INTERNATIONAL JOURNAL OF INNOVATIVE COMPUTING
ISSN 2180-4370
Journal Homepage : https://ijic.utm.my/

A Scheme of Pairwise Feature Combinations to Improve Sentiment Classification Using Book Review Dataset

Haisal Dauda Abubakar
Department of Computer Science
Jigawa State College of Education
Gumel, Nigeria
hdabubakar@gmail.com

Sharin Hazlin Huspi
School of Computing
Universiti Teknologi Malaysia
Johor Bahru, Malaysia
sharin@utm.my

Mahmood Umar
Department of Computer Science
Sokoto State University
Sokoto, Nigeria
mahmood.umar@ssu.edu.ng

Submitted: 5/9/2021. Revised edition: 7/10/2021. Accepted: 26/10/2021. Published online: 16/5/2022
DOI: https://doi.org/10.11113/ijic.v12n1.344

***Abstract*—Sentiment Analysis is a Natural Language Processing (NLP) domain related to the identification or extraction of user sentiments or opinions from written language. Although the approaches to achieve the goals may vary, Machine Learning (ML) methods are gradually becoming the preferred method because of their ability to automatically draw useful insight from data regardless of their complexity. However, an important prerequisite for most ML algorithms to learn from text data is to encode them into numerical vectors. Popular approaches to this include word level representation methods TF-IDF, distributed word representations (*word2vec*) and distributed document representations (*doc2vec*). Each of these methods has demonstrated remarkable success in representing the encoded text, however we found that no method has been set to be excellence in all tasks. Motivated by this challenge, an improved scheme of pairwise fusion are proposed for sentiment classification of book reviews. In the experimental findings, Artificial Neural Networks (ANN) and Logistic Regression (LR) classifiers showed that the proposed scheme improved the performance compared to the single method vectorization method. We see that TF-IDF-word2vec performed best among other methods with a mean accuracy of 91.0% (ANN) and 92.5% (LR); showed an improvement of 0.7% and 0.2% respectively over TF-IDF which is the best single vector method. Thus, the proposed method can used as a compact alternative to the popular bag-of-n-gram models as it captures contextual information of encoded document with a less sparse data.**

***Keywords*—Sentiment Analysis, Text Classification, Machine Learning, Book Review**

I. INTRODUCTION

Computational advancement and ubiquity of mobile internet has resulted in unprecedented amount of generated data globally. This has largely been facilitated by the increasing use of social media, micro-blogs, emails and other electronic media platforms which have now become a norm in today's business and commercial space. Consequently, the need for businesses to look beyond the traditional approaches to consumer service satisfaction like surveys is also a fast-increasing trend in today's competitive environment. The need to leverage data from various electronic media platforms to get customer feedbacks on products and services has become crucial to maintaining competitive advantage over competing businesses.

In most cases, humans determine sentiment in text reviews or comments with little or no effort, the same cannot always be said of machine learning based opinion mining systems as several factors including how informative the numerical vector representations of each text input are play an important role. Consequently, pertinent literatures have shown that majority of text/review classification research for sentiment analysis adopt different approaches for the vectorization of the textual features for optimal results. Popular approaches have mainly been either at text, phrase/sentence, or document levels (collection of sentences).

Sentiment analysis deals with the extraction of user opinion or sentiment for a given textual data [1]. User opinion could be categorised as positive or negative in its most coarse-grained categorization. A positive sentiment indicates a user likeness or support for a subject matter while a negative sentiment indicates otherwise. Fine-grained user opinions like "very bad, bad, neutral, good, very good" can also be achieved depending on the task at hand.

The aim of this study is to propose an improved sentiment classification of book reviews, and to analyse the impact of the pairwise combination: TF-IDF-doc2vec, TF-IDF-word2vec, and doc2vec-word2vec feature representations. We will also discuss the the performance of the proposed scheme, relative to individual feature representations methods.

II. Literature Review

Text mining (in relation to sentiment analysis) is used to identify user intentions in written semantics. The three common sentiment analysis methods are supervised, unsupervised and semi-supervised[2], [3].

Sentiment analysis is an automated extraction of attitudes, beliefs, and emotions from text, voice, and database sources via Natural Language Processing (NLP)[4]. The study of emotions includes the grouping of views in texts into categories such as "positive" or "negative" or "neutral." It is often referred to as subjectivity analysis, opinion mining and appraisal extraction. In e-commerce, consumers want to see others ' thoughts on the product before purchasing this [2].

Sentiment classification can be done at Document level, Sentence level and Aspect or Function level. The document standard classifies the entire document as positive or negative. Classification of sentence levels classifies sentence into a positive, negative, or neutral class. Aspect or Function level sentiment classification involves the detection and extraction of the characteristics of the product from the source data. There are two key sentimental analysis approaches: machine learning and lexicon-based methods. Machine learning based approach uses classification techniques to classify text. The Lexicon method uses a dictionary of emotions of views to assess the polarity. We allocate emotion scores to the words of opinion explaining how optimistic, negative and objective the words in the dictionary [5].

One of the important research areas for data extraction is classification; neural network classification is one of the most widely used classification techniques. ANN is an entry network in which the weight of each connection is linked. This consists of an input layer, an intermediate layer or more and an output layer. Neural network learning is carried out by adjusting the connection weight. The network efficiency is enhanced by changing the weight iteratively. ANN can be categorized as feedback network and recurrent network in two groups, based on relation [6].

