore(r) \geq thr - std \\ very\text{-}low, & \text{if } ConfScore(r) > 0 \\ zero, & \text{otherwise} \end{cases}$$

The predicted reviews with a confidence score above the *thr* fall into *very-high* confidence group. The next two categories (i.e., *high* and *low*) contain predictions whose confidence scores are 0.5 and 1.0 standard deviations (*std*) below the *thr* value, respectively. The *very-low* category contains predictions having pos confidence score and falls below *low* category.

Three criteria are considered while categorizing predictions into multiple groups that are described below-

a. Minimize the inclusion of wrong prediction (i.e., highly accurate pseudo-label) into few groups so that they can be used as training data for ML classifier with minimal error propagation.
b. Maximize the number of reviews (larger training set) utilized as pseudo-labels for ML classifier respecting criteria (a)
c. Show the correlation between confidence score and the accuracy (i.e., high confidence score implies high accuracy).

(a) and (b) both are important for having good performance from machine learning (ML) classifier, as (a) highly-accurate pseudo-label means less error-propagation to ML classifier and (b) higher number of pseudo-label means of the larger training set, that is needed to have good accuracy from machine learning model. (c) is important for group selection, (c) determines which groups should be used as training data and which ones to use as testing data.

We find discretizing the predictions of reviews into five categories fulfills the above criteria best; therefore, five groups are used.

3-class dataset: In 3-class sentiment analysis, predictions are grouped in a different way to incorporate the *neutral* class. The *positive* and *negative* predictions with a confidence score above 0.75 are placed to *very-high* confidence group, while confidence score above 0.5 are placed to *high* confidence groups. Predictions with a confidence score between 0.1 and 0.5 are considered *low* category predictions. The remaining reviews with confidence scores less than 0.1 are considered *neutral* class predictions.

The objective of distinguishing highly confident ((i.e., *very-high* and *high*)) predictions is to generate labels for supervised ML algorithms. When the rule-based classifier predicts the polarity of a review with high confidence (i.e., high *positive/negative* score), it is highly probable that the prediction is correct. As our rule-based method, LRSentiA, relies on the sentiment of individual opinion words, if the overall polarity score is very *positive* or very *negative*, then the review consists of mostly *positive* aspects (very *positive* score) or *negative* aspects (very *negative* score); thus the prediction is correct.

5.2.2. Utilizing pseudo-label

After identifying highly confident predictions (i.e., *very-high* and *high* confidence groups) of LRSentiA, we utilize them as pseudo-labeled training data for the supervised ML classifier.

2-class dataset:

As shown in Fig. 3, utilizing supervised ML algorithms and highly accurate pseudo-labeled training data (i.e., *very-high and high* confidence groups of LRSentiA), the *low* confidence prediction group is classified as either *positive* and *negative*. In the next phase, *very-low* and *zero* confidence groups are considered as testing data and predictions from *very-high*, *high*, *low* groups are employed as training data.

3-class dataset:

To train supervised ML classifiers for ternary classification, we use *very-high* and *high* confidence groups of *positive* and *negative* predictions as training data. As *neutral* class training data, we utilize predicted reviews having 0 polarity scores and contain at least 5 *positive* and *negative* terms.

As mentioned earlier, we want the pseudo-labels used in the training data as much accurate as possible. We observe that having a polarity score of 0 assigned by the lexicon-based method does not necessarily mean a review is *neutral*, as it could be due to the lexicon-coverage problem. For example, a review with 2 *positive* and 0 *negative* terms may yield a 0 polarity score if the *positive* terms do not exist in the lexicon and erroneously can be assigned to *neutral* class. Therefore, we impose an additional constraint to overcome the lexicon-coverage problem.

The minimum number of opinion terms required for a pseudo-labeled *neutral* class review is set to 5. The rationale behind that if the lexicon-based method identifies a high number (i.e., 5) of polarity terms in a review and still provides a total polarity score of 0 (due to the presence of both positive and negative terms), probably it is a *neutral* review. The remaining predictions from all the classes (*positive*, *negative*, and *neutral*) are considered as testing data for the supervised ML classifiers.

The remaining predictions from all the classes (*positive*, *negative*, and *neutral*) are considered as testing data for the supervised ML classifiers.

