A Hybrid Feature Learning Approach for Aspect-based Sentiment Analysis in Drug Reviews

Asmaa Hashem Sweidan a,*, and Nashwa El-Bendary b and Haytham Al-Feel c

a Faculty of Computers and Information, Fayoum University, Fayoum, Egypt
E-mail: aha07@fayoum.edu.eg
b College of Computing and Information Technology, Arab Academy for Science, Technology, and Maritime Transport, Cairo, Egypt
E-mail: nashwa.elbendary@aast.edu, nashwa.elbendary@ieee.org
c Computer Science Department, Imam AbdulRahman Bin Faisal University, KSA, Saudi Arabia
E-mail: Htaafecha@iau.edu.sa

Abstract. This paper aims to develop a novel hybrid feature learning approach for aspect-based sentiment analysis to detect and classify unlabeled data utilizing widely available social data. The proposed approach combines the sentiment lexicon with a pre-trained BERT (Bidirectional Encoder Representations from Transformers) embeddings system based on Ontology and Latent Dirichlet Allocation (LDA) feature extraction for topic modeling. Ontology-based on fuzzy reasoning to describe the semantic knowledge and its relation related to the topics. The BERT with LDA is used to predict the context words to learn the sentence vector and the document vector that is disintegrated into a document weight vector (the weight of each topic) and the topic vector represents one topic that stores related words near each topic. Next using Bi-directional Long Short-Term Memory (Bi-LSTM) to classify extracted sentiment. Various experiments are conducted on social media datasets about Drugs to evaluate the effectiveness of the proposed approach on the aspect as a case-study. Also, several performance evaluation metrics are used to measure their performance. Obtained experimental results showed that the proposed hybrid feature learning approach outperforms other tested feature learning state-of-the-art approaches and improves the feature and topic extraction for unstructured social media text and sentiment classification. Based on the obtained results, it is observed that the performance of the proposed approach increases when using Ontology with BERT embeddings and LDA topic modeling as feature extraction and LSTM as the classifier against using word2vec or BERT individually. The proposed approach achieved an average accuracy of 98.4%, an AUC score of 97.5%, and a F-score of 0.98% for used datasets.

Keywords: sentiment analysis, ontology, topic mining, word embeddings, feature extraction, adverse drug reactions

1. Introduction

Nowadays, Social Networks is considered as platforms where users share their reviews and thoughts. At the same time, it can be utilized in opinion discovery via tracking and analyzing trending topics and feelings, which provides meaningful information for decision making in various domains [1].

Sentiment Analysis (SA) is a Natural Language Processing (NLP) contextual mining method for computational text analysis. It systematically detects, extracts, and investigates subjective information in text data to categorize opinions expressed in text for determining user reviews for entities to obtain user polarity. Although feature extraction and social data analysis are considered as
two main problems facing unstructured data, SA is counted as one of the techniques used to handle unstructured data. Sentiments about features are expressed in the text that the user created in various ways. Therefore, it is significant to discover sentiments regarding a feature or an aspect before determining the sentence polarity [2].

Sentiment analysis is a challenging task in the medical field since it is hard to capture the contextual terms that decide the polarity of sentiment. Besides, there can be major variations in the language expressed by patients, users, and customers and may have several meanings [3]. This field often lacks context-specific lexicons, which help to determine a sentence’s polarity. Many pharmaceutical companies investigate people’s opinions on their products and research the effect of their products on people, such as different medicines, treatments, ADR, and assess consumer satisfaction.

Aspect-level sentiment analysis is very important to extract the relationship between sentiment, target, degree, and negative words. Bi-directional neural networks solve this issue but Neural network structures are also troublesome. In [4], it is found that 40% of the mistakes in aspect-level research based on the fact that no aspects take into consideration. A sentiment classification error is quickly triggered if the target words are neglected.

The current state-of-the-art researches on drug detection with SA utilize two categories: Machine Learning (ML) and Deep Learning (DL) for sentiment classification. In ML, the system performance is substantial but heavily influenced by hand-designed features derived from the textual data of reviews. It is, however, hard to find effective features for extracting the aspect sentiment due to the context similarity for word sentiment. Unlike ML, DL approaches are proven to be suitable for the automatic learning of effective features from the textual data of reviews due to their strong description capability, including convolutional neural network (CNN) or long short-term memory (LSTM). DL outperforms ML though its processing time is too long.

This paper integrates the power of BERT and LDA into one framework. BERT holds the strong relationships between context around a word and captures its meaning, but the output vectors are uninterpreted and do not represent the documents. On the other hand, LDA is interpreted and represents the document. This paper uses the ontology with LDA topic modeling to extract the most relevant topic and the document feature, and exclude inappropriate words to enhance the document representation. The proposed Ontology represents semantic knowledge that enriches the LDA model by forming relationships between aspects. Then, the extracted sentiment knowledge of the input unstructured text data is used as the input of Bi-directional Long Short-Term Memory (Bi-LSTM) to classify sentiment. Finally, the proposed system is able to learn the context around the word and captures its meaning, both syntactically and semantically. The objective of this paper is to investigate the impact of using a hybrid feature learning approach on multi-aspect sentiment evaluation.

The main contributions of this paper are summarized in the following points:

- Integrating lexicon approach based on ontology and sentence embedding with LDA to generate accurate topics and the best document representation for social media data.
- Linking LDA and BERT vectors with hyper-parameter to estimate the corresponding amount of information from each source.
- Utilizing Bi-directional Long Short Term Memory (Bi-LSTM) for classifying aspects in user reviews.
- Developing a novel hybrid feature learning approach for aspect-based sentiment analysis using BERT sentence embeddings with Ontology and LDA models.
- Testing and investigation performance of the proposed approach through implementing multiple experiments on social media datasets.