ANN is proven to have some advantages and disadvantages as [6] indicated that ANN is non-parametric classifier, it efficiently handle noisy inputs, it has a high computation rate, it also provides an association of hierarchical order between variables and classes, on the other hand it becomes complex when values are undecided or not correlated.

The logistic regression model differs from the other algorithms in the sense that they both provide a functional form and parameter vector a to express as the parameters a are determined based on the data set *D*, usually by maximum-likelihood estimation. As the functional form of *f* differs for logistic regression and artificial neural nets, this distinction is important because the contribution of parameters in logistic regression (coefficients and intercept) can be interpreted, whereas this is not always the case with the parameters of *a* neural network (weights).

The most active research and application in the field of classification has been proved by Artificial Networks (ANN). In this work, a back-propagation algorithm was used to train the neural network. The classification of datasets uses the most effective tool known as a neural propagation network. There is further study of the use of the Back Propagation Neural Network (BPNN) to identify objects as remote sensing systems. But BPNN is more efficient than other classification algorithms [7].

Text vectorization or word embedding involves the representation or mapping of words or documents of a corpus to numerical vectors of numbers or real numbers. It is an essential step in machine learning based natural language processing tasks and sentiment analysis since most machine learning algorithms work with numerical input. Several approaches abound in pertinent literatures for representing document/text, however, Bag of words, TF-IDF, word2vec and doc2vec embedding approaches are discussed here due to their relevance to the research at hand [3].

Bag of words model of text vectorization is arguably the earliest and one of the most used approaches that has found extensive application in NLP and information retrieval [8]. It views documents as a collection of words without any regard for grammar or order of words; hence, the name “bag of words”. It represents each document by a fixed length, usually the number of unique words in the corpus, numerical vectors where each feature represents the frequency of occurrence of each word.

For word and phrase level representations, the most common approach has been bag-of-words and bag-of-n-grams. Although loss of word order and semantics are notable disadvantages of bag-of-words, it has produced impressively high accuracy in many sentiments and text classification tasks over the past decade and thus, often used to benchmark new methods. Researchers has been proposing enhanced bag-of-words representation of text reviews by using weights of words as opposed to their frequency for better sentiment scoring and classification.

In an opinion mining of book reviews task, the work of [9] compares performance of five machine learning classifiers on the sentiment analysis of Amazon book reviews using Term-Frequency-Inverse Document Frequency (TF-IDF), a weighted variant of bag-of-words, vectorization. However, the experimental findings ranked Random Forest as the best of the five classifiers in terms of accuracy and processing time.

The bag-of-n-grams vectorization was proposed to address the shortcomings of bag-of-words relating to word order loss by considering short sequence of words. This approach significantly increases dimensionality and leads to a sparse matrix of feature vectors with no consideration for the

semantics of words. [10] conducted a comparative study of different combination of n-gram models for a sentiment classification problem of IMDB movie review dataset. While their findings show impressive results for unigram and bigram models further increase in the number of *n* in the n-gram model led to deterioration in classification performance.

Another word level vectorization approach, which has also been extended to phrase level, that avoids the weaknesses of the bag-of-words based methods is *word2vec* [11]. *Word2vec* models are shallow neural network-based word vectorization approach that consider the context of words such that similar words are closer to each other in the numerical vector space. Given its ability to identify contextual semantics of words, *word2vec* embeddings have found use in languages like Chinese [12], Bengali [13] and Arabic [14,15] for sentiment classification tasks. We also see the use of *word2Vec* in opinion mining of scientific paper citations [16] and hotel reviews [17] are among others.

Word2vec has also been generalized for document level representations. Rather than consider individual words or phrases, document level vectorizations methods consider sequence of sentences, paragraphs, or a whole document. One popular approach is the paragraph vector or *doc2vec* [18]. *Doc2vec* is an unsupervised learning approach that captures the semantics of variable length of texts through the addition of a paragraph matrix/token to the standard word2vec to capture important contexts and semantics of documents. Application of doc2vec to sentiment classification problem has shown competitive results in the IMDB dataset [18], document vectorization for sentiment analysis of clinical discharge documents [19] and Turkish twitter messages [20].

III. METHODOLOGY

The methodology of this study is presented in form of research framework, as shown in Fig. 1. It is grouped into five major phases: Preliminary Study, Data Collection, Data Transformation and Preparation, Experimentation and Evaluation and Result Analysis [21, 22].

The preliminary study phase (Phase 1) deals with the gathering and reviewing of relevant literature and definition of the problems this research aims to address. Phase 2 is where the data used in this research is described with details of pre-processing. Phases 3 and Phase 4 describes the proposed scheme and the experimental set of this research respectively. The findings of this research are described in Phase 5.

A. Phase 1: Preliminary Study

Phase 1 begins with research idea, materials and sources used in literature review for finding the research gap, forming the definition of the problem. We have seen that in previous studies on sentiment analysis have been widely encoded in written native languages using word/n-gram bags, word2vec and doc2vec models. Each of these approaches focuses on different levels of textual granularity that are encoded by generalizing some Thus, the feature vectors generated from each approach represent the abstract level of detail. For example, bag-of-words and n-grams focus on words and short phrase sequences using local representations; word2vec focuses on the representative representation of words / phrases and their context in text, while doc2vec focus words relate their context to the whole document. The common approach for each of these methods to be considered in isolation or comparative analysis between two methods. Rarely, the three methods presented been featured in one sentiment classification research.