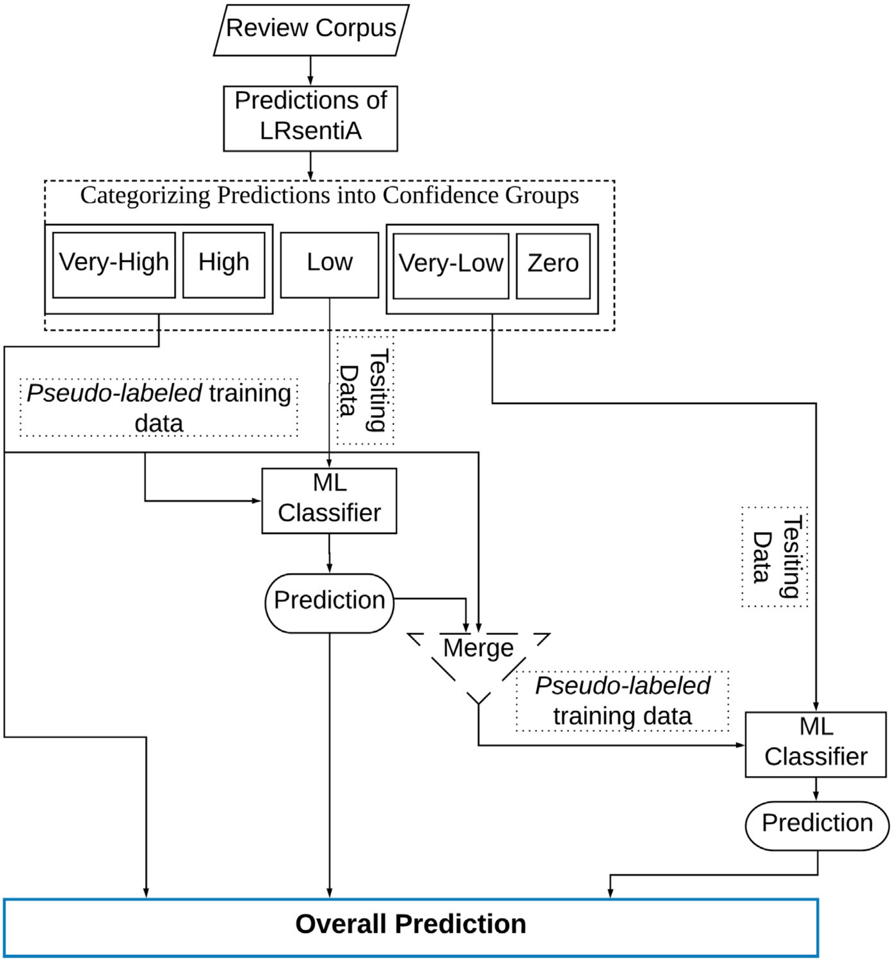


Fig. 3. Steps of SSentiA for binary sentiment classification.

Table 4
The mean confidence score (mcs), standard deviation (std) and threshold (thr) value of the predictions of LRSentiA in 4 binary datasets.

Dataset	mcs	std	thr
TripAdvisor	0.556	0.343	0.727
IMDB	0.352	0.265	0.500
Amazon	0.339	0.348	0.513
Clothing	0.464	0.354	0.641

6. Results

To assess the performances of various classifiers, we calculate the accuracy, precision, recall, F1 score, and Matthews Correlation Coefficient (MCC) score. The accuracy score is biased towards the dominating classes. As most of our datasets are class-imbalanced, the MCC and F1 scores are better metrics to evaluate the performances of the classifiers.

Table 4 provides the mean confidence scores (mcs), standard deviations (std), and threshold (thr) scores of the predictions of LRSentiA in 4 binary datasets. We use these values to categorize predictions into various confidence groups.

Table 5 shows the accuracy and F1 score of four confidence categories of LRSentiA in various datasets. The results suggest a correlation between the confidence level of prediction and accuracy.

To show the correlation between the confidence score and accuracy of the predictions of LRSentiA, we conduct the Chi-squared test for independence, which indicates whether two categorical values are dependent. We consider four categories, *very-high*, *high*, *low*, and *very-low*. The null hypothesis assumes the confidences groups and their corresponding accuracy are independent, while the alternative hypothesis suggests they are not. The threshold value for the significant level is selected as 0.05. Using the Chi-squared test, we find the p-values $<$ 0.05 for all datasets, which reject the null hypothesis and accept the alternative hypothesis.

As shown in Table 5, LRSentiA performs poorly in low confidence groups such as *low, very-low, etc.* Hence, to improve the classification performance on reviews belong to these categories, we employ supervised ML classifiers. However, since no labeled data are available, we utilize the predictions of two high confidence groups (*very-high and high*) of LRSentiA as pseudo-labeled training data.

Table 6 shows employing ML classifiers with pseudo-labels can enhance the accuracy of the low confidence groups of LRSentiA. In all the four datasets, we observe accuracy improvement between 25%–28% in low confidence groups.

Table 7 presents the comparative performance of proposed LRSentiA and SSentiA with the baseline sentiment analysis tools in four different datasets. In the IMDB dataset, LRSentiA provides an F1-score of 0.736, where VADER shows 0.69. The best F1-score is obtained using SSentiA with SVM, which is 0.807. Considering the accuracy, SSentiA with LR shows the highest accuracy of 80.63%, while LRSentiA and VADER show the accuracy of 73.62% and 67.6%, respectively. In the Amazon dataset, LRSentiA provides an F1 score of 0.719, while among the existing tools, VADER shows the highest F1 score of 0.724. The best F1 score is obtained using the proposed hybrid method using LR, which is 0.808. In the Clothing dataset, LRSentiA provides an F1 score of 0.665; Among the other existing lexicon-based tools, VADER shows the highest

Table 5
The comparison of F1 scores and accuracies in four confidence groups of LRSentiA of binary datasets.