The remaining of this paper is organized as follows. Section 2 surveys the state-of-the-art researches regarding topic modeling and sentiment analysis for clinical narrative. Section 3 illustrates the fundamentals of the methods used in this paper. Section 4 depicts the different phases of the proposed approach. Section 5 describes the tested dataset and discusses the obtained experimental results as well as presenting findings and limitations. Section 6 presents conclusions and discusses future work.

2. Related Work

This section reviews state-of-the-art studies tackling the problem of SA based word embeddings.
and topic modeling approach with the application in the medical domain to focus on researches with a similar problem statement and datasets that should support further comparative analysis. In [5], the authors proposed a system based on a discriminant document embedding with an extreme learning machine to classify clinical narratives. The authors used skip-gram and Paragraph Vectors-Distributed Bag-Of-Words (PV-DBOW) with Multiple Differentials Analyses (MDAs) to extract the characteristics of clinical texts. Each clinical narrative data have five possible labels associated with symbols for current procedural words. The proposed system achieved 21% improvement in Discrimination Document Embedding (DEE) performance for dataset class2 compared to Document Embedding (DE) and it achieved a F-score of 0.59 and 0.8 using DE and 504 features DDE, respectively.

Also in [6], the authors proposed a novel approach based on feature selection technology to retrieve important features and eliminate noise. The proposed system used a bag of words and Term Frequency/Inverse Document Frequency (TF-IDF) to extract features then used Fuzzy based rough sets to minimize features then used machine learning as a classifier. Experimental findings on the drug dataset show that the proposed approach reduces the difficulty of the final feature.

After that, in [7], the authors proposed a ML approach based on word weight to classify drug sentiment. The proposed approach used word embedding, TF-IDF weighing, and FastText as features extraction. These feature sets were fed into SVM for training and sentiment classification label prediction. The approach examined the drug review dataset. Based on the obtained experimental results, the proposed approach achieved an accuracy of 94.6% and a F1-score of 90.2%, respectively.

On the other hand, in [8], the authors proposed a DL-based system to classify the drug usage relation. The proposed system used the Word2vec algorithm to extract the embedding feature with context feature from medical ontology and used a convolutional LSTM (CLSTM) to detect serendipitous drug usage. The output of Word2vec is then used as an input to CNN for data-driven tasks. The proposed system was examined on the WebMD dataset. The proposed system achieved an AUC of 0.937, precision of 0.811, and recall of 0.476.

Moreover in [9], the authors proposed a weak learning mechanism (WSM) model based on intensive learning, which combined the power of both CNN and LSTM to classify the spirit of drug review. In the proposed model, WSM used poorly labeled data to pre-train and use the model structured data to set initial network parameters. After that, the GloVe embedding was utilized to represent the text and then fed it into CNN, which had the facility to extract a vector layer from the word vector. Then, the model used LSTM to classify the drug. The proposed model was examined on the AskaPatient Forum data. The proposed model WS-CNN-LSTM achieved accuracy, F-score, precision, and recall of 86.72, 86.81, 87.92, and 85.73, respectively.

Additionally, in [10], the authors proposed a DL-based approach, which used BERT models for adverse drug events detection and extraction to identify drug side effects. The proposed approach used BERT word embedding and sentence embedding to extract features and utilized LSTM as a classifier. The proposed approach was investigated on Medline datasets such as WebMD and Drugbank medical articles. The proposed approach achieved 0.97, 0.974, and 0.966 for F-score, precision, and recall, respectively, using BERT Word with Sentence Embeddings.

Also in [11], the authors proposed an approach to classify the medical relationship in the clinical record based on in-depth studies. The proposed approach used Word2vec word embedding to extract the feature and the word position feature, and then CNN was used to feed the Convolution layer to generate local features. After that, multi-pooling operations were used to get more local features in every sentence. The proposed approach examined the clinical record and achieved accuracy, recall, and F-score of 72.9%, 66.7%, and 69.6%, respectively.

Moreover in [12], the authors proposed an approach to generate sentiment lexicons for medical fields that use the word embedding technique. The proposed approach used the Sentiment Lexicon approach using SentiWordNet, term-weighted feature representation methods, position-encoding, and Word2vec to extract feature and Naive Bayes (NB), Support vector machines (SVMs), Random Forests (RF) for drug reviews classification. The proposed system named medical SWN lexicon achieved a
Previous research conducted that feature extraction and sentiment analysis relieved the problem to some extent. However, the existing sentiment analysis approaches do not extract the exact context semantic mining for word and document topics, according to the advantage of Bi-LSTM, BERT, and ontology. An ontology with BERT and LDA was developed to improve feature extraction and information from this model is being explored by Bi-LSTM to improve text classification.

3. Preliminaries

This section describes the fundamentals of LDA based topic mining, Ontology, and Word2vec algorithms.

3.1. Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) represents a method to automatically detect the topics contained in documents or sentences. It is a general model and generates a probabilistic model of how words are produced/written in each document. LDA will determine which words are likely to be created from a particular topic, and then determine the subject of the document by examining these possibilities. LDA takes the Document Term Matrix as input and we do need to provide the number of topics we need to get as a parameter a priori [18, 19]. The LDA steps are depicted in Fig. 1. The LDA can automatically locate a source of large amounts of data and discard the order of words and groups of related words on the same topic depending on their textual identity. However, the LDA has three major problems that affect its effectiveness. First, articles created using LDA include inappropriate features where there is another related text. Second, it produces the noisiest topics from short text and discards important topics when used in limited data sets. Third, the LDA ignores the relationship between the subject and the document where the document contains potentially weak words [3].

