Journal of Theoretical and Applied Information Technology
30th November 2019. Vol.97. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CURRENT TRENDS AND RESEARCH DIRECTIONS IN THE DICTIONARY-BASED APPROACH FOR SENTIMENT LEXICON GENERATION: A SURVEY

[1]MOHAMMAD DARWICH, [2]SHAHRUL AZMAN MOHD NOAH, [3]NAZLIA OMAR, [4]NURUL AIDA OSMAN, [5]IBRAHIM SAID AHMAD

[1,2,3,4] Center for Artificial Intelligence Technology, Faculty of Information Science & Technology,

Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia

E-mail: [1]modarwish@hotmail.com, [2]shahrul@ukm.edu.my, [3]nazlia@ukm.edu.my, [4]n.aidaosman@gmail.com, [5]isahmad.it@buk.edu.ng

ABSTRACT

Modern sentiment analysis models rely on a sentiment lexicon, which is the most essential feature that drives their performance. This resource is indispensable for, and greatly contributes to, sentiment analysis tasks. This is evident in the emergence of a large volume of research devoted to the development of automated sentiment lexicon generation models. The task of tagging sentiment-bearing words with a positive or negative connotation, and sometimes with a strength, comprises of two core approaches: the dictionary-based approach and the corpus-based approach. The former involves making use of a digital dictionary to tag words, while the latter relies on co-occurrence statistics or syntactic patterns embedded in text corpora. The end result is a linguistic resource comprising a priori information about words, across the semantic dimension of sentiment. This paper contributes to the existing literature by providing a survey on the most prominent research works that have employed lexical resources, dictionaries and thesauri for sentiment lexicon generation. We also conduct a comparative analysis on the performance of state-of-the-art models proposed for this task, and shed light on the current progress and challenges in this area.

Keywords: *Sentiment Lexicon, Opinion Lexicon, Sentiment Lexicon Generation, Sentiment Analysis, Opinion Mining*

1. INTRODUCTION

Sentiment Analysis (SA), or Opinion Mining (OM), is essentially a Natural Language Processing (NLP) task that involves the detection of user sentiment, attitude, emotion and opinion in natural language text [1], [2], [3]. User reviews generated on the Web and social media have become the de-facto standard for measuring the overall quality of products and services. Nevertheless, it is a costly and time-consuming process for organizations to manually monitor the overwhelmingly massive stream of user-generated reviews on the Web. Consequently, organizations turn to automated SA systems to monitor user sentiment in online reviews, which provides valuable cues for decision making [3]. SA has made possible a rich set of applications in numerous domains, including commercial products and services, movie reviews, and recommender systems [4], among others. Modern SA systems typically use either unsupervised (corpus-based) or supervised machine learning [5] based techniques.

Prior to performing SA on larger pieces of text, however, there is an essential need to classify the smallest sentiment-carrying lexical units of text, *terms*, with their corresponding sentiment properties. The outcome is a sentiment lexicon, which is a linguistic resource that consists of sentiment words tagged with their corresponding polarities. The problem is that, manually tagging words to produce a sentiment lexicon is prohibitively costly in terms of annotator time and effort, and at the same time, is often associated with subjective bias in annotation, since the judgment of (even skilled) annotators varies to a certain degree [6], [7], [8]. Consequently, this area has witnessed the emergence of a large volume of work in the literature concentrated on the SA subtask of marking words with sentiment properties. The explosive interest in the field of SA during the past couple of years has resulted in a high demand for an automated means of generating reliable, high-coverage sentiment lexicons, which is beyond the means of manual hand-labelling. This demand has consequently warranted numerous research efforts in academia on automated sentiment lexicon generation. The dictionary-based approach involves leveraging online dictionaries [9], [10], while the corpus-based approach involves exploiting co-occurrence statistics or syntactic patterns in a text corpus [11], [12].

This survey covers the most prominent research works that utilize the dictionary-based approach for sentiment lexicon generation. This paper is structured as follows: Section 2 presents the works that use the dictionary-based approach to label words with sentiment properties. Section 3 provides a comparative analysis of the accuracy of existing state-of-the-art models. Section 4 discusses the progress made in this area to date, and the challenges that come along with this approach. Section 5 concludes.

2. PRIOR WORKS INSPIRED BY THE DICTIONARY-BASED APPROACH

Numerous works in the related literature have been devoted to the automatic generation of sentiment lexicons using online dictionaries and lexical resources. The underlying intuition is that entries in an online dictionary are not only semantically related by meaning, but for the most part, are also related in terms of their sentiment properties. An online dictionary in this approach plays the role of a semantic knowledge base that typically includes extensive coverage of the entire span of vocabulary entries defined in a natural language. It also possesses a neatly structured layout, in which the entry term on the left-hand and the human-defined gloss on the right-hand hold strong semantic equivalence. Moreover, entries abide formal grammatical conventions and styles, eliminating noise that is typically found in unstructured, free-form text sources. Some lexical resources also exhibit an ontological network that organizes terms based on their lexical/semantic relationships (e.g., synonymy/hyponymy).

The methodology carried out involves an in-depth search of the related literature for works in the scope of sentiment lexicon generation using a dictionary-based approach. For each work, the methodology/technique, evaluation procedure and performance results were the main evaluation criteria considered. The works that have used the dictionary-based approach for sentiment lexicon generation are discussed hereafter.

2.1 Walking WordNet Relations

The technique of 'walking' refers to simply traversing the semantic relations of a seed term and labeling terms encountered, based on their relationship with the original term. A work by [13] develops Q-WordNet, a lexicon generated using a sense-level polarity assignment algorithm. The algorithm is fully unsupervised, and involves traversing all WordNet synsets via all semantic relations of all POS categories, assigning a polarity to each synset along its path. Following the synsets of quality.n.01 and quality.n.02 via the *attribute* relation, positive.a.01, negative.a.01, good.a.01, bad.a.01, superior.a.01 and inferior.a.02 are used as the initial seed terms, from which the walk initiates. All semantic relations are traversed, and not only sentiment-preserving relations, to ensure that terms from all POSs are retrieved, without prioritizing only adjectives (even though all seed terms are adjectives). Any synsets labeled as both positive and negative simultaneously are considered sentiment-ambiguous and filtered out, at every step.

They evaluate and compare the proposed algorithm to that used to generate SentiWordNet 1.0 (Esuli and Sebastiani 2006b) by using Micro-WN(Op), demonstrating that it performs better in terms of accuracy. A limitation inherent in the Q-WordNet algorithm is that it treats word polarity classification as a 'hard' classification problem, and does not consider polarity strength, whereas the SentiWordNet algorithm treats this problem as 'graded' classification by assigning numerical scores to synsets, indicative of both sentiment direction (polarity) and magnitude (polarity strength). Moreover, although it performs well to filter out objective terms, it has a conservative nature in its classification technique, labeling subtly subjective terms as objective (i.e. favoring accuracy at the cost of coverage). For example, Q-WordNet contains only about half the size of subjective synsets in SentiWordNet 1.0 (15,510 vs 35,049). Moreover, it fails to utilize gloss information during the categorization process.