Dataset	Group	ConfScore	F1 Score	MCC	Acc.(%)	#Correct/Total
TripAdvisor	*very-high*	0.727 - 1.0	0.922	0.844	95.33%	6391/6706
	high	0.556 - 0.727	0.893	0.786	90.26%	2345/2598
	low	0.384 - 0.556	0.831	0.662	80.29%	2144/2670
	very-low	(>0) - 0.384	0.679	0.353	61.67%	3498/5672
IMDB	*very-high*	0.500 - 1.0	0.878	0.757	88.04%	6290/7144
	high	0.367 - 0.500	0.826	0.653	82.71%	2460/2974
	low	0.234 - 0.367	0.735	0.471	73.59%	3564/4843
	very-low	(>0) - 0.234	0.610	0.220	61.06%	4979/8153
Amazon	*very-high*	0.513 - 1.0	0.808	0.617	91.04%	11425/12549
	high	0.339 - 0.513	0.778	0.557	86.41%	3652/4226
	low	0.166 - 0.339	0.691	0.383	74.28%	4645/6253
	very-low	(>0) - 0.166	0.585	0.171	61.36%	1563/2547
Clothing	*very-high*	0.641 - 1.0	0.720	0.431	91.38%	10755/11769
	high	0.464 - 0.641	0.675	0.342	85.30%	3041/3565
	low	0.288 - 0.464	0.660	0.320	75.40%	2376/3151
	very-low	(>0) - 0.288	0.597	0.193	66.48%	1335/2008

Table 6
The performance improvement in three low-confidence predictions groups of LRSentiA (i.e., *low, very-low, and zero*) utilizing ML classifiers and pseudo-labels from LRSentiA (*very-high and high confidence groups*).

Dataset	Method	F1 Score	Accuracy	[a]Improvement
TripAdvisor	LRSentiA	0.696	60.12%	NA
	RF	0.677	67.86%	−0.027%
	ET	0.696	60.53%	0.0%
	LR	0.849	87.77%	21.98%
	SVM	0.883	90.72%	26.86%
IMDB	LRSentiA	0.635	63.26%	NA
	RF	0.677	65.03%	7.27%
	ETs	0.692	69.3%	8.97%
	LR	0.778	77.61%	22.51%
	SVM	0.794	79.37%	25.03%
Amazon	LRSentiA	0.630	69.61%	NA
	RF	0.673	73.09%	6.82%
	ET	0.678	73.47%	7.61%
	LR	0.798	81.00%	26.66%
	SVM	0.80	80.59%	26.98%
Clothing	LRSentiA	0.593	68.80%	NA
	RF	0.597	69.22%	0.67%
	ET	0.600	69.37%	1.18%
	SVM	0.745	78.19%	25.63%
	LR	0.760	78.10%	28.16%

[a]F1 score improvement.

F1-score of 0.666. The best F1-score is obtained using the SSentiA and LR, which is 0.774.

We conduct the McNemar test (McNemar, 1947) to determine whether there exists a significant difference between the performances of the lexicon-based methods and SSentiA. The McNemar test determines whether two experimental results disagree with each other by comparing their sensitivity and specificity on the same data. The null hypothesis claims that the two results are the same, while the alternative hypothesis indicates the opposite.

For each dataset, the best performing lexicon-based method (prediction-1) is compared with SSentiA (prediction-2) using a 2 X 2 contingency table. A value of 0.05 is used as a level of significance. For all the four binary datasets, we find p-values less than 0.005, which infer significant differences between the performance of the best lexicon-based method and SSentiA.

Table 8 shows the comparison between SSentiA and several unsupervised and self-supervised methodologies in the Cornel movie review dataset. The results indicate that SSentiA performs better than all of them by some margin. When SVM is integrated into SSentiA, it performs best with an accuracy of 77.3%. All the other methods except Vashishtha and Susan (2020) show accuracy between 73%–75%.

Table 7
The comparisons of lexicon-based classifiers with SSentiA in four binary datasets (sorted by F1 score).