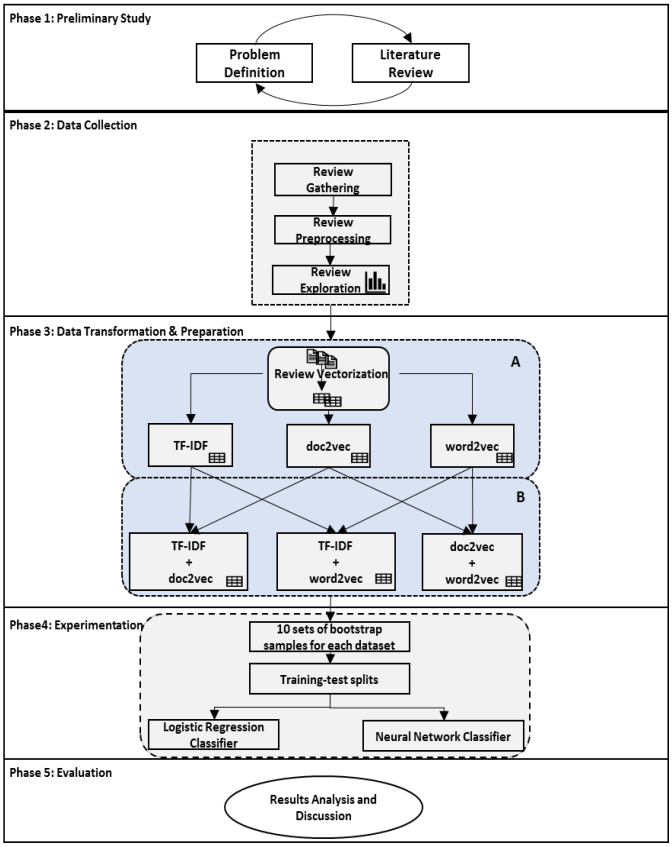

Fig. 1. Research framework

In addition, although the combination of data has been explored to improve some information retrieval tasks, previous research in sentiment analysis has found no proven performance improvement that can lead to the task of sentiments classification. Therefore, this work proposed a combination of word-for-word combinations (TF-IDFs), word2vec feature vectors and doc2vec for better user classification for book reviews.

B. Phase 2: Data Collection

The choice of data used in an experiment is one of the most important elements of a research as the wrong choice of data could result to misleading findings. Thus, the dataset used in this research has been carefully selected from one of

recent publicly available data sources to ensure benchmarking and reproducibility of this research by other researchers.

The dataset used in this work is a collection of detailed book reviews and anonymous user interactions as well as book metadata that were originally collected from *goodread.com* in the 2017 and now publicly available for academic use.

The original and complete dataset consist of 29,154,523 records that were collected from 876,145 users who interacted with a total 2,360,655 books. Given computational limitations encountered in handling the original large-scale data, a representative subset of the entire dataset has been chosen for this research. Specifically, the poetry genre which contains 154,555 detailed reviews of 36,514 books from a total of 2,734,350 user interactions [12] was chosen for this research. Since the text reviews are central to the aims and objectives of this research, the detailed reviews of the three selected genres were merged to form a comprehensive data of 9,176,412 reviews.

For the whole dataset, after pre-processing there were 8,263,614 total number of words, with a vocabulary size of 110,959 words. The length of the longest sentence became 2,116 words while the sentence with minimum length has 3 words. Fig. 2 shows the process of data pre-processing.

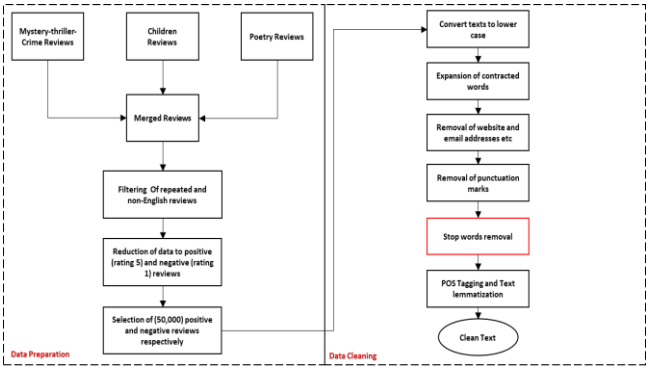

Fig. 2. Data Pre-processing

These transformation to the data happened after extensive stop word removal, text lemmatization, removal all special characters, and numbers. A word cloud showing the most frequent words used in the negative reviews is displayed in Fig. 3. Negative words like disorganize, hard, unnecessary, and pathetic can be seen to common.