The work by [9] proposes Q-WordNet as a Personalized PageRanking Vector (QWN-PPV). They use a WordNet graph and a personalized pagerank algorithm to assign positive or negative labels to synsets, and evaluate their algorithm with prior algorithms both intrinsically (using the GI) and extrinsically in a polarity labeling task on full-text. They try two different seed set scenarios: the first is similar to their previous work [13], while the second is similar to the 14 paradigm positive and negative terms used by [14]. Four different WordNet graphs are tested, namely, synonymy and antonymy relations (G1), all relations and glosses (G2), all relations except antonymy (G3), and all relations except antonymy and glosses (G4). When walking the graph, a personalized pageranking vector is applied for each polarity on the graph. Several different lexicons are generated depending on the seed set and graph scenario employed.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

They demonstrate the effectiveness of the QWN-PPV algorithm across a series of evaluation experiments with the other similar lexicons. They also mention it is nearly unsupervised, since it only depends on a small seed set and WordNet, and that it can be implemented for other languages, provided an aligned WordNet exists for the target language. They do not consider assigning polarity strength scores to senses, leaving this as a plan for future work.

2.2 Gloss Classification

Gloss classification refers to the classification of a term based on the textual representation of its glosses in a supervised classification task. An early work by [15] determines the polarity of subjective terms by using their glosses as features for a supervised classifier. Although their approach relies on a classifier learning from training data, it is considered semi-supervised, as opposed to (fully) supervised, in the sense that they only use a small set of hand-labeled seed words. The training data is automatically generated by expanding the initial seed set by means of WordNet relations.

They first compile a seed set of predefined positive and negative words (S_p, S_n). They experimented with two different seed sets. The first was adopted from [14], and contains seven positive terms and seven negative terms, denoted as S_p = {good, nice, excellent, positive, fortunate, correct, superior} and S_n = {bad, nasty, poor, negative, unfortunate, wrong, inferior} respectively. The second seed set was adopted from [16], and contains the two singleton sets S_p = {good} and S_n = {bad}.

For expanding the seed set, they considered the following WordNet relations: synonymy (of adjectives), synonymy (regardless of POS), direct antonymy (of adjectives), direct antonymy (regardless of POS), indirect antonymy (of adjectives), indirect antonymy (regardless of POS), hyponymy (regardless of POS) and hypernymy (regardless of POS). They further experiment with combinations of relations, including intersections and unions of relations, e.g., *synonyms (adjectives only)* ∪ *direct antonyms (adjectives only)*. After iteratively extracting these relations between the seed words and WordNet, the seed set is updated (S'_p, S'_n).

The resultant seed set ($S`_p$, $S`_n$) was used for gloss extraction. For every term t_i in the training set, its gloss was extracted from an online dictionary and converted to a vector of words. Each term was then represented by its corresponding gloss in WordNet. They also experimented with adding descriptive terms (if any) and sample phrases (if any) along with the gloss of a word. These vector representations of the terms were then used as a training set to train a binary (positive and negative) classifier. Finally, the supervised classifier was trained and employed to classify the feature vector representations of the words in the test set as positive or negative. Three different classifiers were employed, namely, multinomial naïve Bayes (NB), support vector machines (SVM) and the PrTFIDF Rocchio model.

In a subsequent work, [17] expand their model from binary classification of terms as either positive or negative, to ternary classification of terms as positive, negative or objective. Their approach is similar to that of their previous work [15], except here they consider objective words, and attempt to classify them into a separate class. For the objective seed set, Tr^1_o = {*entity*} was used, which is a root term in the largest generalization graph in WordNet. Synonymy, antonymy and hyponymy relations were used for propagation. For each word extracted from WordNet, it is added to the objective list under the condition that it is not already in the positive or negative lists. The glosses of the generated terms in the resultant seed set are then converted to vector representations, and used as a training set for classification, which is similar to the method of their previous work [15].

Three independent approaches for classification were used. Approach 1 involved two binary classifiers, where the first determines whether a word is subjective or objective, and the second determines whether a subjective word is positive or negative. The first classifier was trained with the expanded seed list, where the subjective class contains the union of the features of terms in both the positive and negative seed lists, and the objective class contains the features of terms in the objective seed list. Approach 2 also involves two binary classifiers. The first one classifies words as positive or *not* positive, and the second one classifies words as negative or *not* negative. Words labeled as positive by the first classifier, and words labeled as *not* negative by the second classifier, are all added to the positive class. The same is performed for the negative class. Finally, words classified into both the positive and negative classes, and also the *not* positive and *not* negative classes, are labeled as objective. Approach 3 involves a ternary classifier. The positive, negative and objective classes are used to categorize input words.

The same three classifiers employed in their previous work [15] were also employed here (NB, SVM and Rocchio classifiers). For features selection, every term that occurs in the glosses of one or more training terms is considered, and its mutual information is computed. The Rocchio classifier was demonstrated to be the most effective classifier compared to the others in this case, since

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

there was no balance between the number of samples in each class, and the other classifiers tend to give a higher priority to the class that contains the largest number of samples. This is due to the manner in which they compute the prior probability of the class itself, which becomes larger as the number of samples in the class becomes larger. Consequently, all of the classifiers employed, with the exception of the Rocchio classifier, classified more terms to the objective class. They conclude by mentioning that their work is a suitable baseline for further investigation on the detection of term orientation and subjectivity.

They [18] expanded their previous work [17] by developing SentiWordNet 1.0, in which every synset in WordNet is assigned three numerical values that range from 0 to 1, denoted as *Pos(s)*, *Neg(s)* and *Obj(s)*. The summation of all three scores is 1.0. These scores represent a synset's likelihood to be positive, negative or objective. They refer to this as *graded classification*, which gives more fine-grained polarity information about a word. This is in contrast to *hard classification*, where a word is assumed to fall into only one of the three categories, rather than to have a numerical score assigned for each category. They consider synsets rather than terms because each term may be associated with multiple synsets, each of which has a different meaning, hence, may express a different sentiment.

For the seed set, they use L_p and L_n, which represent the positive and negative lists respectively. They adopt the seed set used by [14], which contains a total of 14 terms. In contrast to their previous work, they do not use the actual terms, but instead derive a total of 105 synsets from these terms. The seed lists are then expanded through WordNet propagation, applying the similarity, *direct antonymy*, *pertains to*, *derived from*, *also see* and *attribute* relations. For the objective seed set (Lo), they add any synset in WordNet not already available in L_p or L_n, and also not available in the list of positive and negative words in the GI. The glosses of the synsets in the three resultant seed lists are then used as training data for the classification task. Every synset's gloss is represented as a vector, in which stop words are removed and a cosine-normalized tf.idf is computed. These feature vector representations of glosses are used as features for classification.