Dataset	Model	Method	P	R	F1 Score	MCC	Acc.
TripAdvisor	Lexicon	OpinionFinder	0.627	0.595	0.610	0.220	59.53%
		TextBlob	0.805	0.707	0.753	0.503	70.74%
		SentiStrength	0.793	0.735	0.763	0.521	73.51%
		AFINN	0.814	0.731	0.770	0.539	73.10%
		VADER	0.820	0.752	0.784	0.567	75.17%
		LRSentiA	0.829	0.766	0.796	0.593	76.63%
	Hybrid	SSentiA (LR)	0.912	0.907	0.913	0.828	91.00%
		SSentiA (SVM)	0.926	0.922	0.924	0.834	92.28%
IMDB	Lexicon	OpinionFinder	0.587	0.585	0.586	0.171	58.47
		SentiStrength	0.647	0.644	0.645	0.301	64.45%
		VADER	0.704	0.676	0.69	0.378	67.6%
		AFINN	0.723	0.696	0.709	0.417	69.55%
		TextBlob	0.753	0.695	0.723	0.445	69.56%
		LRSentiA	0.736	0.736	0.736	0.4551	73.62%
	Hybrid	SSentiA (LR)	0.803	0.802	0.802	0.605	80.2%
		SSentiA (SVM)	0.807	0.806	0.807	0.600	80.62%
Amazon	Lexicon	OpinionFinder	0.566	0.557	0.561	0.123	71.07%
		SentiStrength	0.691	0.653	0.672	0.327	78.79%
		TextBlob	0.771	0.622	0.688	0.363	81.09%
		AFINN	0.762	0.668	0.712	0.420	81.99%
		LRSentiA	0.740	0.699	0.719	0.435	81.59%
		VADER	0.764	0.689	0.724	0.445	82.45%
	Hybrid	SSentiA (SVM)	0.791	0.809	0.800	0.606	85.45%
		SSentiA (LR)	0.816	0.800	0.808	0.616	86.82%
Clothing	Lexicon	OpinionFinder	0.555	0.562	0.559	0.118	72.07%
		AFINN	0.735	0.545	0.614	0.189	82.38%
		SentiStrength	0.680	0.574	0.622	0.221	82.12%
		TextBlob	0.714	0.579	0.639	0.260	82.87%
		VADER	0.749	0.600	0.666	0.314	83.75%
		LRSentiA	0.715	0.621	0.665	0.318	83.22%
	Hybrid	SSentiA (SVM)	0.770	0.684	0.725	0.432	85.55%
		SSentiA (LR)	0.759	0.728	0.744	0.463	85.60%

In the 3-class Amazon dataset, as shown in Table 9, among the lexicon-based methods, LRSentiA obtains the highest F1 score of 0.495; SentiStrength achieves the highest F1 score in the 3-class TripAdvisor dataset. LRSentiA shows the highest accuracy in the TripAdvisor dataset, while VADER provides the highest accuracy for the Amazon dataset. When SSentiA is employed, both the accuracy and F1 score are improved significantly over the lexicon-based methods. In the TripAdvisor dataset, SSentiA utilizing SVM achieves an F1-score of 0.740, which is approximately 32% enhancement over the best-performing lexicon-based classifier. In terms of accuracy, an improvement of 30% is observed, from 61.27% to 79.98%. In the Amazon dataset, SSentiA escalates the F1 score to 0.587 from 0.495 (LRSentiA), an improvement of over 20%.

Table 8
The comparison with the existing unsupervised/self-supervised methods in Cornel movie review dataset.

Method	Description	P[a]	R[b]	F1 Score	Acc[c]
Fernández-Gavilanes et al. (2016)	Dependency Parsing-based	0.749	0.748	0.748	74.80%
He and Zhou (2011)	Self-supervised	–	–	–	74.70%
Zhou et al. (2014)	Graph Co-Regularization	–	–	–	73.60%
Vilares et al. (2017)	Rule-based	–	–	–	74.10%
Vashishtha and Susan (2020)	Fuzzy Logic-based	–	–	–	65.45%
SSentiA (LR)	Self-supervised	0.754	0.754	0.754	75.39%
SSentiA (SVM)	Self-supervised	0.777	0.774	0.775	77.39%

[a] Precision
[b] Recall
[c] Accuracy

Table 9
The performance comparison of SSentiA with existing lexicon-based methods in 3-class datasets.

Dataset	Model	Method	Precision	Recall	F1 Score	Accuracy
TripAdvisor	Lexicon	OpinionFinder	0.437	0.414	0.425	41.71%
		TextBlob	0.540	0.468	0.501	53.45%
		AFINN	0.539	0.497	0.518	59.02%
		VADER	0.539	0.502	0.520	60.09%
		LRSentiA	0.563	0.537	0.551	61.76%
		SentiStrength	0.569	0.552	0.561	60.66%
	Hybrid	SSentiA (LR)	0.717	0.705	0.711	76.75%
		SSentiA (SVM)	0.742	0.738	0.740	79.98%
Amazon	Lexicon	OpinionFinder	0.394	0.384	0.389	40.57%
		TextBlob	0.535	0.421	0.471	71.24%
		SentiStrength	0.479	0.471	0.475	61.47%
		AFINN	0.507	0.456	0.480	73.76%
		VADER	0.504	0.461	0.482	75.04%
		LRSentiA	0.512	0.479	0.495	70.07%
	Hybrid	SSentiA (LR)	0.595	0.566	0.580	79.51%
		SSentiA (SVM)	0.588	0.586	0.587	78.28%

Table 10
The performance of SSentiA using partial labeled dataset.

Dataset	Classifier	F1 Score FL/SSentiA-PL	Accuracy(%) FL/SSentiA-PL	(%) of Annotated data FL/SSentiA-PL
Clothing	LR	0.828/0.819	88.4/ 88.6	100%/32.32%
	SVM	0.817/0.797	87.54/87.34	
Amazon	LR	0.885/0.878	91.50/90.90	100%/41.06%
	SVM	0.886/0.876	91.77/91.14	
TripAdvisor	LR	0.946/0.946	94.61/94.7	100%/51.11%
	SVM	0.958/0.956	95.87/95.78	
IMDB	LR	0.873/ 0.867	87.32/ 86.76	100%/59.60%
	SVM	0.891/ 0.879	89.19/87.95	

6.1. SSentiA as a semi-supervised tool (using limited labeled data)

We further show that SSentiA can also be utilized as a semi-supervised tool. We demonstrate that it yields similar accuracy to a fully labeled (FL) training dataset utilizing only a portion of manually labeled data. The comparison is shown with the fully labeled (FL) dataset utilizing two supervised ML classifiers, LR and SVM.