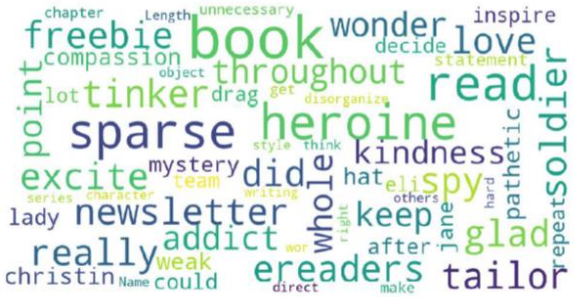

Fig. 3. Wordcloud for Negative Reviews

However, positive words like kindness, excite, love, glad can also be seen to be commonly used in the negative reviews. It should be noted that these positive words were used in the negative sense. For example, a negative review that used the word kindness (highlighted) is as follows:

"*It's probably not fair to write a review of a book I didn't complete, but I just cannot read that book anymore. I got about 3/4 of the way (so close) when I realized I didn't HAVE to finish it. This seriously was the most miserable book I have ever read. My sister put it really nicely when she said there was no human kindness in the book at all-- so true. The funny thing is Emily Bornte's sister Charlotte wrote my number one all time favorite book Jane Erye! Maybe someday, when I have more time to spend on books that are awful I will finish it . . .*"

Fig. 4 showed the word cloud of the most frequently used words in the positive reviews. While positive words like beautiful, favour, smart and glad can be found in the reviews, seemingly negative words are also apparent. For instance, the word 'disappoints' seem to reoccur in quite a number of positive reviews, however its usage came in the positive sense.

Fig. 4. Wordcloud for Positive Reviews

A sample positive review that contains the word disappoint is as follows:

"*This was a wonderfully entertaining book. Part history, part future, part mystry all adds up to one great book. The characters were very well developed. The plot well thought out and executed. It keeps your attention and interest throughout. The editing was well done, also. One of my pet-peeves is a poorly edited book. Highlly recommend. You will not be disappointed. The Christian aspect was not pushy but very well incorporated*."

C. Phase 3: Data Transformation and Preparation

An important step in NLP and specifically sentiment identification in texts is the representation of textual contents in formats understandable by computer programs and machine learning algorithms. This step is often called vectorization; the transformation or encoding of texts into numerical vectors for machine learning. For the sentiment classification task in this research, we adopt three popular vectorization methods: TF-IDF, word2vec and doc2vec. Given that this research seeks to investigate the impact of feature fusion/combination on sentiment classification, different pairwise combinations of the three methods were also created.

Term frequency–inverse document frequency (TFIDF): TFIDF [24] belong to the classical document approach is a weighted variant of the popular bag-of-words. It represents each word, *i*, in a document with the product of the frequency of the word in the document (term frequency) and logarithm of the division of the total number of documents in the corpus by total number of documents in which the word, *i*, occurs in the corpus (inverse document frequency).

D. Phase 4: Experimentation

100,000 Bootstrap samples of the original data was drawn randomly 10 times, split into training and test dataser, and fed as input to each classifier: Logistic Regression and Artificial Neural Networks. Each classifier was trained and tested on the drawn bootstrap samples and the mean of the test performance were reported. Further details on the bootstrapping and Machine learning modelling are given in the following sections.

Logistic Regression

Logistic regression classifier has been chosen as a baseline classifier in this research due to its simplicity and its wide use in related research. 10 bootstrap samples were taken from the original dataset and split into two sets (Training and testing with 7:3 ratio). For each drawn sample a logistic regression model is built and tested while incrementing the bootstrap count, BC, and storing the test result of each model for each iteration. The mean result of the model is reported after the 10th iteration.

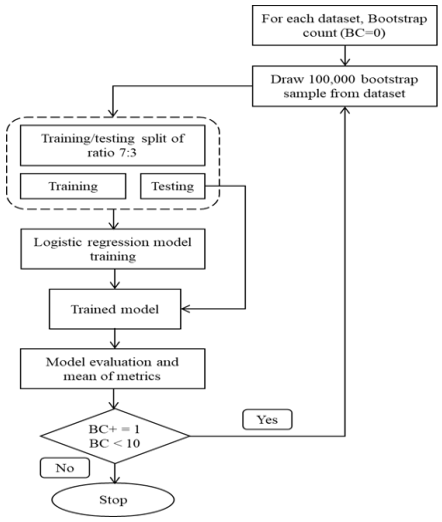

Fig. 5. Logistic regression model

Artificial Neural Network (ANN)

Similar steps were taken for the artificial neural networks as done for the logistic regression model in the preceding section. 10 bootstrap samples were taken from the original dataset and split into two sets (Training and testing with 7:3 ratio). For each drawn sample an ANN model is built and tested while incrementing the bootstrap count, BC, and storing the test result of each model for each iteration. The mean result of the model is reported after the 10th iteration.

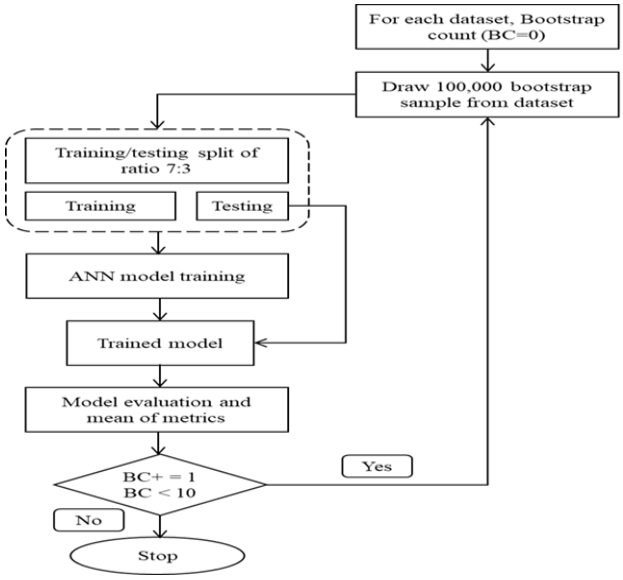

Fig. 6. ANN model

The ANN model used has the following parameters: 2 hidden layers (with 50 and 20 neurons respectively), each hidden layer has a batch normalization, dropout rate of 40%

with a Leaky Rectified Linear Unit activation function. The output layer has a sigmoid activation function. The Stochastic Gradient Descent (SGD) optimizer is used with learning rate=0.01, momentum=0.9 and decay=0.01. The ANN used in this work was implemented using the Keras Deep Learning framework.