3.2. Ontology

Ontology is an information model to outline a domain, which contains a set of types, properties, and relationship types. There is also a general ex-
pectation that the Ontology model features should be related to the object. An ontology consists of five elements as follows: concepts domain, its relationships, instances, rules and axioms, and constraints values of properties. It uses Ontology to exchange knowledge in the field between different systems. An Ontology is provided to determine the semantic relationship between concepts in a particular field. An Ontology is an effective semantic model of the concept structure of domain knowledge. At this stage, a variety of methods are available for constructing ontology from free textual data using natural language processing (NLP) techniques such as Part-of-Speech (POS) Tagging, syntactic parsing, lemmatization, stemming, and some algorithms for information extraction. To reveal the interconnection and interaction of various ontologies, it is necessary to determine the mapping relationship between two domain ontologies, or ontological alignment. Ontology Alignment, also known as Ontology Mapping is the process of finding semantic relationships between entities or concepts in the ontology. A variety of methods are available for Ontology Mapping such as TransH algorithm, Word2vec algorithm, and Node2vec algorithm [20–22].

3.3. Bidirectional Encoder Representations from Transformers (BERT)

In 2018 Google proposed BERT (Bidirectional Encoder Representation from Transformers) for language representation [23]. A bidirectional language representation model with bidirectional semantic information is finally obtained through the MLM (Masked Language Model) method. BERT is developed for pre-training on bidirectional representations of unsupervised data by common adaptation in both the right and left context. As an outcome, the pre-trained BERT model can be adjusted using just one extra output layer to produce a modern model for a wide area of NLP responsibilities. BERT utilizes what transformers have been determined, which is developed to provide sentence encoding. BERT is a model of a language based on a particular model of deep learning. It is determined to provide a contextual, numerical, description of a sentence, or a series of sentences. The input to a shallow and uncomplicated model is this digital representation. Not just that, but the outcomes are usually superior and need a fraction of the input data to be solved for a problem yet to be. BERT works as Bidirectional where pre-training is done by masking some words in the input sequence and trying to train the model on two tasks:

- Masked Language Modeling: Expecting the masked words (only the masked words) and learn to understand how words are related and generating the word relationship then predict the missing words.
- Next Sentence Prediction: Predict whether two sentences are actually following each other (Is Next Sentence)

[4, 24]

4. The Proposed Aspect-based Sentiment Analysis Approach

The proposed system consists of four phases; namely Pre-processing, Feature Polarity Identification, Feature Extraction, and Sentiment Classification. Fig. 2 describes the general structure of the proposed system. In the first phase, pre-processing for the entire corpus to remove unnecessary words. Next, in the feature extraction phase, the Ontology and word embeddings are associated with LDA to
discover the features and provide the semantic meaning of those words that are not in the field. Subsequently, the embeddings sequence words are passed to LSTM to classify unstructured text (social media data).

4.1. Pre-processing:

In this phase, the proposed approach prepares text for the features extraction phase. The major steps in pre-processing input text are presented as the following steps:

1. **Input social text are tokenizing** to present each word as a token and removes the blank spaces.
2. Deleting stop words that are inappropriate words using nltk stop words such as “is, the, a” etc. from sentences because they do not carry any information. and punctuation, slang words converting into their standard form.
3. **Stemming** means that a set of various words with the same meaning we use the Rule-based stemming that provides general rules to stem the words.
4. **Lemmatization** mean as removes inflectional endings and return to the base or dictionary word format. the proposed system used the Natural Language Tool Kit (NLTK) suffix-dropping algorithm for lemmatization and stemming. After lemmatization easy to obtain context lexical for each word. The proposed system used stemming and lemmatization to improve analysis.
5. **Convert uppercase to lowercase** and a sequence of characters that is repeated more than twice to show words as normal words.
6. **Change negative reference** Tweets have different views of negativity. Typically, a negative role plays an important role in determining the sentimentality of a tweet. Here, the negation process can convert “won’t”, “can’t”, and “n’t” into “will not”, “cannot”, and “not”, separately.
7. **Spelling correction** in this step generates all keyword for lexical variants using Levenshtein distance then filter utilized for misspelling keyword.
8. **POS tagging** identifies each word as Noun, Adjective, . . . etc.
4.2. Feature polarity identification

In this phase, analysis of sentiment “POS tagging”, which means assigning the labels to words according to their function in the sentence. We used POS for reducing the dimensionality of the matrix to improve the topic mining model. Stanford POS is used to assign a label (tag) to each word. As the output of the pre-processing phase is word frequencies in each document, the POS recognizes the word property and using it to obtain the score for word polarity. Next POS is searched in SentiWordNet (SWN) and Calculate features polarity [25]. The Proposed system assigns zero scores to the opinion word if it doesn’t contain in SWN else assign one-score for the sentiment word. The major steps in Feature polarity are presented as the following steps:

1. After data pre-processing take POS tag (noun, verb, adjective, and adverb) are searched in SentiWordNet.
2. assign zero scores to the opinion word if it don’t contain in SWN.
3. else assign one score for the sentiment word.
   this calculation using equation (1):

   \[ \text{Polarity}_{\text{positive}} = \frac{\sum_{j=1}^{N} \text{Polarity}_{\text{positive},j}}{n_{\text{syn}}} \]  

   - where \( \text{Polarity}_{\text{positive}} \) is positive polarity score and \( n_{\text{syn}} \) is number of syn-sets word.
4. after compute positive polarity ,negative polarity score and objective polarity score , compute final word score word using equation (2):

   \[ \text{Polarity}_{\text{syn}} = \text{Polarity}_{\text{positive}} - \text{Polarity}_{\text{negative}} \]  

5. compute final sentiment word score according to equation (3):

   \[ \sum_{j=1}^{N} \text{Polarity}_{\text{syn}} \]  

6. Output: is positive if result score is greater than zero and negative if less than zero and neutral if equal zero.

4.3. Feature extraction

Current feature extraction-based DL approaches to sentiment analysis aim to improve their efficacy by using pre-trained embedding words and display the relationship between text classification and feature. Several methods have been developed to take advantage of unstructured data, such as training domain-specific embedding of the word, transfer learning, and fine-tuning of pre-trained models. A pre-training and the fine-tuning process take a process that has already been prepared for a particular task and applies it to a similar second task. This leverages the feature extraction from scratch on the first task without training on the second task. Therefore, it is a form of learning through inductive transfer.