SSentiA leverages predictions from *high and very-high* confidence groups of LRSentiA as pseudo-labeled data (i.e., no manual labels are required for reviews belong to these groups). Only for the reviews that belong to *low, very-low, zero* confidences groups of LRSentiA, manually annotated data are used, therefore, we refer to it as SSentiA-PL (partially labeled). In a fully Labeled (FL) dataset, all labels are manually annotated.

For both the supervised ML classifiers, default parameter settings and class-balanced weights are used. As an input feature, unigram and bigram-based term frequency–inverse document frequency (tf–idf) scores are used. We perform 5-fold cross-validation (80% train data and 20% testing data) and compute averaged F1 score and accuracy. The assessment of the F1 score and accuracy of the 20% test split is performed considering the true label, not the pseudo-label.

As shown in Table 10, SSentiA-PL (partially labeled) reaches similar accuracy to the fully labeled (FL) dataset for binary classification using only 30%–60% of the labeled data. In the Clothing dataset, we find only 32% of labeled data are required for SSentiA-PL to attain a similar accuracy and F1 score of the fully labeled (FL) dataset. For the Amazon, TripAdvisor, and IMDB datasets, SSentiA-PL requires around 41.06%, 51.11%, and 59.60% of labeled data, respectively.

7. Discussion

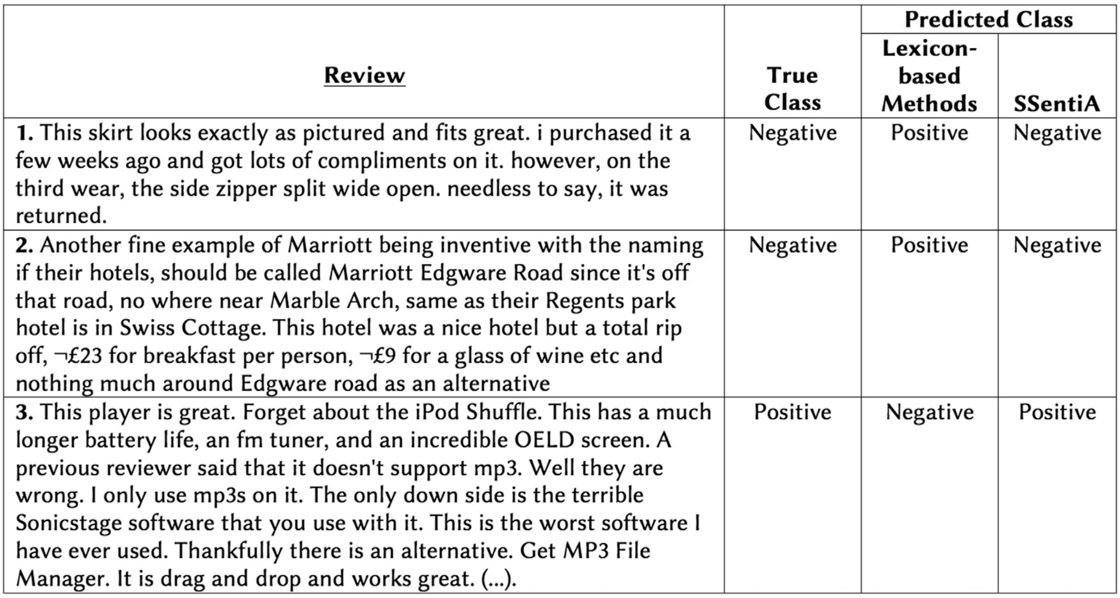
The experimental results reveal that leveraging binary-level sentiment lexicon (i.e., +1, -1), LRSentiA, performs similarly or better than other lexicon-based tools that utilize intensity-based polarity lexicon. In all the four binary datasets, LRSentiA outperforms SentiStrength and TextBlob and yields similar performance to VADER. Among the two 3-class datasets, LRSentiA achieves the highest F1 score in TripAdvisor and highest accuracy in the Amazon dataset.

From Table 5, it is evident that LRSentiA classifies fairly accurately when the review polarity is easily distinguishable (i.e., high confidence score) as implied by the higher accuracy and F1 score in *very-high* and *high* confidence groups. The Chi-squared test indicates that there exists a correlation between the prediction confidence score and accuracy. Therefore, if the user review is comprised of mixed opinions, relying only on the explicit rules or polarity lexicon is often ineffective. Also, assigning weights to individual opinion words does not solve this issue, as seen by the inferior performance of SentiStrength and TextBlob in Table 7.