E. Phase 5: Evaluation and Result Analysis

In Phase 5 the evaluation of the model was performed. Cross validation approach was used to validate the model while four evaluation metrics were used to establish the performance of each proposed approach in this study. The cross validation divides the dataset into a minimum of 2 training and test partitions. The models were built from the training partition and validated on the testing set. The true performance of the model is its performance on the out of sample test set. Hence the presented performance of the proposed approaches in this study are the test performance. The entire dataset is simply split into two parts, 70% training and 30% testing set.

The performances of the machine leaning techniques are evaluated using parameters like precision, recall, f-measure, and accuracy. The results obtained in this research indicate, the higher values of accuracy for the combined method when compared with other individual method.

IV. Result Analysis

The results of the experiments carried out in this study are discussed in this section. The performance each of the classifier (logistics regression and ANN) on each data vectorization approach is also discussed in this section. The performance of each classifier on each text vectorization method of the book reviews and their combinations are evaluated using accuracy, F1score, precision and recall evaluation metrics. For each classifier, the experiment was repeated 10 times for each data using different bootstrap samples of the data in each round. This is to have an objective view of the performance of each classifier on the datasets as well performance improvement of the fused data over the single vectorization approaches. Thus, for each evaluation metric, the highest, lowest and the mean scores are reported. Likewise, variation of each metric over the 10 rounds of experiments for each data is reported.

A. Logistic Regression Result Analysis

The experimental findings on the performance of logistic regression on each of the 6 vectorizations of the review texts are discussed here as a baseline. The maximum, mean and minimum accuracy, precision, recall and F1 scores for both basic datasets (TFIDF, word2vec and doc2vec) and their combined forms based on logistic regression are discussed below.

Table I shows the performance of logistic regression on TFIDF, word2vec and doc2vec. It is important to examine the performance of the model on these datasets to highlight the performance gains or otherwise that their combined forms offer. Although logistic regression performed well on all the dataset across all evaluation metrics, it achieved the best performance on TFIDF. The performance on doc2vec turns out to be the worst over the 10 rounds of experiments with its best classification accuracy of 0.86: 6% lower than the minimum accuracy of TF-IDF. The performance of doc2vec could be due to the distributed bag of words implementation used for the text vectorization. Here, [18] concatenated two implementations of doc2vec, distributed memory and distributed bag-of-words, to achieve a better result than bag-of-words models in s sentiment classification task of IMDB reviews.

TABLE I. EXPERIMENTAL FINDINGS ON LOGISTIC REGRESSION WITH BASIC DATASET

	LOGISTIC REGRESSION WITH BASIC DATASET					
	ACCURACY			F1 SCORE		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf	**0.924067**	**0.92298**	**0.921567**	**0.924684**	**0.923424**	**0.922087**
word2vec	0.888567	0.886307	0.8836	0.888926	0.886748	0.884417
doc2vec	0.860267	0.857167	0.8542	0.86008	0.856865	0.853585
	PRECISION			RECALL		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf	**0.920838**	**0.91814**	**0.915228**	**0.932267**	**0.928773**	**0.925933**
word2vec	0.887043	0.883335	0.878254	0.892733	0.890193	0.8858
doc2vec	0.86123	0.858679	0.854733	0.858933	0.855067	0.85

We also performed the performance logistic regression on the combined dataset are presented here in Table II. The performance of the classifier on TFIDF+word2vec turned out to be the best over all the evaluation metrics with mean values of 0.925107, 0.92555, 0.920109 and 0.93106 for accuracy, F1 score, precision and recall respectively. Similar but slightly lower performance is reported on the TFIDF+doc2vec dataset. Although the classifier produced as high as 0.905317 in mean accuracy on the word2vec+doc2vec data, its performance range of 0.902434 to 0.908634 across all the evaluation metrics makes it the least performing data of the three combined datasets.

TABLE II. EXPERIMENTAL FINDINGS FOR LOGISTIC REGRESSION WITH COMBINED DATASET

	LOGISTIC REGRESSION WITH COMBINED DATASETS					
	ACCURACY			F1 SCORE		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf+word2vec	**0.926667**	**0.925107**	**0.922733**	**0.927152**	**0.92555**	**0.923306**
tfidf+doc2vec	0.9232	0.921622	0.919333	0.923465	0.92188	0.919639
word2vec+doc2vec	0.907	0.905317	0.903667	0.906832	0.9054	0.903643
	PRECISION			RECALL		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf+word2vec	**0.922522**	**0.920109**	**0.916513**	**0.934**	**0.93106**	**0.929533**
tfidf+doc2vec	0.92216	0.918906	0.914494	0.927	0.92487	0.922333
word2vec+doc2vec	0.908634	0.904642	0.902434	0.910133	0.90616	0.9028

Here we can see that, in terms of classification accuracy, the performance of the classifier is generally higher on the combined data over the 10 bootstrap rounds compared to the basic ones except for TFIDF which showed comparable results with TFIDF+doc2vec and TFIDF+word2vec. While TFIDF+word2vec showed marginally better performance than TFIDF across all the evaluation metrics, TFIDF alone showed better performance than TFIDF+doc2vec in terms of accuracy, recall and F1 score over most of the bootstrap rounds. For the precision score, TFIDF+doc2vec showed better performance than TFIDF alone in most of the bootstrap round. As for word2vec+doc2vec, the performance of the classifier across all evaluation metrics over the 10 rounds was better than both word2vec and doc2vec alone but not as good as TFIDF alone, as shown in Fig. 7.