Word and sentence vectors were utilized as feature input for NLP models such as LSTMs entries within the type of numerical vectors, which generally indicates features like vocabulary and elements of speech into digital representations [26]. In the past, words were identified as certain displayed values (one-hot encoding) or better used as neural input words where vocabulary words matched the fixed-length feature from models like Word2Vec. While BERT represents a more meaningful context and Word2vec gets a more representative representation. BERT generates powerful word presentations within the context, compared to, word2vec in which each word receives a specific presentation regardless of the context in which it comes from [4, 24].

In this phase, the proposed approach utilizes BERT, LDA, and Ontology-based vector representation models to extract features. The proposed approach identifies aspects that are multi-aspect, implicit, and comparative. After labeling the data, the proposed approach uses the BERT model to predict the words of the surrounding context that occur in a given current word and the whole sentence context that involves other features estimated by BERT. The proposed approach introduces the word and sentence embedding with topic mining to improve the sentiment analysis and uses BERT to
overcome the context problem limitation.

This phase includes two parts: 1) the generation of the word vector by ontology and BERT and 2) the generation of the document vector by LDA to predict a word so that all vectors are trained concurrently. In short, the purpose word is used to predict the context words to learn the vector. Then, the sequence of the BERT word embedding is used to compute the sentence embedding. Finally, this vector is passed to LDA as an encoder. The document vector is disintegrated into a document weight vector (the weight of each topic) and the topic vector represents one topic that stores related words near each topic. The document vector formula is depicted in equation (4):

\[
d_j = Wd_j \cdot \vec{r}_0 + Wd_{j1} \cdot \vec{r}_1 + \cdots + Wd_{jn} \cdot \vec{r}_n
\]  

where \( \vec{d}_j \) is document vector and \( Wd_{jn} \) is document weight in \( n \) topic, \( \vec{r}_n \) \( n \) topic vector.

In this phase, after preprocessing, the data is used to compute its feature polarity. Then, the proposed approach calculates BERT embedding by using a masked language model, which means that certain terms are randomly masked and expected during the pre-training phase so that the representation of two different text directions can be merged. After that, each piece of data is assigned a sentiment label. Finally, attention weights are utilized to extract important features. The greater the weight of the attention is, the more important the word is. Then, the weight of attention is calculated using equation (5). Finally, the BERT word embedding is used to calculate sentence words that are linked with each word embedding to get meaningful sentence embedding. As illustrated in Algorithm 1. In this approach, in the original sentence, a mask token [MASK] is replaced with an opportunity of 80%; the word is replaced with a random word with a chance of 10%; the sentence is not modified in any respect with a percentage of 10%. Using the BERT model sometimes loses the basic semantics of the feature, so Ontology is used to extract semantic relationships among various concepts. Protégé-OWL is used to build a classical ontology. A fuzzy OWL plug-in is then applied to extend classical ontology to vague ontology. Valuable knowledge in Drugs are collected from various Drug-related social data about the condition, Side Effects (ADR), Dosage or Effectiveness, and other features. Then, they have delivered to ontology. The proposed approach uses ontology for the following main purposes: The ontology includes fixed entities associated with drug and their relationships, which may be used to extract entities from social information. Fuzzy ontology can also be used to efficiently categorize reviews and tweets and calculate the polarity of drug features. The major steps in this phase are presented in Algorithm 2.

### Algorithm 1 Feature extraction with BERT

**INPUT:** The pre-processing phase output

**OUTPUT:** Context vector

1. Each word form pre-processing phase set a predefined word ID using WordPiece vocabulary.
2. Place special tokens at the start [CLS] and finish [SEP] of the sequences.
3. For Each token is converted into a representation of the vectors \([V \times N \times d]\), where \(V\) is the number of layers, \(N\) is the sentence length, and \(d\) is the embedding vector length.
4. Set segment ID to indicate whether token belongs to each sentence and its position to indicate its position
5. Search and select random token to masked set
6. Calculate the weight of attention using Softmax function which generates the probability distribution of words in the context of the original words. This model is described in equation (5):

\[
\text{softmax} \left( \frac{(X \cdot W^Q) \cdot (X \cdot W^K)^\top}{\sqrt{d_k}} \right) \cdot (X \cdot W^V)
\]

Where \(X\) is an embedding matrix of one sentence with \(n\) words; \(m\) is the embedding dimension
7. Calculate the attention then summarized as one before going through a perception of a single layer.
8. Predict the enclosing context words that occur in a given document that is a greater weight of the attention.
Algorithm 2 Feature extraction with Ontology

**INPUT:** POS and feature polarity score

**OUTPUT:** Features vector of the Ontology (concept and relationship)

1. Build Ontology (concept (class)) and their relationships (property).
2. Search in Ontology and extract feature.
3. Compute weigh using Term Frequency and Inverse Document Frequency (TF-IDF) as shown in equation (6):
   \[ tf - idf_{t,d} = (1 + \log tf_{t,d}) \cdot \log \frac{N}{df_t} \]  
   \[ (6) \]
   Where \( t \) refer to the terms; \( d \) is document; \( D \) is the collection of documents [?].
4. Select the top \( N \) feature based on (TF-IDF).
5. Use similarity metric to compute similarity between concept as shown in equation (7):
   \[ Sim(onto, toic) = \sum_{t=1}^{T} f_{onto, t_i} \cdot tf - idf_{t,d} \]  
   \[ (7) \]
   where \( f_{onto} \) is Ontology feature, \( toic \) is topic keyword and \( i \) importance feature.
6. Extract the opinion words which are used to find features.