As Table 5 indicates, LRSentiA can generate highly accurate pseudo-labels (*very-high* and *high* confidence groups) when labeled data are unavailable. The resultant pseudo-labels can be utilized to incorporate supervised ML algorithms to improve the performance of sentiment classification for the hard-to-differentiate reviews. Tables 7 and 9 suggest that integrating supervised ML algorithms can be highly effective as SSentiA outperforms baseline methods consistently and significantly in binary as well as 3-class sentiment analysis.

Fig. 4 shows examples of reviews which lexicon-based methods, such as LRSentiA, VADER, AFINN, and TextBlob misclassify but SSentiA by incorporating ML classifier predicts correctly. For example, the review- *"This skirt looks exactly as pictured and fits great. i purchased it a few weeks ago and got lots of compliments on it. however, on the third wear, the side zipper split wide open, needless to say, it was returned. "* comprised of *positive* words ‘great’ and ‘compliments’. Therefore, all the lexicon-based methods predict it as *positive*.

The following *positive* review, *"This player is great. Forget about the iPod Shuffle. This has a much longer battery life, an fm tuner, and an incredible OELD screen. A previous reviewer said that it does not support mp3. Well they are wrong. I only use mp3s on it. The only down side is the terrible Sonicstage software that you use with it. This is the worst software I have ever used. Thankfully there is an alternative. Get MP3 File Manager. It is drag and drop and works great.”* contains both *positive* and *negative* words, but all the lexicon-based methods predict it as *negative*. Reliance



Review	True Class	Predicted Class	
		Lexicon-based Methods	SSentiA
1. This skirt looks exactly as pictured and fits great. i purchased it a few weeks ago and got lots of compliments on it. however, on the third wear, the side zipper split wide open. needless to say, it was returned.	Negative	Positive	Negative
2. Another fine example of Marriott being inventive with the naming if their hotels, should be called Marriott Edgware Road since it's off that road, no where near Marble Arch, same as their Regents park hotel is in Swiss Cottage. This hotel was a nice hotel but a total rip off, ¬£23 for breakfast per person, ¬£9 for a glass of wine etc and nothing much around Edgware road as an alternative	Negative	Positive	Negative
3. This player is great. Forget about the iPod Shuffle. This has a much longer battery life, an fm tuner, and an incredible OELD screen. A previous reviewer said that it doesn't support mp3. Well they are wrong. I only use mp3s on it. The only down side is the terrible Sonicstage software that you use with it. This is the worst software I have ever used. Thankfully there is an alternative. Get MP3 File Manager. It is drag and drop and works great. (…).	Positive	Negative	Positive

Fig. 4. Examples of reviews which lexicon-based methods (LRSentiA, VADER, AFINN, and TextBlob) misclassify but SSentiA predicts correctly.

on word-level polarity without considering the contextual meaning, as well as dependence on lexicon coverage, makes the lexical rule-based methods ineffective for complex cases.

7.1. Findings and implications

- We find that the lexicon-based systems yield lower accuracy compared to ML methods, particularly for the reviews with mixed sentiments. Hence, it is essential to incorporate the ML method to enhance the performance of sentiment classification.
- We notice a correlation between the presence of sentiment terms/ phrases in a review and the correctness of the prediction using a lexicon-based method. If a review is strongly *positive* or *negative*, lexicon-based methods are quite effective to classify them correctly. Therefore, lexicon-based methods are most suitable for classifying highly polarized reviews.
- We observe that utilizing a binary polarity lexicon, LRSentiA exhibits the better or similar performance of Sentistrength, VADER, TextBlob, and AFINN, which suggests that word-level polarity strength does not influence the performance of a lexicon-based method much. Thus, to improve the accuracy of the lexicon-based method, it is necessary to focus on other aspects.
- We find that it is important to select pseudo-labeled training data carefully to minimize error propagation. Therefore, it is crucial to improve the accuracy of lexicon-based methods so that highly accurate pseudo-labels can be generated.
- Finally, we learn that coupling the ML classifier with the lexicon-based method is essential to building an effective sentiment classifier for the unlabeled dataset.

8. Conclusion

In this work, we study both binary and multi-class sentiment classification problems from the unlabeled data using a hybrid approach. As data labeling often requires domain expertise and manual-effort, our study focuses on identifying sentiments from unlabeled data. We show that the lexicon-based methods, including LRSentiA, are not robust enough to deal with the ambiguity and variations of the natural language as they rely on word-level polarity and simple linguistic rules. Therefore, we integrate the ML algorithm to improve the performance of complicated subsets. Leveraging highly accurate pseudo-labels generated from LRSentiA, our proposed hybrid method SSentiA employs a supervised ML classifier to predict the sentiments of complex reviews. Combining the best of both approaches, interpretability of the rule-based approach, and implicit pattern learning capabilities of ML algorithms, SSentiA significantly enhances the performance of sentiment classification. Besides, we show that SSentiA utilizing around 30%–60% labeled data can achieve similar performance to a fully labeled dataset.