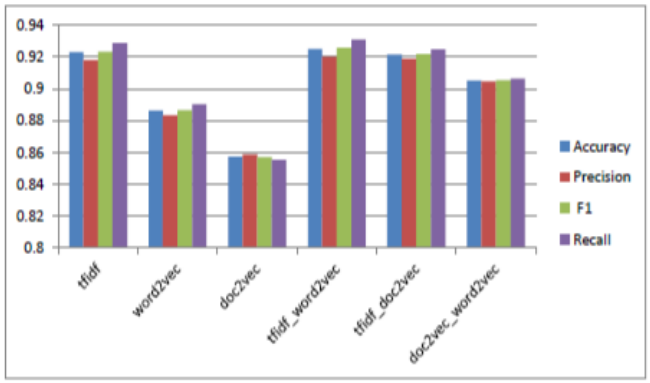

Fig.7. Average Classification Performance of Logistic Regressio*n*

Fig. 7 shows the plot of the mean performance of logistic regression for all six datasets on the accuracy, precision, F1 score and recall. The mean performance serves as a more objective performance measure of the classifier performance on the basic and combined datasets since the experiment were repeated over 10 bootstrap rounds. The plot shows that, in general, combining datasets led to improved performance across all the evaluation metrics over their respective single counterparts. An exception to this is TFIDF compared to TFIDF+doc2vec where TFIDF alone marginally performed better than TFIDF+doc2vec across all evaluation metrics except in terms of precision.

B. ANN Result Analysis

In the discussion of the experimental findings on the performance of Artificial Neural Networks (ANN) on each of the 6 vector representations, please note that extensive search for optimal parameters was not carried out for ANN due to the large number of optimizable parameter space and that the essence of this research is not to investigate the best performance of the classifier, but rather on the investigation of the performance improvements that can be achieved through the combination of feature vectors. Thus, the performance of ANN in terms of the maximum, mean and minimum accuracy, precision, recall and F1 score respectively for both the basic datasets (TFIDF, word2vec and doc2vec) and their combined forms based on ANN are reported.

The classification performance of ANN on TFIDF, word2vec and doc2vec is shown in Table III. Although the performance of ANN on each of the basic datasets can be considered good given that the minimum accuracy reported for the least performing dataset over the 10 bootstrap rounds is 0.857467, ANN performed quite differently on the three datasets in terms of the evaluation metrics.

Similar to what was reported in Logistic Regression; ANN performed best on the TFIDF dataset with average accuracy, precision, F1 score and recall of 0.90345, 0.897632, 0.904156 and 0.910813 respectively. The performance of ANN with doc2vec returned the least scores across all four-evaluation metrics: ranging from as low as 0.851091 precision score to the highest score of 0.873333 in recall over the 10 bootstrap rounds. As discussed previously in Logistic Regression, the variant of doc2vec used in this research, distributed bag of words model, could be the reason why it performed least.

TABLE III. EXPERIMENTAL FINDINGS FOR ANN WITH BASIC DATASET

	ARTIFICIAL NEURAL NETWORK WITH BASIC DATASETS					
	ACCURACY			F1 SCORE		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf	0.9053	0.90345	0.900967	0.905813	0.904156	0.902713
word2vec	0.8916	0.88869	0.886567	0.892211	0.88891	0.886389
doc2vec	0.860267	0.858887	0.857467	0.862069	0.859134	0.856722
	PRECISION			RECALL		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf	0.902323	0.897632	0.886114	0.9202	0.910813	0.905133
word2vec	0.8926	0.887159	0.880586	0.901133	0.890707	0.885
doc2vec	0.862832	0.857628	0.851091	0.873333	0.860707	0.852267

The experimental findings on the performance of ANN with the combined datasets are presented in Table IV. Similarly, to what was reported with Logistic Regression, ANN with TFIDF+word2vec turned out to be the best over all the evaluation metrics with mean values of 0.91044, 0.911317, 0.902477and 0.92036 for accuracy, F1 score, precision and recall respectively. The performance of ANN with TFIDF+doc2vec is similar to its performance with word2vec+doc2vec in terms of classification accuracy and F1 score. However, there is slight variation in their performance in terms of precision and recall.