After extracting the feature word vectors using BERT and Ontology, the word vectors are passed to the LDA algorithm to merge the document vector that is disintegrated into a document weight vector (the weight of each topic) and the topic vector aiming at balancing the relative importance of information from each source. Since the sequential vector is in a high-dimensional space, where information is sparse and interconnected, the proposed approach uses an autoencoder to learn how to represent a latent space with lower dimensions of the sequential vector. Assuming the sequential vector must be in a variegated form in high-dimensional space. It obtained representations of lower-dimensions with more intensive information. It utilized LSTM methods and received qualitative topics. The procedure of LDA is shown in Algorithm 3.

Algorithm 3 LDA with BERT& Ontology model

**INPUT:** Context Vector: sentence vector from BERT model with Ontology vector

**OUTPUT:** Topic Feature Matrix

1. Search in document and assign every word randomly inside the document to one in all \( N \) topic matter.
2. For every assigned document presents record topic representations and word distributions of all the topics.
3. In each document \( D \) search about the word and calculate:
   i. Calculate the word rate that is assigned to topic \( t \).
   ii. Calculate the assignment’s rate for topic \( t \) for all documents \( d \), which comes from the word \( w \).
4. Reassign word \( w \) a new topic \( t \) where we choose topic \( t \) with probability
5. Repeat five-step for a large number of times.
6. Generate document vector(topic word matrix and a document topic matrix) which helps us in looking at potential topics in the corpus.
7. Generate context vector that is the sum of a document vector and the word vector.
   \[ \vec{c_j} = \vec{w_j} + \vec{d_j} \]
8. Projects latent document weights onto topic vectors
   \[ \vec{d_j} = a_{j0} \cdot \vec{f_0} + a_{j1} \cdot \vec{f_1} + \cdots \]
9. Generate the feature vector.

4.4. Sentiment classification

The main aim of this phase is to detect all topics the user mentioned about the service from their reviews. The purpose of this paper is to distinguish the feature or entity types associated with issues that may be granted feature identification tasks. Also, in social media, words can be used before and after the specified word to determine its semantic meaning. So, the proposed approach processes text forward and backward and deals with the long text in the document. LSTM is used to classify data.
The Bi-LSTM layer uses a 150-dimensional semantic vector. To avoid over-fitting, 150 neurons are used in Bi-LSTM. The major steps in this phase are presented in the following: Textual content vectors and parameters are fed into LSTM, then those parameters are used inside the layers of LSTM to perform feature learning, updating and supplying sufficient consequences of sentiment classification.

LSTM is one of the RNN architectures. The LSTM layer is built as a memory cell to store the previous information. The unit of LSTM involves the following components in each layer:

- Inputs: Previous output (weight of the previous) \( (h_{t-1}) \) and current input \( (x_t) \)
- Forget gate: is compute from \( \sigma \) determines whether to discard the information from the current state of the cell \( C_t \)
  * Output: \( f_t = \sigma(W_f \times [h_{t-1}, x_t] + b_f) \) A number between 0 and 1 for each number in cell state \( C_{t-1} \).
- Input gate: to monitor the current information flow on the value of the memory cell state. \( \sigma \) decides which values will be updated
  * \( i_t = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \)
  * \( \tanh \) creates a vector of new candidate value that calculated at the current state as shown in \( \tilde{C}_t = \tanh(W_C \times [h_{t-1}, x_t] + b_C) \)
  * \( i_t \times \tilde{C}_t \)
- After Forget and Input gates compute: the status value of current memory cell \( C_t \) is calculate \( C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \)
- Final step \( \sigma \) determines what parts of the cell state \( C_t \) will be outputted
  * \( o_t = \sigma(W_o \times [h_{t-1}, x_t] + b_o) \)
  * And \( \tanh \) used to generate a value between -1 and 1, multiply by \( o_t \)
- Last unit in LSTM compute \( h_t = o_t \times \tanh(C_t) \)

LSTM’s final output is \( h_t \), which is determined using the cell state-input matrix and output of the output gate’s feature matrix at this time, applying the LSTM output layer to determine the probability value using the Softmax function. The probability outputs \((0,1)\) indicate that the feature found in the social text may be positive or negative, which predicts the sentiment label inside the entered text.

\[
\text{Softmax}(z_i) = \frac{\exp(z_i)}{\sum_{k=0}^{K} \exp(z_k)}
\]

5. Experimental Results and Discussion

This section presents and discusses all the details regarding the experiments conducted to examine and evaluate the performance of the proposed system. Two problems are assessed through the experiment to present the availability of the proposed approach. To clarify how the proposed approach is used to derive advantages from unbalanced data.

5.1. Datasets and evaluation metrics

The datasets used for experiments were constructed based on a set of drug reviews and Twitter extract from different resources:

1. **AskaPatient**: contain set of 63,782 drug reviews. Each reviewer opinion contains on Condition, Side Effects (ADR), Dosage or Effectiveness for the used drug.

2. **WebMD**: Web-based health-related services containing a patients forum to share their experiences with medications. Each reviewer opinion contains on Condition, Side Effects (ADR), Dosage or Effectiveness for the used drug. It contains about 14,000 general drugs that have been crawled at the side of their associated overview totaling 241,980.

3. **DrugBank**: dataset contains a list of 215063 reviews on drug conditions, effectiveness, and side effects for the total number of drugs amount to 6345.

4. **Twitter**: Twitter can be accessed via its API from a range of popular programming languages using Tweepy in Python dataset consists of 267,215 Twitter posts that contain drug conditions and ADR, Indication, Beneficial reaction and Negative.