CRediT authorship contribution statement

Salim Sazzed: Designed the methodology, Wrote the paper, Implemented the methodology. **Sampath Jayarathna:** Designed the methodology, Wrote the paper.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data and code availability

The datasets used in this article are available in the following URLs-IMDB: https://ai.stanford.edu/~amaas/data/sentiment/, TripAdvisor: https://figshare.com/articles/TripAdvisor_reviews_of_hotels_and_restaurants_by_gender/6255284, Amazon: http://times.cs.uiuc.edu/~wang296/Data/, and Clothing: https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews. Our source code is available at https://github.com/sazzadcsedu/SSentiA.

References

Agarwal, B., & Mittal, N. (2016). Machine learning approach for sentiment analysis. In *Prominent feature extraction for sentiment analysis* (pp. 21–45). Springer.

Ahmed Abbasi, A. H., & Dhar, M. (2014). Benchmarking twitter sentiment analysis tools. In *Proceedings of the ninth international conference on language resources and evaluation*. LREC'14, Reykjavik, Iceland, May, European Language Resources Association (ELRA).

Andreevskaia, A., & Bergler, S. (2008). When specialists and generalists work together: Overcoming domain dependence in sentiment tagging. In *Proceedings of ACL-08: HLT* (pp. 290–298).

Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. *Knowledge-Based Systems*, *108*, 110–124.

Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., & Iglesias, C. A. (2017). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications*, *77*, 236–246.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Lrec, Vol. 10* (pp. 2200–2204).

Becker, L., Erhart, G., Skiba, D., & Matula, V. (2013). Avaya: Sentiment analysis on twitter with self-training and polarity lexicon expansion. In *Second joint conference on lexical and computational semantics (* SEM), Volume 2: Proceedings of the seventh international workshop on semantic evaluation (SemEval 2013)* (pp. 333–340).

Cai, Y., Yang, K., Huang, D., Zhou, Z., Lei, X., Xie, H., et al. (2019). A hybrid model for opinion mining based on domain sentiment dictionary. *International Journal of Machine Learning and Cybernetics*, 1–12.

Cambria, E., Speer, R., Havasi, C., & Hussain, A. (2010). Senticnet: A publicly available semantic resource for opinion mining. *Artificial Intelligence*, 14–18.

da Silva, N. F. F., Coletta, L. F., Hruschka, E. R., & Hruschka, E. R., Jr. (2016). Using unsupervised information to improve semi-supervised tweet sentiment classification. *Information Sciences*, *355*, 348–365.

Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining* (pp. 231–240).

Fang, X., & Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data*, *2*(1), 5.

Fernández-Gavilanes, M., Álvarez-López, T., Juncal-Martínez, J., Costa-Montenegro, E., & González-Castaño, F. J. (2016). Unsupervised method for sentiment analysis in online texts. *Expert Systems with Applications*, *58*, 57–75.

Gamon, M. (2004). Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the 20th international conference on computational linguistics* (p. 841). Association for Computational Linguistics.

García, A., Gaines, S., Linaza, M. T., et al. (2012). A lexicon based sentiment analysis retrieval system for tourism domain. *Expert Systems with Applications Interantional Journal*, *39*(10), 9166–9180.

Ghiassi, M., Skinner, J., & Zimbra, D. (2013). Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. *Expert Systems with Applications*, *40*(16), 6266–6282.

Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G., & Chatzisavvas, K. C. (2017). Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications*, *69*, 214–224.

Gilbert, C., & Hutto, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international conference on weblogs and social media (ICWSM-14)* (p. 82). Available At (20/04/16) http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf, vol. 81.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, *1*(12), 2009.

He, Y., & Zhou, D. (2011). Self-training from labeled features for sentiment analysis. *Information Processing & Management*, *47*(4), 606–616.

Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. (in press).

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 168–177).

Jiménez-Zafra, S. M., Martín-Valdivia, M. T., Martínez-Cámara, E., & Ureña-López, L. A. (2016). Combining resources to improve unsupervised sentiment analysis at aspect-level. *Journal of Information Science*, *42*(2), 213–229.

Jurek, A., Mulvenna, M. D., & Bi, Y. (2015). Improved lexicon-based sentiment analysis for social media analytics. *Security Informatics*, *4*(1), 1–13.

Kang, H., Yoo, S. J., & Han, D. (2012). Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews. *Expert Systems with Applications*, *39*(5), 6000–6010.

Lee, S., & Kim, W. (2017). Sentiment labeling for extending initial labeled data to improve semi-supervised sentiment classification. *Electronic Commerce Research and Applications*, *26*, 35–49.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167.

Liu, B., et al. (2010). Sentiment analysis and subjectivity.. *Handbook of Natural Language Processing*, *2*(2010), 627–666.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1* (pp. 142–150). Association for Computational Linguistics.

Malandrakis, N., Kazemzadeh, A., Potamianos, A., & Narayanan, S. (2013). SAIL: A hybrid approach to sentiment analysis. In *Second joint conference on lexical and computational semantics (*SEM), Volume 2: Proceedings of the seventh international workshop on semantic evaluation (SemEval 2013)* (pp. 438–442).

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, *12*(2), 153–157.

Melville, P., Gryc, W., & Lawrence, R. D. (2009). Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1275–1284).