A total of two ternary classifiers were employed, each with four different training sets, thus, forming the committee of eight classifiers used to vote and assign three numerical scores (*Pos(s)*, *Neg(s)* and *Obj(s)*) to each synset. The various training sets were formulated by using different WordNet propagation iterations (K = 0, 2, 4, 6). The Rocchio and SVMs classifiers were used for classification. If the classifiers all assign the same label to a synset, the maximum score (1.0) is assigned to that synset label. For example, if all eight classifiers classify a synset as positive, it would be assigned scores of *Pos(s)* = 1, *Neg(s)* = 0 and *Obj(s)* = 0. Otherwise, every label is assigned a score that is proportional to the amount of classifiers that have assigned it. Since the eight classifiers work together, each of the three labels may be assigned a score on an eight-point scale ranging from 0 to 1 (normalization is performed by dividing by 8). Finally, this resultant model was used to label all of the synsets found in WordNet to compose SentiWordNet 1.0.

Moreover, the scores tend to be more related to the likeliness of a word being in a particular class, rather than to the actual strength (or the degree of how positive or positive a word is). This is because the assignment of synsets with scores purely relies on statistics, and there is no semantic mechanism that explicitly measures a synset's actual sentiment strength.

They [10] later developed SentiWordNet 3.0. The updated version uses an additional random walk step, following the ternary semi-supervised step previously discussed. Note that [19] also propose a random walk model for word-polarity labeling, but the graph they construct is based on semantic relations among terms, e.g., synonyms and hypernym relations. In contrast, [10] construct a graph between two synsets if the first appears in the gloss of the second.

The random walk is applied on the Pos, Neg and Obj scores from the previous classification step, where they may change based on the results of the random walk. The intuition here is that, the more terms with a certain label (e.g. positive) appear in the gloss of a target term, the more likely it is for the target term to be assigned that label (e.g. positive). The Princeton WordNet Gloss Corpus is used to disambiguate terms appearing within glosses.

2.3 Bootstrapping via Semantic Relations

Bootstrapping methods initiate with set of a labelled terms, and then subsequently and iteratively building a more comprehensive lexicon from this initial set of labelled terms through a semantic network (e.g., synonym and antonyms in WordNet). A work by [20] was the first to propose a bootstrapping algorithm to generate a sentiment lexicon using the dictionary-based approach. They manually compile a seed set of 30 adjectives (each tagged with a positive or negative polarity) that is used to automatically expand into a domain independent sentiment lexicon.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Their technique is based on the intuition that the sentiment properties of a set of carefully hand-labeled seed words are preserved as they strategically propagate through the network of semantic relations in WordNet. Adjectives share an equal polarity as their synonyms, and an opposite polarity as their antonyms. They compile a list of adjectives extracted from a corpus of user-generated product reviews. Figure 1 depicts their bootstrapping algorithm.

```
Procedure OrientationPrediction(adjective_list, seed_list)
begin
  do {
    size1 = # of words in seed_list;
    OrientationSearch(adjective_list, seed_list);
    size2 = # of words in seed_list;
  } while (size1 ≠ size2);
end

Procedure OrientationSearch(adjective_list, seed_list)
begin
  for each adjective wi in adjective_list
  begin
    if (wi has synonym s in seed_list)
      { wi's orientation= s's orientation;
      add wi with orientation to seed_list; }
    else if (wi has antonym a in seed_list)
      { wi's orientation = opposite orientation of a's
                      orientation;
      add wi with orientation to seed_list; }
  endfor;
end
```

Figure 1: Bootstrapping algorithm

For each adjective on the list, they use WordNet to check if it has a synonym that is available in the seed list. If it does, then it is labeled with the same polarity as that of its synonym, and then added to the seed list. The same is done to check if the adjective has an antonym available in the seed list. If it does, then it is labeled with a polarity that is opposite to its antonym, and then added to the seed list. This is repeated in an iterative fashion, and the initial seed list continues to expand until all of the adjectives in the list are labeled.

It uses WordNet to check whether the input adjective has a synonym or an antonym in the seed list, and is labeled accordingly and added to the seed list. If it does not have a synonym nor an antonym in the seed list, then it is skipped, and compared again during subsequent iterations, since the seed word list continuously expands. In other words, the algorithm skips this adjective and moves on to the next, since the skipped adjective's orientation may be determined in a later call of the procedure with a larger seed list. Finally, they incorporate the resultant lexicon into their Feature-Based Summarization system, which performs aspect-based SA on product reviews, and then generates a summary of the results.

Several limitations in this simple approach are pointed out. Any adjectives on the list that are not found in WordNet, or are not labeled, are discarded, since they are assumed to be invalid words, or those that do not express any opinion. Furthermore, their approach allows for an adjective to be labeled as both positive and negative in the case where one of its synonyms and one of its antonyms are both found in the seed list in the same run. In this case they simply rely on the first polarity given to the adjective. Their approach also allows for adjectives to be labeled and added to the seed list multiple times, hence, duplicate words in the resultant seed list must be removed manually. It is also worthy to note that they only consider adjectives for their lexicon, and do not include other word classes (e.g., nouns, verbs and adverbs), which may sometimes also possess sentiment. Moreover, they do not consider providing any information on the sentiment strength of words in their lexicon. Regardless of the limitations in their approach, however, it has set the foundation for further work in exploiting online lexical knowledge bases to label words with their corresponding polarities.

The work by [6] extracts and labels sentiment-bearing adjectives in WordNet with use of a proposed bootstrapping algorithm: Sentiment Tag Extraction Program (STEP). They claim that some words are central, and have a strongly defined polarity (such as good and bad), while others are less central and peripheral, and are therefore more ambiguous. They note that a fall in inter-annotator agreement results is not due to difference in annotation judgment, but rather to the ambiguous nature of peripheral words that lie at the boundary of the neutral category. This makes the task of sentiment assignment to a particular category gradual and fuzzy rather than abrupt.

The STEP algorithm extracted adjectives based on synonymy, antonymy, hypernymy and word glosses. For the first run, a seed set expanded about five times its original size with accuracy comparable to human annotation (78 percent). For the second run, glosses for all words were considered, and if a gloss contains a word found in the seed set, the lexeme of the word associated with that gloss was added to the seed set. For the third run, Brill's POS tagger was used to eliminate POS ambiguity. Contradicting words found in both positive and negative categories were discarded.

Next, they produce a centrality measure by using the 58 seed sets (HM). After applying STEP on each seed set, some words in the seed sets where extracted multiple times to the positive (or negative) category, which shows that these words are more central in their corresponding categories. The Gross Overlap measure is the number of times a word has been extracted during STEP for all 58 sets. The Net Overlap score shows the number of times a word was placed in the positive category subtracted by the number of times a word was placed in the negative category. Hence, the greater the Gross Overlap, and the better the agreement of the assignment of a word to a certain category, the greater the Net Overlap score assigned for that word. Next, the Net Overlap score for every word was used as an indicator to classify it into the positive or negative category, and also as a distance measure from a value of 0 (neutrality). A 0 value was assigned to words that were not extracted during STEP runs, or were identified as positive and negative an equal amount of times. For both positive and negative categories of words based on Net Overlap scores, automatic (using the STEP algorithm) and manual (using the HM test set) tagging were compared to the GI-H4 gold standard.