For performance evaluation, the **accuracy** measure corresponds to the ratio of correctly classified inputs to the whole number of inputs. Accuracy can be calculated as shown in equation (9), where \( TP \)
refer to the \textit{true positives}, i.e., the number of positive samples that were classified correctly as such; \( TN \) corresponds to the \textit{true negatives}, i.e., the number of false samples that were correctly classified as such; \( FP \) corresponds to the \textit{false positive}, i.e., the number of negative samples incorrectly classified as positive. Finally, \( FN \) corresponds to the \textit{false negatives}, i.e., the number of positive samples incorrectly classified as positive.

\[
\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}
\]

However, as the accuracy cannot accurately show the actual work performance performed on the unbalanced data, the Receiver Operating Characteristic (ROC) curve illustrates related trade-offs between true positive rates (TPR or recall), as shown in equation (13) and false positive rates (FPR or specificity), as shown in equation (10). Also, the Area Under Curve (AUC) allows the analysis of multiple classifiers, with the area equal to the probability that the classifier will score a randomly selected positive instance higher than a randomly selected negative instance. On the other hand, the F-measure is computed as the harmonic mean that combined effect of both the recall and precision measures as shown in equations (11), (12), and (13).

\[
\text{Specificity} = \frac{TN}{TN + FP} \tag{10}
\]

\[
F\text{-score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{11}
\]

\[
\text{precision} = \frac{TP}{TP + FN} \tag{12}
\]

\[
\text{recall} = \frac{TP}{TP + FN} \tag{13}
\]

Also, the average precision, recall, function measure, Residual Mean Square Error (RMSE): known as the standard error is the square root of the variance, and mean absolute error (MAE).

\[
\text{Mean absolute errors are calculated as an average of absolute mistake values, where the word “absolute” means “positive” in order that it can be added collectively.}
\]

\[
\text{MAE} = \frac{1}{n} \sum_{t=1}^{n} |e_t| \tag{15}
\]

5.2. Experimental Analysis

Simulation experiments in this paper are done on a PC with Intel Core i7 @ 2.7 GHz CPU and 32GB memory with 12GB GPU. We developed this system by using using the TensorFlow model of the Keras Python module and the Protégé OWL tool for Ontology.

The proposed multi-classification sentiment of the drug reviews system with 10-fold cross-validation was tested using different datasets. The used features for extraction are a combination of Ontology features with BERT and LDA topic modeling features. Moreover proposed system employed with LSTM to assess drug reason for taking and drug side effects and dosage with its sentiment.

Fig. 3 shows the ROC curve for the proposed approach with cross-validation, the utilized approach distributes each dataset from each one of the rest datasets by Area Under Curve (AUC), which is commonly used to evaluate the unbalanced dataset. Fig. 3 shows that the proposed approach is similar to the Y-axis, meaning it works best; i.e., it yields the most favorable ratio of false positives to true positives. The output of the ROC operator indicates that this approach obtained an AUC score of 0.975, 0.968, 0.951, and 0.94 for Askpaint, WebMD, DrugBank, and Twitter dataset, respectively. This means that a randomly selected positive review is rated higher than a randomly selected negative review with an estimated 98.2% probability. It is clear that this approach improved the Sentiment Analysis model, but this may not be sufficient to evaluate the approach. So, other metrics are used to evaluate the proposed approach. Table 1 shows the performance of the proposed approach using different datasets. The proposed approach achieved 98%, 97%, 98.2%, and 98% for precision, recall, F-measure, and accuracy respectively for
Table 1
Performance measures of the proposed approach using different datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
<th>accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>AskaPatient</td>
<td>0.951</td>
<td>0.942</td>
<td>0.940</td>
<td>0.96</td>
</tr>
<tr>
<td>WebMD</td>
<td>0.941</td>
<td>0.841</td>
<td>0.938</td>
<td>0.955</td>
</tr>
<tr>
<td>DrugBank</td>
<td>0.883</td>
<td>0.895</td>
<td>0.908</td>
<td>0.917</td>
</tr>
<tr>
<td>Twitter</td>
<td>0.848</td>
<td>0.843</td>
<td>0.845</td>
<td>0.929</td>
</tr>
</tbody>
</table>

AskaPatient datasets.

Figure 3. ROC curve of the proposed model with cross-validation

The result shows that the proposed approach is sufficient to distinguish and extract semantic and sentiment relations among objects from an online review, and it provides and help users to distinguish knowledge about drug.

5.3. Comparative analysis against state-of-the-art approaches

In this subsection, the proposed approach performance is compared against other state-of-the-art approaches in terms of feature extraction, which includes Ontology, Word2vec and LDA, Word2vec with LDA, and Ontology with LDA. Fig. 5 illustrates the results obtained by the system applied to a different approach. The result clarifies that Ontology with BERT and LDA (98%) has improvement against Ontology with Word2vec and LDA (96%), Ontology with Word2vec (90%), Word2vec with LDA (88%), and Ontology with LDA (83%). This high accuracy means that the proposed approach can identify the most important features from the dataset. The obtained results shows that the proposed approach outperformed all other models in terms of feature extraction accuracy due to its capability to obtain significant word semantics and sentiment relation with it’s topic and help to detect drug reaction.
Table 2
Proposed approach performance with different number of neurons

<table>
<thead>
<tr>
<th>Neurons</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>84.1</td>
<td>83.9</td>
<td>84.8</td>
<td>84.8</td>
</tr>
<tr>
<td>100</td>
<td>87.43</td>
<td>88.0</td>
<td>88.7</td>
<td>93.0</td>
</tr>
<tr>
<td>150</td>
<td>97</td>
<td>98.2</td>
<td>98</td>
<td>98</td>
</tr>
<tr>
<td>200</td>
<td>95.60</td>
<td>96.0</td>
<td>95.0</td>
<td>96.0</td>
</tr>
<tr>
<td>250</td>
<td>84.64</td>
<td>83.32</td>
<td>83.32</td>
<td>93.23</td>
</tr>
</tbody>
</table>

Figure 5. Accuracy of the proposed model against the other implemented approaches

It was found that the number of LSTM units impacts the efficiency of the results obtained. LSTM may not be appropriate for classifying complex data with a limited number of neurons. In this study, the impact of the number of LSTM neurons in two layers was checked. The results of the experiment showed that the Bi-LSTM neural network comprising 150 neurons in the classification task achieved better accuracy among them as in Fig. 6.