TABLE IV. EXPERIMENTAL FINDINGS FOR ANN WITH COMBINED DATASET

	ARTIFICIAL NEURAL NETWORK					
	ACCURACY			F1 SCORE		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf+word2vec	0.912233	0.91044	0.908133	0.913545	0.911317	0.908766
tfidf+doc2vec	0.912267	0.899263	0.8974	0.913072	0.899091	0.896829
word2vec+doc2vec	0.899967	0.897953	0.8956	0.899783	0.898355	0.896312
	PRECISION			RECALL		
	MAX	MEAN	MIN	MAX	MEAN	MIN
tfidf+word2vec	0.909326	0.902477	0.89881	0.9274	0.92036	0.9146
tfidf+doc2vec	0.907207	0.900686	0.889506	0.921533	0.897563	0.890333
word2vec+doc2vec	0.901439	0.894874	0.887489	0.909333	0.9019	0.8944

Although the performance of ANN on all the datasets showed fluctuations in terms precision and recall, the performance of the ANN with the combined dataset over the 10 bootstrap rounds compared to the basic ones generally showed improved performance, except for TFIDF which showed comparable or better results over TFIDF+doc2vec and word2vec+doc2vec. ANN with TFIDF+word2vec produced the best results across the evaluation metrics, followed by TFIDF which in most cases marginally performed better than TFIDF+doc2vec and word2vec+doc2vec.

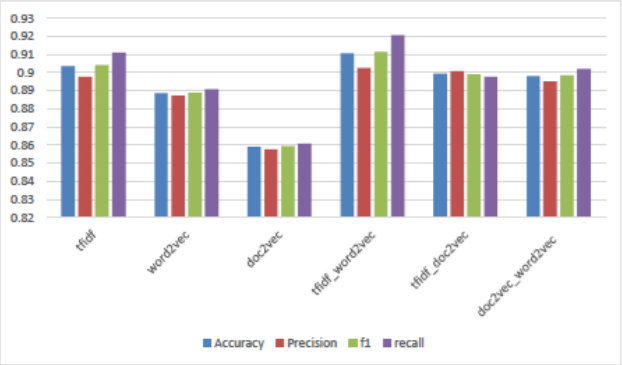

Fig. 8. Average Classification Performances of ANN

Fig. 8 shows the plot of the mean performance of ANN for all six datasets on the accuracy, precision, F1 score and recall evaluation metrics respectively. The mean performance serves as a more objective performance measure of the classifier performance on the basic and combined datasets. The plot shows that, in general, combining datasets led to improved performance across all the evaluation metrics over their respective single counterparts.

The performance of two classifiers, Logistic Regression and ANN, with three basic text vectorization methods their pairwise combination was investigated. Experimental findings show that combining the basic approaches generally brought about improved performance except for few cases where the classifiers produced comparable results.

Thus, from the presented results, it can be deduced that an easy way to improve a model's performance for sentiment classification task is to combine two different text vectorization methods especially data that represent different levels of the text they encode. The reason behind this is not farfetched as the combination of information from different levels of text granularity combined the strength of both methods which in turn produces a more informative data.

V. Conclusion

TF-IDF feature vector representations generally outperform word2vec and doc2vec in book review sentiment classification. From the single vector representation approaches, experimental results show that the performance of TF-IDF is better than both word2vec and doc2vec individually on logistic regression and artificial neural network for the sentiment classification task. Although the word order and semantics is lost in TF-IDF vector representation, it is still capable of identifying keywords that expresses the opinion of users in reviews explicitly. In addition, reviews are often made up of several sentences in which user opinion might only be expressed within a single sentence while the rest of the text might just be what the reviewer read or other uninformative discuss. Such expression or words within large documents will still be represented by TF-IDF unlike word2vec and doc2vec, which seek to vectorise texts relative to their context within a sentence or entire document respectively.

Combined scheme of TF-IDF-word2vec, TF-IDF-doc2vec, and doc2vec-word2vec lead to improved sentiment classification of book reviews relative to single feature vectorization approaches. The proposed schemes generally led to improved performance on the two classifiers (logistic regression and Artificial Neural Network) than single vectorization approaches. Although, the performance of TF-IDF alone is in most cases comparable to or better than TF-IDF-doc2vec and doc2vec-word2vec feature combinations, the performance of doc2vec and word2vec alone improved considerably when combined with TF-IDF. The obvious reason behind this is that the combination of information from different levels of text granularity i.e., word level information of TF-IDF is combined with document level information for doc2vec.

TF-IDF-word2vec performed best compared to all other methods either combined or singly. Word level information from TF-IDF combined with contextual information from word2vec resulted in more informative feature vectors. The performance improvement cuts across the four considered evaluation metrics; classification accuracy, precision, recall and F1-score.

The dimensionality of the proposed feature vector space increased because of the concatenation of different vectorization approaches. While this obviously increases the computational time, this increase in compensated for by the increased in classification accuracy. Besides, the proposed scheme is a more conservative approach which also incorporates contextual information of encoded document compared to bag-of-n-words methods with results in higher dimensionality and sparser data.

The proposed scheme of feature combination can serve as a compact alternative to bag-of-n-gram models. Although bag-of-n-gram models were proposed to capture the order words in short sequences within sentences or documents, they suffer the curse of high dimensionality, and the resulting data is very spares.

VI. Limitation and Future Works

In contrast to the two classifiers used in this study (Logistic Regression and ANN), more classifiers like Naïve Bayes, Support Vector Machine, Random Forest should be

explored to further establish the findings of this research. In the present study, only a unigram model was considered for TF-IDF, a comparative study of the proposed scheme and bag-of-n-gram models should be conducted.

The doc2vec variant that was used in this study is Distributed Bag of Words Model, future works should consider the other the Distributed memory model. Future research should explore the combination of the pairwise combination Distributed memory model with Distributed Bag of Words model and with other feature vectorization approaches like TF-IDF and word2vec for sentiment classification.