The highest accuracy achieved for automatic and manual classification was 66.5 percent and 78.7 percent respectively. This proves that automatic and manual classification have a strong correlation, and it is not poor human judgment that results in inter-annotator disagreement, but rather the fuzzy and ambiguous nature of peripheral words that lie on the boundary between their sentiment category and the neutral category. This is in line with the observation by [21], who demonstrated that inter-human agreement was higher when the neutral category was removed from the manual task of labeling words with a polarity.

2.4 WordNet Distance

Distance refers to the path length between a set of seed terms and a set of target terms to be labeled in the semantic network. Terms act as nodes while the semantic relations among them act as edges. The work by [16] also focuses solely on adjectives, and propose a WordNet distance-based similarity measure to determine their semantic orientation. This similarity measure is called the *geodesic distance function*. There are four main subgraphs for each syntactic category (nouns, verbs, adjectives and adverbs) in WordNet. Each has a relatively large component that connects a large amount of synsets.

Synonymous words are connected to form a network in WordNet. The geodesic distance d between a pair of nodes (i.e., words) is the shortest path between them. More specifically: geodesic distance = $d(w_i, w_j)$, where w_i and w_j are two connected nodes in the network. This can be used as a measure as to how similar two words are [16]. Using this, they measure the distance between an adjective and the word *good* to yield a measure of *goodness*, but they claim it is a weak relation. This weakness in relation is due to the wide applicability of bipolar adjectives such as *good* and *bad*, each of which holds 25 and 14 different senses respectively. It takes four steps to get from *good* to *bad*, which can be seen in the following sequence: *good*, *sound*, *heavy*, *big*, *bad*. In other words, $d(good, bad) = 4$. This shows that, although these words are exact opposite in meaning, they are closely related in terms of the synonymy relation.

Using this to their advantage, they measure Osgood's [22] three factors for any affective word. For the evaluative factor, they measure the distance of a word to *good* and *bad*, and use the distance between good and bad as a means of normalization that constrains the resultant value within the range of [-1, 1]. The formula they use to measure the evaluative factor of a word is as follows:

$$\mathrm{EVA}(w) = d(w, bad) - d(w, good) / d(good, bad) \quad (1)$$

where EVA(w) represents the evaluative factor of the word. Words with a negative value are nearer to *bad*, hence, would be considered as having a negative polarity.

Unlike previous works that used WordNet's taxonomic hyponymy structure to determine the semantic orientation of nouns and verbs, they exploited the synonymy network to determine the semantic orientation of adjectives. Their proposed geodesic distance function, however, is constrained to only words that are connected in the network. For example, for the evaluative factor, words that have no connection path with both *good* and *bad* cannot be measured using this function. The evaluative function would assign a value of infinity to a word that is not connected to good as follows: $d(w, good) = \infty$. Consequently, the distance between that word and bad would also receive the same value, which results in a final value of: EVA(w) = 0. Finally, the restriction of the seed words to only one bipolar pair has a negative impact on the overall accuracy of the algorithm.

The work by [23] advances the work of [16] to assign a semantic orientation to adjectives by exploiting the synonymy network in WordNet. The difference is that here they use the resultant value of their function as an indicator of an adjective's sentiment strength. They build an adjective graph that is used to measure the overall distance between a target word and reference words with a known polarity (or seed words). Non-polar (objective) adjectives are assigned dampened sentiment strength. They start with a set of seed words and recursively query WordNet to extract words (a_i) connected to the adjective graph (G_a).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Their work expands on the work of [16] in several ways. First, multiple bipolar adjective pairs are used to compute the semantic orientation of unseen adjectives connected to them in the graph. Second, edge weights are considered to reduce the impact on computation of senses that are at a large distance to a word. Finally, they expand coverage by considering *antonyms*, *similar words* and *related words*, and attempt to assign a lower strength to non-polar objectives as a filtering mechanism.

The work by [24] expands initial positive and negative seed words into sentiment lexicons, using WordNet synonym and antonym paths from the seed words to the target words. They overcome the issue of relying on distance by decreasing the significance of a path as its distance from a seed word increases, and also rank higher words that have less antonyms (or *sentiment flips*) within their path. Taking domain sensitivity into account, they independently construct seven different lexicons based on specific domain or topic (sports, politics, business, etc.). The problem inherent in their algorithm is that using synonym and antonym relations only apply to adjectives, and that using distance as an estimation of polarity is naïve when compared to connectivity.

2.5 WordNet Connectivity

The work by [19] exploit WordNet's semantic structure and propose a Markov random walk algorithm to assign a polarity to a word. They construct a graph G(W, E), where W represents nodes (word-POS tag pairs) and E represents edges that link nodes via synonymy and hypernymy relations. A random walk starts at an arbitrary word with an unknown polarity, and traverses the graph until it stops at a seed word with a known polarity. The time taken to reach the seed word acts as an indication of the polarity of the word. If this is repeated N times, the average time it takes a word to reach a set of seed words with known positive and negative polarities is used to label that word.

This algorithm comes with several benefits. It is easy to implement, fast and is does not require a corpus. The random walk algorithm tends to achieve a high accuracy due to its ability to exploit the multiple relations between words in the WordNet graph. The researchers do not consider strength for labeled words. A possible technique that merits further investigation is to use a word's average hitting time to generate its corresponding sentiment strength value. Figure 2 presents the random walk algorithm.

Algorithm 1 Word Polarity using Random Walks

Require: A word relatedness graph G

Given a word w in V

Define a random walk on the graph. the transition probability between any two nodes i, and j is defined as: $P_{t+1|t}(j|i) = Wij / \sum_k W_{ik}$

Start k independent random walks from w with a maximum number of steps m

Stop when a positive word is reached

Let $h^*(w|S^+)$ be the estimated value for $h(w|S^+)$

Repeat for negative words computing $h^*(w|S^-)$

if $h^*(w|S^+) \leq h^*(w|S^-)$ **then**

  Classify w as positive

else

  Classify w as negative

end if

Figure 2: Random walk algorithm

The work by [23] later applied the above mentioned random walk algorithm on Arabic and Hindi to construct lexicons of words and their semantic orientations in the non-English languages.

2.6 Synset Member Classification

Synset classification is a technique whereby a term is classified with a polarity based on the occurrence of its synset members in the positive and negative classes. The work by [21] proposes a word-sentiment classifier to generate a lexicon of sentiment words and assign a sentiment strength that corresponds to each word. They start with a seed set of sentiment words with manually labeled polarities. They use the intuition that positive words have synonyms that are mostly positive, and antonyms that are mostly negative.