In Table ref tb: compare and ref tb: comp, the proposed approach is compared with other systems. The proposed BERT + Ontology + LDA as feature extraction achieved results 98 %, 97 %, 98.2 %, and 98 % for accuracy, precision, recall, and F-score, respectively, for the AskAPatient dataset. However, Word2vec + Ontology + LDA as feature extraction achieved results of 96 %, 95 %, 94.2 %, and 94 % for accuracy, precision, recall, and F-score, respectively, for the AskAPatient dataset. However, Word2vec with Ontology achieved results of 0.891, 0.9028 and 0.919 precision, recall and F-score, respectively. Also, BERT with the LDA method obtained results of 0.928, 0.944, and 0.956 precision, recall, and F-score respectively. Based on the obtained experimental effects, it’s concluded that the proposed approach outperformed both Word2vec with Ontology and LDA, BERT with LDA, and Ontology with LDA as shown in Table ref tb: compare. In Table ref tb: comp, an excellent output index for the proposed model is displayed by system error analysis. With the growth of the training data, the proposed approach shows a much lower MAE (0.13), as opposed to the other models; the MAE of Ontology with Word2vec, BERT with LDA, and Ontology with LDA has a slight increase. Based on this result, the proposed feature extraction model accuracy increased, and RMSE decreased. It is noticed that the proposed approach performed better with the extracting features task and classifying sentiments.

Fig. 7 shows the output of the ROC operator for Word2vec with Ontology and LDA. Using LSTM shows us that this approach received an AUC score of 0.942, 0.93, 0.901, and 0.889 for Askpaint, WebMD, DrugBank, and Twitter dataset, respectively. This means that a randomly selected positive review is rated higher than a randomly selected negative review with an estimated 94% probability.
Table 4

Comparative analysis of the proposed approach performance error measures against other tested approaches

<table>
<thead>
<tr>
<th>Method</th>
<th>Measure</th>
<th>Askapatient dataset</th>
<th>WebMD dataset</th>
<th>DrugBank dataset</th>
<th>Twitter dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word2vec+Ontology</td>
<td>RMSE</td>
<td>0.34</td>
<td>0.33</td>
<td>0.34</td>
<td>0.36</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.27</td>
<td>0.26</td>
<td>0.25</td>
<td>0.28</td>
</tr>
<tr>
<td>Word2vec+LDA</td>
<td>RMSE</td>
<td>0.36</td>
<td>0.35</td>
<td>0.39</td>
<td>0.37</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.22</td>
<td>0.23</td>
<td>0.27</td>
<td>0.21</td>
</tr>
<tr>
<td>BERT+LDA</td>
<td>RMSE</td>
<td>0.33</td>
<td>0.3</td>
<td>0.31</td>
<td>0.32</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.19</td>
<td>0.2</td>
<td>0.21</td>
<td>0.21</td>
</tr>
<tr>
<td>Ontology+LDA</td>
<td>RMSE</td>
<td>0.43</td>
<td>0.37</td>
<td>0.41</td>
<td>0.33</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.2</td>
<td>0.26</td>
<td>0.24</td>
<td>0.23</td>
</tr>
<tr>
<td>Word2vec+Ontology+LDA</td>
<td>RMSE</td>
<td>0.28</td>
<td>0.3</td>
<td>0.29</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.16</td>
<td>0.17</td>
<td>0.16</td>
<td>0.15</td>
</tr>
<tr>
<td>BERT+Ontology+LDA (Proposed approach)</td>
<td>RMSE</td>
<td>0.25</td>
<td>0.28</td>
<td>0.27</td>
<td>0.27</td>
</tr>
<tr>
<td></td>
<td>MAE</td>
<td>0.14</td>
<td>0.13</td>
<td>0.14</td>
<td>0.15</td>
</tr>
</tbody>
</table>

Figure 8. ROC curve for Word2vec with Ontology with cross-validation

Fig. 8 shows the ROC curves for the best features for Word2vec with Ontology using LSTM. Fig. 9 shows the ROC curve for the best feature, Word2vec with LDA. From the ROC curve as shown in this figure, the proposed system achieved the best performance another system.

Fig. 8 shows that the Word2vec with Ontology method is the Y-axis, which yields the most desirable ratio between false positives and true positives. The output of the ROC operator shows us that this approach received an AUC score of 0.893, 0.84, 0.807, and 0.797 for Askpaint, WebMD, DrugBank, and Twitter dataset, respectively. This means that a randomly chosen positive review is rated higher than a randomly chosen negative review with an estimated 89% probability.

Fig. 9 shows that the Word2vec with LDA method is on the Y-axis, which yielded the most desirable ratio between false positives and true positives. The output of the ROC operator shows us that this approach received an AUC score of 0.868, 0.803, 0.7901, and 0.779 for Askpaint, WebMD, DrugBank, and Twitter dataset, respectively. This means that there is roughly an 86% probability we
will rank a randomly chosen positive review higher than a randomly chosen negative review.