A major limitation of this research is that it only considered sentiment classification of book reviews from Goodreads.com only, future research should consider addition of reviews from other sources like Amazon and IMDB reviews.

ACKNOWLEDGMENT

The authors would like to thank Universiti Teknologi Malaysia (UTM) Malaysia for providing the access and place to conduct the research.

REFERENCES

[1] F. F. Balahadia. (2016). Teacher 's Performance Evaluation Tool Using Opinion Mining with Sentiment Analysis. *2016 IEEE Reg. 10 Symp*, 95-98.

[2] M. Ahmad, S. Aftab, S. Muhammad, and S. Ahmad. (2017). Machine Learning Techniques for Sentiment Analysis: A Review. *Int. J. Multidiscip. Sci. Eng*, 8(3), 27-32.

[3] N. Walasek. (2018). Semi-supervised Opinion Mining: Learning Sentiment Towards Vaccination on Dutch Tweets. Master Thesis, Radboud University, The Netherlands.

[4] K. M. A. Hasan, M. S. Sabuj, and Z. Afrin. (2016). Opinion Mining Using Naïve Bayes. *2015 IEEE Int. WIE Conf. Electr. Comput. Eng. WIECON-ECE 2015*, 6(04), 511-514.

[5] V. Pellakuri, N. Revathi, P. Sravya, P. Kavya, and S. Harshitha. (2021). Sentiment Analysis and Predictions of Tweet Emotions Using Different Visualization Approaches. 12(9), 23-30.

[6] R. Bala and D. Kumar. (2017). Classification Using ANN : A Review. 13(7), 1811-1820.

[7] G. Zhang, B. E. Patuwo, and M. Y. Hu. (1998). Forecasting with Artificial Neural Networks : The State of the Art Forecasting with Artificial Neural Networks : The State of the Art. 2070, (March 2018).

[8] S. Stieglitz, M. Mirbabaie, B. Ross, and C. Neuberger. (2017). Social Media Analytics – Challenges in Topic Discovery, Data Collection, and Data Preparation. *International Journal. Information Management*, 39(December), 156-168.

[9] Srujan, K. & Nikhil, s & Rao, Raghav & Kedage, Karthik & Harish, B. S. & Keerthi Kumar, H. M. (2018). Classification of Amazon Book Reviews Based on Sentiment Analysis. 10.1007/978-981-10-7512-4_40.

[10] Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of Sentiment Reviews Using n-gram Machine Learning Approach. Expert Systems with Applications, 117-126. https://doi.org/10.1016/j.eswa.2016.03.028.

[11] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *1st International Conference on Learning.*

[12] Zhang, X., & LeCun, Y. (2015). Text Understanding from Scratch, 1–9. Retrieved from http://arxiv.org/abs/1502.01710.

[13] Al-Amin, Md & Islam, Md Saiful & Uzzal, Shapan. (2017). Sentiment Analysis of Bengali Comments with Word2Vec and Sentiment Information of Words. 10.1109/ECACE.2017.7912903.

[14] Alayba, Abdulaziz & Palade, Vasile & England, Matthew & Iqbal, Rahat. (2018). Improving Sentiment Analysis in Arabic Using Word Representation.

[15] Nasser, Ahmed & Sever, Hayri. (2020). A Concept-based Sentiment Analysis Approach for Arabic. *The International Arab Journal of Information Technology*, 17, 778-788. 10.34028/iajit/17/5/11.

[16] Liu, Haixia. (2017). Sentiment Analysis of Citations Using Word2vec.

[17] Polpinij, J., Srikanjanapert, N., and Sopon P. (2017). Word2Vec Approach for Sentiment Classification Relating to Hotel Reviews. *Proceedings of 13th International Conference on Computing and Information Technology, Bangkok*, 308-316.

[18] Le, Quoc & Mikolov, Tomas. (2014). Distributed Representations of Sentences and Documents. *31st International Conference on Machine Learning, ICML 2014*, 4.

[19] Chen, Q., Sokolova, M. (2021). Specialists, Scientists, and Sentiments: Word2Vec and Doc2Vec in Analysis of Scientific and Medical Texts. *Sn Comput. Sci*, 2, 414. https://doi.org/10.1007/s42979-021-00807-1.

[20] Bilgin, Metin & Köktaş, Haldun. (2019). Sentiment Analysis with Term Weighting and Word Vectors. *International Arab Journal of Information Technology*, 5, 953-959.

[21] M. Abbas, K. Ali Memon, and A. Aleem Jamali. (2019). Multinomial Naive Bayes Classification Model for Sentiment Analysis. *IJCSNS Int. J. Comput. Sci. Netw. Secur*. 19(3), 62.

[22] G. A. A. J. Alkubaisi, S. S. Kamaruddin, and H. Husni. (2018). Conceptual Framework for Stock Market Classification Model using Sentiment Analysis on Twitter based on Hybrid Naïve Bayes Classifiers. *Int. J. Eng. Technol.*, 7(2), 57-61.

[23] Wan, M., & McAuley, J. (2018). Item Recommendation on Monotonic Behavior Chains. *RecSys 2018 - 12th ACM Conference on Recommender Systems*, 86-94.

[24] Trstenjak, B., Mikac, S., & Donko, D. (2014). KNN with TF-IDF based Framework for Text Categorization. *Procedia Engineering*, 69(November 2013), 1356-1364.