Some extracted words were not listed properly (i.e., mislabeled), or were neutral (i.e., objective). Moreover, common words that have no polarity, such as *get* and *take*, were also extracted. This issue brought up the motivation to assign each word with a corresponding sentiment strength. This way, they kept words with a sentiment strength measured over a predefined threshold on the list. Any words below this threshold were assumed to be neutral, and were thus removed from the list. After the classification process, a lexicon of sentiment words was generated, each with a numerical value representative of its strength. Figure 3 shows some sample words from the lexicon generated by the first model. For example, *abysmal* is weakly negative, while *afraid* is strongly negative.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

abysmal : NEGATIVE
[+ : 0.3811][- : 0.6188]

adequate : POSITIVE
[+ : 0.9999][- : 0.0484e-11]

afraid : NEGATIVE
[+ : 0.0212e-04][- : 0.9999]

ailing : NEGATIVE
[+ : 0.0467e-8][- : 0.9999]

amusing : POSITIVE
[+ : 0.9999][- : 0.0593e-07]

answerable : POSITIVE
[+ : 0.8655][- : 0.1344]

apprehensible: POSITIVE
[+ : 0.9999][- : 0.0227e-07]

averse : NEGATIVE
[+ : 0.0454e-05][- : 0.9999]

blame : NEGATIVE
[+ : 0.2530][- : 0.7469]

Figure 3: Some generated words in final lexicon

This technique was used to classify unseen words from WordNet into the positive list or the negative list based on their sentiment strength. For all new words in WordNet and not in the updated sentiment lexicon, a naïve Bayes classifier was employed to detect their polarity and assign them with a sentiment strength. This model computes the polarity by taking the number of times each of a word's synset members occurs in a class, and dividing this by the total count of words in the class.

They mention that a unigram model is not sufficient, since some words are difficult to annotate into their corresponding class without considering context. As a possible solution, bigrams or trigrams may be added to the seed list to detect neutral words that may carry sentiment when taken together, which may help to increase accuracy.

Their approach relies strictly on an unseen word's synonyms, as well as the occurrence frequency of a word in its synset, to assign it with a polarity and strength. The strength values generated by both models represents the likeliness of a word belonging to some class, rather than the actual strength of a word, and would reliably contribute to polarity assignment, but not to strength assignment.

They [26] later adopt their binary classifier model for three-class classification instead of two-class to filter out sentiment-neutral words from being classified as positive or negative. The same limitations in their previous work also apply here, since they use the exact same model, but apply ternary classification instead of binary to filter out objective terms.

2.7 Label Propagation

Label propagation is a semi-supervised machine learning algorithm that labels terms with a polarity based on the polarity of predefined seed terms. This has been applied to generate reliable sentiment lexicons in the literature. The work by [27] proposes a semi-supervised, graph-based label-propagation algorithm for the development of a sentiment lexicon. They use WordNet to exploit synonymy and hyponymy relations between words in a graph, where nodes represent words and edges represent relations. Several nodes are used to compile a manually labeled seed set of positive and negative words, and this set is then used to classify the remaining nodes as positive or negative.

Only words that intersect in both WordNet and the GI were used, which were split in two equal sets: seed (i.e., train) and test. They test three different semi-supervised learning methods: (1) Mincuts, which attempts to classify any data points by dividing the similarity graph so that it reduces the amount of data points that have different labels; (2) Randomized Mincuts, which is an updated version of Mincuts that makes use of max-flow to generate one out of a set of various possible Mincuts; and (3) Label Propagation, which classifies an input iff it is transitive and possesses some degree of relatedness among the examples.

Another notable point to include about their work is that they experiment with the impact of various numbers of seed words on performance. They show that their label propagation semi-supervised algorithm, with consideration of both synonym and hypernym relations, significantly outperforms baseline work, even with as low as 10 seed words. This demonstrates the effectiveness of this algorithm in cases where labeled data is sparse.

The work by [28] expands a seed set of positive (P), negative (N) and neutral (M) words by propagating through WordNet's synonym and antonym links using a label propagation algorithm. They tag the seed set words with a POS (noun, verb, adjective or adverb) to distinguish among a word's multiple senses and reduce ambiguity. The seed set, along with the synonym set *syn*(*w*) and antonym set *ant*(*w*) for each word *w* within the seed set, are used as input to their algorithm.

The first step in the algorithm is to create a score vector (s^m) and assign a score for each word in WordNet. A score vector is initialized with positive seed words assigned a score of +1, negative seed words assigned a score of -1, and neutral seed words assigned a score of 0. The consideration of the neutral set is to stop the traversal of sentiment

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

tagging to neutral words. Sentiment scores are propagated over the graph by iteratively multiplying the adjacency matrix by the score vector s^m. Some words are maintained in the correct category manually. For example, the word *fast* would be labeled as neutral, but frequently occurs as a positive word in reviews. Consequently, the signs of these words are preserved. For every round, words adjacent to a large amount of words with the same sentiment are assigned a higher score. The decaying value $\lambda = 0.2$ was used to decrease the scores of words that had a greater distance from seed words. This was applied for $M = 5$ iterations. The score vector s was scaled logarithmically. Figure 4 presents some words generated by the label propagation algorithm, along with their corresponding sentiment strength scores.

Positive	Negative
Good_a (7.73)	Ugly_a (-5.88)
Luckily_r (1.68)	Tasteless_a (-4.38)
Intellectual_a (5.07)	Displace_v (-3.65)
Gorgeous_a (3.52)	Beelzebub_n (-2.29)
Angel_n (3.06)	Regrettably_r (-1.63)

Figure 4: Some words generated by label propagation algorithm

They do not compare their approach to previous works, or provide any metrics regarding its accuracy, but mention that most of the sentiment scores generally agree with human judgment. They consider this lexicon domain independent, and employ it in their model to perform aspect-based SA on sentences and text fragments.

2.8 Ising Spin Model

The work by [29] proposes a more sophisticated method to classify words in a dictionary as positive or negative based on their glosses. They employ the *Ising spin model*, which imitates a group of electrons with spins. According to energy reasoning, each electron comprises a spin direction: either up or down. Similarly, every word carries a semantic orientation: either positive or negative. In their method, words act as electrons, and a mean field approximation is used to label them with the average probable semantic orientation. They use magnetization as a criterion for parameter selection, which acts as the semantic orientation for each word. A spin model is a vector of electrons, each with a +1 (an up) or -1 (a down) value. Two neighboring electrons carry the same orientation.

They build a lexical network by linking two words if one of the words is seen in the gloss of the other word. Every link between two words carries either a similar orientation link (SL) or a different orientation link (DL). A weight w_{ij} is set between every. This is referred to as *the gloss network*, *G*. The *gloss thesaurus network*, *GT*, is formulated using synonyms, antonyms and hypernyms, as well as the words in *G*. They apply conjunctive properties on two co-occurring words in a corpus. If two words are separated by *and*, then they have the same orientation (SL), but if they are separated by *but*, they have an opposing orientation (DL). This is called the *gloss thesaurus corpus network*, *GTC*. The polarity of words is measured based on the average values of spins. They built a network of 88,000 words from WordNet by using synonyms, antonyms, hypernyms and glosses. Then they used the Tree-Tagger for POS tagging and stop words removal.