Meškelė, D., & Frasincar, F. (2020). ALDONAr: A hybrid solution for sentence-level aspect-based sentiment analysis using a lexicalized domain ontology and a regularized neural attention model. *Information Processing & Management*, *57*(3), Article 102211.

Mittal, A., & Goel, A. (2012). Stock prediction using twitter sentiment analysis. Standford University, CS229 (2011 http://cs229.stanford.edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis.pdf), vol. 15.

Mudinas, A., Zhang, D., & Levene, M. (2012). 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* (pp. 1–8).

Muhammad, A., Wiratunga, N., & Lothian, R. (2016). Contextual sentiment analysis for social media genres. *Knowledge-Based Systems*, *108*, 92–101.

Musto, C., Semeraro, G., & Polignano, M. (2014). A comparison of lexicon-based approaches for sentiment analysis of microblog posts. In *DART@ AI* IA* (pp. 59–68).

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903.

Ortigosa, A., Martín, J. M., & Carro, R. M. (2014). Sentiment analysis in facebook and its application to e-learning. *Computers in Human Behavior*, *31*, 527–541.

Pandey, A. C., Rajpoot, D. S., & Saraswat, M. (2017). Twitter sentiment analysis using hybrid cuckoo search method. *Information Processing & Management*, *53*(4), 764–779.

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. arXiv preprint arXiv:cs/0409058 .

Pang, B., & Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. arXiv preprint.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, *3*(2), 143–157.

Qiu, L., Zhang, W., Hu, C., & Zhao, K. (2009). Selc: a self-supervised model for sentiment classification. In *Proceedings of the 18th ACM conference on information and knowledge management* (pp. 929–936).

Ribeiro, F. N., Araújo, M., Gonçalves, P., Gonçalves, M. A., & Benevenuto, F. (2016). Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, *5*(1), 1–29.

Saif, H., He, Y., Fernandez, M., & Alani, H. (2016). Contextual semantics for sentiment analysis of Twitter. *Information Processing & Management*, *52*(1), 5–19.

Sazzed, S. (2020a). Cross-lingual sentiment classification in low-resource bengali language. In *Proceedings of the sixth workshop on noisy user-generated text (W-NUT 2020)* (pp. 50–60).

Sazzed, S. (2020b). Development of sentiment lexicon in bengali utilizing corpus and cross-lingual resources. In *2020 IEEE 21st International conference on information reuse and integration for data science (IRI)* (pp. 237–244). IEEE.

Sazzed, S., & Jayarathna, S. (2019). A sentiment classification in bengali and machine translated english corpus. In *2019 IEEE 20th international conference on information reuse and integration for data science (IRI)* (pp. 107–114). IEEE.

Schouten, K., & Frasincar, F. (2018). Ontology-driven sentiment analysis of product and service aspects. In *European semantic web conference* (pp. 608–623). Springer.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, *37*(2), 267–307.

Tan, S., Wang, Y., & Cheng, X. (2008). 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* (pp. 743–744).

Thelwall, M. (2018). Gender bias in machine learning for sentiment analysis. *Online Information Review*, *42*, 343–354.

Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, *61*(12), 2544–2558.

Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, *57*, 117–126.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. *ICWSM*, *10*, 178–185.

Turney, P. D. (2002). 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* (pp. 417–424). Association for Computational Linguistics.

Vashishtha, S., & Susan, S. (2019). Fuzzy rule based unsupervised sentiment analysis from social media posts. *Expert Systems with Applications*, *138*, Article 112834.

Vashishtha, S., & Susan, S. (2020). Fuzzy interpretation of word polarity scores for unsupervised sentiment analysis. In *2020 11th international conference on computing, communication and networking technologies (ICCCNT)* (pp. 1–6). IEEE.

Vilares, D., Gómez-Rodríguez, C., & Alonso, M. A. (2017). Universal, unsupervised (rule-based), uncovered sentiment analysis. *Knowledge-Based Systems*, *118*, 45–55.

Wang, H., Lu, Y., & Zhai, C. (2011). Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 618–626).

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing* (pp. 347–354).

Wilson, T., Wiebe, J., & Hoffmann, P. (2009). Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, *35*(3), 399–433.

Xiang, B., & Zhou, L. (2014). Improving twitter sentiment analysis with topic-based mixture modeling and semi-supervised training. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 434–439).

Yu, L.-C., Wang, J., Lai, K. R., & Zhang, X. (2017). Refining word embeddings for sentiment analysis. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 534–539).

Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., & Liu, B. (2011). *89, Combining lexicon-based and learning-based methods for Twitter sentiment analysis: HP Laboratories, Technical Report HPL-2011*.

Zhang, P., & He, Z. (2013). A weakly supervised approach to Chinese sentiment classification using partitioned self-training. *Journal of Information Science*, *39*(6), 815–831.

Zhang, W., Zhao, K., Qiu, L., & Hu, C. (2009). SESS: A self-supervised and syntax-based method for sentiment classification. In *Proceedings of the 23rd Pacific Asia conference on language, information and computation, volume 2* (pp. 596–605).

Zhou, G., Zhao, J., & Zeng, D. (2014). Sentiment classification with graph co-regularization. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: technical papers* (pp. 1331–1340).