They conclude by claiming that a large corpus such as the Web would improve their model's accuracy, and that the integration of their model with that of [14] would yield promising results. Several limitations exist in their proposed spin model. It does not disambiguate word senses, does not consider structural information within the context of a gloss, and does not have the ability to label idiomatic expressions a polarity. They mention that their proposed model may easily become misled by noisy data in word glosses. They later propose two different models for detecting semantic orientation of two-word phrases [30], [31].

2.9 Morphological Derivation

Morphological derivation has been used to exploit the concept that marked words such as unhappy and impure carry a negative orientation, while their unmarked counterparts (happy and pure respectively) carry a positive one. The work by [32] proposes a high-coverage semantic orientation lexicon that consists of both individual words and phrases (multi-word expressions). They use affix patterns and a Roget-like thesaurus to label words. Their algorithm involves two steps, which are to automatically compile a list of seed words, and then use a thesaurus to label synonyms of the positive seed words as *positive* and synonyms of the negative seed words as *negative*. They make use of the *Macquarie Thesaurus*, which consists of both words and phrases.

To compile the seed set, they use the concept that overtly marked words such as ***un****happy*, ***im****pure* and ***dis****honest* carry a negative semantic orientation, while their unmarked counterparts (*happy*, *pure* and *honest*) carry a positive semantic orientation respectively. Moreover, overtly marked words and their unmarked counterparts tend to carry contrasting orientations. They used 11 types of affix patterns to compile overtly marked words and their unmarked counterparts.

The overtly marked words were labeled as negative, and their unmarked counterparts as positive. There were exceptions such as *bias* and *unbiased*, which, according to the algorithm, would be labeled as positive and negative, respectively, but in reality should be labeled the other way around. Other exceptions included contrasting words that do not have semantic orientations, such as *import* and *export*. They claim that these exceptions were rare compared to the word pairs that abide by the concept of overtly marked words and their antonyms having opposing semantic orientations.

The affix patterns used generated about 2,692 word pairs that were found in the Macquarie Thesaurus, which were automatically marked with semantic orientations. They call this list of orientation-labeled word pairs the *affix seeds lexicon* (ASL). They further expand the coverage of their lexicon by using orientation-labeled words from the GI lexicon.

The Macquarie Thesaurus contains about 1,000 categories, each consisting of sets of similar (roughly synonymous) words and phrases called *paragraphs*. Note that paragraphs are here are similar to the concept of synsets in WordNet. There are about 100,000 paragraphs in the Macquarie Thesaurus. They use the seed words from the ASL and the GI to check for word matches in each paragraph. If a paragraph contains more positive words than negative, then it is labeled as positive, along with all the words it contains. This new list now contains an expanded set of words with labeled semantic orientations. They call this new list the *Macquarie Semantic Orientation Lexicon* (MSOL).

They denote the list generated with use of the words in the ASL as seed words MSOL(ASL), and the list generated with use of the words in the GI as seed words MSOL(GI). The list generated with the use of words in both the ASL and the GI as seed words is called MSOL(ASL and GI), and contains a total of 76,400 entries (51,208 word entries and 25,192 phrase entries). They also consider SentiWordNet (SWN), the Pittsburg subjectivity lexicon (PSL), and the Turney and Littman lexicon (TLL).

Notable limitations in their work include errors in terms of contrasting word pairs having been generated using the affix patterns that carried no sentiment, such as *import* and *export*. Furthermore, there are antonymous word pairs that were labeled incorrectly, since the assumption that overtly marked words are negative (e.g., *impure*), and their unmarked counterparts are positive (e.g., *pure*), does not always hold. This leads to a loss of accuracy in the word labeling task. On another note, the algorithm used in the phrase labeling task for extrinsic evaluation solely relies on word occurrence.

The work by [33] proposes SentiFul by synonymy relations in WordNet and morphological modification of predefined lexical units. They mark words with a polarity based on their underlying emotional vectors from the Affect Database [34]. Then they derive new words through modifying their morphological properties (i.e., affixes). For example, the noun *harmony* is transformed to the adjective *harmonious* using a propagating affix.

The sentiment features of the original lexeme are preserved when a propagating affix is applied, and are transferred to the derived lexical unit. Other types of affixes are reversing, intensifying and weakening; the roles they have on the sentiment features of the original lexeme are inherent in their names. They do not consider other WordNet relations such as antonyms or hypernyms, nor glosses, which may also potentially increase coverage.

3. COMPARATIVE ANALYSIS OF PROMINENT WORKS

In this section, we illustrate comparative analysis of prominent works in this area that have used the General Inquirer lexicon (GI) [35] as a benchmark. These works evaluate their proposed models for accuracy against the intersecting words found in the GI lexicon. The GI was manually compiled, and contains English words that are labeled with semantic categories, among them are lists of positive and negative words (about 3,600 hand-labeled entries altogether). For ease of comparison, Table 1 lists the accuracies achieved by several notable models in this area.

Table 1: Comparative analysis of prominent works.

Research Work	Model	Accuracy
Vicente et al. (2017)	Walking	75.00
Esuli and Sebastiani (2006b)	PNO	66.00
Andreevskaia and Bergler (2006)	STEP	66.50
Kamps et al. (2004)	ShortestPath	68.19
Rao and Ravichandran (2009)	Label Prop	71.63*
Hu and Liu (2004)	Bootstrap	72.80
Mohammed et al. (2009)	MSOL(ASL)	74.30
Takamura et al. (2005)	Spin	83.60
Hassan and Radev (2010)	RandomWalk	**93.10**

** F1-score instead of accuracy*

Note that only the works that use the GI as a benchmark are compared. Also, for the label prop model, the average F1-score for adjectives, verbs and nouns is shown rather than the accuracy. Furthermore, the achieved accuracy for the spin model and the bootstrap model are recorded from [19], who test the accuracy of these models in their work.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The shortest-path model, the STEP model and the PNO model lie within the 60-70 accuracy range. In the shortest-path model, [16] employ a distance-based similarity measure to determine polarity of adjectives. Their model is only applicable on adjectives that are connected to the predefined reference adjective pair. In the STEP model, [6] employ the Sentiment Tag Extraction Program (STEP) to label adjectives. They claim that, it is due to the ambiguous nature of the terms themselves, and not due to the quality of human annotation, that makes the task of sentiment classification at the word level complex. In the PNO model, [18] use a committee of eight ternary classifiers to label WordNet synsets with a *Pos(s)*, *Neg(s)* and *Obj(s)* score, which are trained with gloss information.

The walking model, the bootstrap model, the MSOL(ASL) model and the label prop model lie within the 70-80 accuracy range. In the bootstrap model, [20] set the foundation of exploiting online dictionaries to automatically label words with a polarity, although their model was relatively simple and prone to error. In the MSOL(ASL) model, [32] used affix patterns to automatically generate a seed list, and then used the gloss information in this list to label words found in the Macquarie Thesaurus. The reason the accuracy in the model was off by about a quarter was mainly because word groups in the thesaurus sometimes contain words with different orientations. Moreover, the approach of using word matches between a word group and some training data to label all the terms in that word group a certain polarity purely relies on statistics. In the label prop model, [27] employ a semi-supervised label propagation algorithm, and demonstrate its effectiveness in scenarios where training data is sparse, as well as its ability to be applied on other languages. The work by [9] 'walks', or traverses all synsets in the path, initiating from a few predefined seeds. They employ a fully unsupervised approach, and achieve an average accuracy of 75.00 across all experiments. This result is at the cost of coverage, since the model aggressively filters out even subtly-subjective terms as objective.

The spin model lies within the 80-90 accuracy range. The work by [29] employs a sophisticated Ising spin model from energy reasoning, where terms play the role of electrons. It does not discriminate words in terms of POS, and does not apply to multi-word expressions. The random walk model lies within the 90-100 range. The work by [19] employs a Monte Carlo random walk model to label a word with a polarity based on the average time it takes for that word to iteratively traverse the network at random and hit a predefined reference word. This model's remarkable accuracy is attributable to its ability to exploit the dense network of word relations derived from WordNet, as well as its tendency to effectively use predefined absorbing boundaries.

Based on this comparative analysis, the work done using the dictionary-based approach to compile sentiment lexicons since 2004 to date has continuously improved. Although the dictionary-based approach remains open to further research, there exist inherent limitations, which are discussed in section 4.

4. PROGRESS AND CHALLENGES

Although relatively good progress has been made with the dictionary-based approach, where the automatic labeling of polarity to words matches that of human judgment, there remain challenges for further research in this area. Liu [3] mentions that mislabeled polarity words can be manually cleaned up, and that this task is a one-time effort.

In contrast, Dragut et al. [8] mention that sentiment lexicons that are generated using dictionaries and lexical resources contain complex inaccuracies, beyond the mislabeling of polarity words, which are difficult to manually detect; they also mention that these lexicons exhibit: (a) intra-dictionary inaccuracies, where words are labeled incorrectly; (b) inter-dictionary inconsistencies, where there is contrast between the polarity of words in two different dictionaries; and (c) no consideration of these inconsistencies that occur due to the automatic nature in the approach used to induce them. They attempt to pinpoint inconsistencies found within an individual dictionary, or across multiple dictionaries, with use of a satisfiability problem (SAT). Once these inconsistencies are identified, the lexicon(s) can be improved.

Several prominent issues inherent in sentiment lexicons generated using dictionaries are highlighted hereafter. The first issue is that a dictionary only contains formal words of a natural language, and thus informal words and internet slang commonly used on social media would not be taken into consideration in the final lexicon [36], [37]. For example, in the sentence "*Your new bicycle is so kewl*", the word *kewl* would not be flagged as a positive word. Furthermore, newly coined terms and neologisms such as *Tweet*, *app* and *crowdsourcing* would also not be flagged by using dictionary-generated lexicons.

The second issue is that this approach does not have the capability to generate context-dependent or domain-dependent sentiment words that have a particular polarity in one domain that is different, or even contrasting, in another. For example, consider the sentence '*his laptop is huge'*. The word *huge* in this sentence holds a negative sentiment. In the sentence '*The rooms in this hotel are huge*', *huge* holds a positive sentiment. It can be inferred that the word *huge* has a certain polarity based on the domain it is used in (in this case, laptops vs hotel reviews). Note that this issue is also apparent in non-English languages. Therefore, the dictionary-based approach limits the ability to determine whether some words are positive or negative in a specific context or domain.

The third issue is that, in certain occasions, a sentence that contains sentiment words may not reflect any sentiment. This is especially true in the case of conditional sentences (e.g., "Once I find the most luxurious car, I will certainly buy it!") and interrogative sentences ("Which of these cars is the most luxurious?"). Finally, the fourth issue is that, in contrast to the previous issue, a sentence that does not contain any sentiment words may express a sentiment, based on its surrounding context. For example, "I found hair in the bathroom of my hotel room when I first entered" expresses a negative sentiment, although it does not explicitly contain any sentiment words.

The mentioned issues are also applicable to sentiment lexicon generation algorithms for non-English languages [38], [39], [40], [41], [42].

Therefore, relying solely on the dictionary-based approach is insufficient. Researchers consider exploiting the alternative corpus-based approach to: (a) generate domain and context dependent sentiment words, and deal with subjectivity; (b) increase coverage of new terms not found in an (outdated) dictionary; and (c) include informal social media terms and internet slang. Regardless of the limitations involved, using online dictionaries and lexical resources such as WordNet tends to be an essential initial step in sentiment lexicon generation algorithms, which is attributable to their extensive coverage of words defined in a natural language, rich gloss information, as well as their semantic relations networks.

5. CONCLUSION

In this paper we presented a comprehensive review on the notable research works that focus on the dictionary-based approach for sentiment lexicon generation. In prior work, online dictionaries were applied to label words with their corresponding sentiment properties, in an automated manner. This prior information about words, across the semantic dimension of *sentiment*, is then used for sentiment analysis tasks on larger pieces of text. The underlying intuition in using a dictionary to generate a sentiment lexicon is that words are not only semantically related by meaning, but for the most part, are also related in terms of their semantic properties. Moreover, a dictionary allows for extensive coverage of words defined in a natural language, possesses rich gloss information, and links words together by semantic relations. The end result is a high coverage, domain independent sentiment lexicon that may be used for sentiment analysis tasks on larger pieces of text.

A corpus may be used as a subsequent step for domain adaptation, as well as to include informal words not found in online dictionaries. However, utilizing a dictionary is always an essential first step for the task of sentiment lexicon generation.

ACKNOWLEDGEMENT

This research was partially supported by the Publication Grant (PP-FTSM-2019) and the Fundamental Research Grant Scheme (FRGS/2/2013/ICT02/UKM/02/1).

REFRENCES:

[1] Chaturvedi, I., Cambria, E., Welsch, R. E., & Herrera, F. (2018). Distinguishing between facts and opinions for sentiment analysis: Survey and challenges. *Information Fusion*, *44*, 65-77.

[2] Mäntylä, M. V., Graziotin, D., & Kuutila, M. (2018). The evolution of sentiment analysis—A review of research topics, venues, and top cited papers. *Computer Science Review*, *27*, 16-32.

[3] Liu, B. 2015. Sentiment Analysis: Mining opinions, sentiments, and emotions. Cambridge University Press.

[4] Osman, N. A., Noah, S. A. M., and Darwich, M. 2019. Contextual sentiment based recommender system to provide recommendation in the electronic products domain. *International Journal of Machine Learning and Computing*, *9*(4), 425-431.

[5] Ahmad, I. S., Bakar, A. A., and Yaakub, M. R. 2019. A review of feature selection in sentiment analysis using information gain and domain specific ontology. International Journal of Advanced Computer Research, 9(44), 283-292.

[6] Andreevskaia, A., and Bergler, S. 2006. Mining WordNet for a fuzzy sentiment: sentiment tag extraction from WordNet glosses. *EACL* Vol. 6, pp. 209-216.

[7] Schneider, A., Male, J., Bhogadhi, S. & Dragut, E. (2018). DebugSL: An Interactive Tool for Debugging Sentiment Lexicons. In Proceedings of the 2018 *Conference of the North American Chapter of the Association for*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Computational Linguistics: Demonstrations. pp. 36–40.

[8] Dragut, E., Wang, H., Yu, C., Sistla, P., and Meng, W. 2012. Polarity consistency checking for sentiment dictionaries. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-*Volume 1, pp. 997-1005. Association for Computational Linguistics.

[9] Vicente, I. S., Agerri, R., and Rigau, G. 2017. Q-wordnet ppv: Simple, robust and (almost) unsupervised generation of polarity lexicons for multiple languages. *arXiv preprint arXiv:1702.01711*.

[10] Baccianella, S., Esuli, A., and Sebastiani, F. 2010. SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. *LREC* Vol. 10, pp. 2200-2204.

[11] Alqasemi, F., Abdelwahab, A., and Abdelkader, H. 2019. Constructing automatic domain-specific sentiment lexicon using KNN search via terms discrimination vectors. *International Journal of Computers and Applications*, *41*(2), 129-139.

[12] Deng, S., Kwak, D. H., Wu, J., Sinha, A., and Zhao, H. 2018. Classifying Investor Sentiment in Microblogs: A Transfer Learning Approach.

[13] Agerri, R., and García-Serrano, A. 2010. Q-WordNet: Extracting Polarity from WordNet Senses. In *LREC*.

[14] Turney, P. D., and Littman, M. L. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, *21*(4), 315-346.

[15] Esuli, A., and Sebastiani, F. 2005. Determining the semantic orientation of terms through gloss classification. *Proceedings of the 14th ACM international conference on Information and knowledge management*, pp. 617-624. ACM.

[16] Kamps, J., Marx, M., Mokken, R. J., and De Rijke, M. 2004. Using WordNet to Measure Semantic Orientations of Adjectives. *LREC* Vol. 4, pp. 1115-1118.

[17] Esuli, A., and Sebastiani, F. 2006a. Determining term subjectivity and term orientation for opinion mining. *EACL* Vol. 6, p. 2006.

[18] Esuli, A., and Sebastiani, F. 2006b. Sentiwordnet: A publicly available lexical resource for opinion mining. *Proceedings of LREC* Vol. 6, pp. 417-422.

[19] Hassan, A., and Radev, D. 2010. Identifying text polarity using random walks. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 395-403. Association for Computational Linguistics.

[20] Hu, M., and Liu, B. 2004. Mining and summarizing customer reviews. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 168-177. ACM.

[21] Kim, S. M., and Hovy, E. 2004. Determining the sentiment of opinions. *Proceedings of the 20th international conference on Computational Linguistics*, p. 1367. Association for Computational Linguistics.

[22] Osgood, C. E. 1957. A behavioristic analysis of perception and language as cognitive phenomena. *Contemporary approaches to cognition*, 75-118.

[23] Williams, G. K., and Anand, S. S. 2009. Predicting the polarity strength of adjectives using wordnet.

[24] Godbole, N., Srinivasaiah, M., and Skiena, S. 2007. Large-Scale Sentiment Analysis for News and Blogs. *Icwsm*, *7*(21), 219-222.

[25] Hassan, A., Abu-Jbara, A., Jha, R., and Radev, D. 2011. Identifying the semantic orientation of foreign words. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2 (pp. 592-597). Association for Computational Linguistics.

[26] Kim, S. M., and Hovy, E. 2006. Identifying and analyzing judgment opinions. In Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (pp. 200-207). Association for Computational Linguistics.

[27] Rao, D., and Ravichandran, D. 2009. Semi-supervised polarity lexicon induction. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 675-682. Association for Computational Linguistics.

[28] Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G. A., and Reynar, J. 2008. Building a sentiment summarizer for local service reviews. *WWW Workshop on NLP in the Information Explosion Era* Vol. 14.

[29] Takamura, H., Inui, T., and Okumura, M. 2005. Extracting semantic orientations of

words using spin model. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 133-140. Association for Computational Linguistics.

[30] Takamura, H., Inui, T., and Okumura, M. 2006. Latent variable models for semantic orientations of phrases. *EACL*.

[31] Takamura, H., Inui, T., and Okumura, M. 2007. Extracting semantic orientations of phrases from dictionary. *HLT-NAACL* Vol. 2007, pp. 292-299.

[32] Mohammad, S., Dunne, C., and Dorr, B. 2009. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pp. 599-608. Association for Computational Linguistics.

[33] Neviarouskaya, A., Prendinger, H., and Ishizuka, M. 2009. Sentiful: Generating a reliable lexicon for sentiment analysis. *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*, pp. 1-6. IEEE.

[34] Neviarouskaya, A., Prendinger, H., and Ishizuka, M. 2007. Textual affect sensing for sociable and expressive online communication. *Affective Computing and Intelligent Interaction*, pp. 218-229. Springer Berlin Heidelberg.

[35] Stone, P.J., Dunphy, D.C. and Smith, M.S. 1966. The general inquirer: A computer approach to content analysis.

[36] Peng, W., and Park, D. H. 2011. Generate adjective sentiment dictionary for social media sentiment analysis using constrained nonnegative matrix factorization. *Fifth International AAAI Conference on Weblogs and Social Media*.

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

[38] Asgari, E., Braune, F., Ringlstetter, C., and Mofrad, M. R. 2019. UniSent: Universal Adaptable Sentiment Lexica for 1000+ Languages. arXiv preprint arXiv:1904.09678.

[39] Darwich, M., Noah, S. A. M., and Omar, N. 2017. Minimally-supervised sentiment lexicon induction model: A case study of malay sentiment analysis. In *International Workshop on Multi-disciplinary Trends in Artificial Intelligence* (pp. 225-237). Springer, Cham.

[40] Darwich, M., Noah, S. A. M., and Omar, N. 2016. Automatically generating a sentiment lexicon for the Malay language. Asia-Pacific Journal of Information Technology and Multimedia, 5(1).

[41] Al-Saffar, A., Awang, S., Tao, H., Omar, N., Al-Saiagh, W., and Al-bared, M. 2018. Malay sentiment analysis based on combined classification approaches and Senti-lexicon algorithm. PloS one, 13(4), e0194852.

[42] Alexander, N. S., and Omar, N. 2017. Generating a Malay Sentiment Lexicon Based on WordNet. Asia-Pacific Journal of Information Technology and Multimedia, 6(1).