Table 5 shows the performance of the proposed approach against different ML approaches. Feeding SVM with the extracted features by the proposed approach obtained 0.82, 0.84, 0.83, and 0.85 for precision, recall, F-score, and accuracy, respectively. While feeding Logistic regression with the extracted features by the proposed approach obtained 0.78, 0.87, 0.79, and 0.79 for precision, recall, F-score, and accuracy, respectively. Feeding Random Forest with the extracted features by the proposed approach obtained 0.82, 0.8, 0.85, and 0.85 for precision, recall, F-score, and accuracy, respectively. On the other hand, feeding CNN with the extracted features by the proposed approach obtained 0.79, 0.84, 0.87, and 0.89 for precision, recall, F-score, and accuracy, respectively. It is concluded from Table ref tb: Comparativeanalysis that the best results were obtained when feeding Bi-LSTM with the extracted features by the proposed approach, where Bi-LSTM outperformed SVM, Logistic regression, Random Forest, and CNN in all performance metrics. This is due to the nature of Bi-LSTM, which has a memory feature in context and thus designs text patterns in two directions that influence the decision to identify sentiment’s polarity characteristics.

The comparative study presented in this paper shows that the combination of Bi-LSTM, ontology with BERT, and LDA is effective in classifying drug sentiment. Traditional methods ignore the meaning of the traditional text, leading to erroneous polar classification decisions.

Fig. 10 presents a comparison among the proposed approach using Bi-LSTM, CNN, SVM, Logistic Regression, and Random Forest Machine Learning classifiers [27]. CNN implements a multi-layer network with a sigmoid activation function using the TensorFlow model of the Keras Python module [28], Random Forest (a meta estimator that outfits multiple decision tree classifiers on different dataset sub-samples and uses an average to improve predictive accuracy and overfitting control), and Logistic regression (Regression analysis can predict class results based on some predictions). A multinomial logistic regression model was used in terms of training from 0 to 50 epochs. It is noticed that the performance of all used ML techniques starts to be improved after 10 epochs, except SVM, it seems that the performance in all cases is balanced after 25 epochs. The SVM swings accuracy for more than 20 times, due to its sensitivity and varying weights after each training vector [29].

Based on the results, the proposed approach outperformed the current lexicon-based approaches, statistic-based systems, machine learning approaches, and deep learning approaches. This is because of that the current lexicon-based approaches and statistic-based systems depend on the dictionary that not contains all terminologies resulting in poor experimental results. The proposed approach presents a hybrid lexicon-based approach with word and sentence embedding to overcome drawbacks in the current approaches. The proposed approach obtained a higher F1-score. This is most likely due to the enrichment of extracted features using BERT with ontology, the extraction of the semantic word and their relationships from ontology followed by the prediction and estimation of the semantics similarity between entities on ontology using BERT, and finally, the extraction of the accurate topic using LDA. This increases the performance compared to other models, adds context to the prediction task, and combines several different features to improve sentiment classification performance.

Figure 10. Comparative results between the proposed approach against several machine learning and deep learning classifiers

Table 6 presents a comparison between the proposed approach and other sentiment classification systems.
Table 5

<table>
<thead>
<tr>
<th>Model</th>
<th>Avg. Precision</th>
<th>Avg. Recall</th>
<th>Avg. F-measure</th>
<th>Accuracy</th>
<th>RMSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hybrid Feature Learning+SVM</td>
<td>0.82</td>
<td>0.84</td>
<td>0.83</td>
<td>0.85</td>
<td>0.43</td>
<td>0.36</td>
</tr>
<tr>
<td>Hybrid Feature Learning+Logistic Regression</td>
<td>0.78</td>
<td>0.87</td>
<td>0.79</td>
<td>0.79</td>
<td>0.44</td>
<td>0.38</td>
</tr>
<tr>
<td>Hybrid Feature Learning+Random Forest</td>
<td>0.820</td>
<td>0.8</td>
<td>0.85</td>
<td>0.85</td>
<td>0.33</td>
<td>0.41</td>
</tr>
<tr>
<td>Hybrid Feature Learning+CNN</td>
<td>0.79</td>
<td>0.84</td>
<td>0.87</td>
<td>0.89</td>
<td>0.29</td>
<td>0.3</td>
</tr>
<tr>
<td>Hybrid Feature Learning +Bi-LSTM (Proposed)</td>
<td>0.981</td>
<td>0.972</td>
<td>0.982</td>
<td>0.98</td>
<td>0.14</td>
<td>0.13</td>
</tr>
</tbody>
</table>

Table 6

<table>
<thead>
<tr>
<th>Study</th>
<th>Dataset</th>
<th>Methods</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>[30]</td>
<td>Twitter</td>
<td>Extract the event entities from Adverse drug events(ADE) based on lexicon-based approach</td>
<td>matching rates ranging from 74% to 88%</td>
</tr>
<tr>
<td>[31]</td>
<td>UMLS, CHV, FAERS, MeSH</td>
<td>Lexicon Approach with Support Vector Machines (SVM)</td>
<td>F-score increased to above 80%</td>
</tr>
<tr>
<td>[32]</td>
<td>tweets and forum posts</td>
<td>Use ADRMine to extract feature and use CRF as classifier</td>
<td>achieved F-measure 79.57% to 80.14% for DailyStrength part</td>
</tr>
<tr>
<td>[33]</td>
<td>Twitter Posts</td>
<td>word-embedding with RNN model that labels words in an input sequence with ADR membership tags</td>
<td>F-measure of 0.755% for ADR</td>
</tr>
<tr>
<td>[34]</td>
<td>Tweets</td>
<td>Use semi-supervised CNN to classify ADE</td>
<td>70.21% precision, 59.64% recall, and 64.50% F-score</td>
</tr>
<tr>
<td>[35]</td>
<td>Electronic Health Records</td>
<td>Joint AB-LSTM With Embedded Lemmas for Discovering ADR</td>
<td>achieving an f-measure of 73.3%</td>
</tr>
<tr>
<td>[36]</td>
<td>Clinical notes</td>
<td>use long short-term memory (BiLSTM) and conditional random fields (CRF) neural network to detect medical entities relevant to Adverse drug events(ADE)</td>
<td>achieved F measures of 0.83% for ADE-relevant medical entity detection and 0.87% for relation detection.</td>
</tr>
<tr>
<td>[9]</td>
<td>AskaPatient.com.</td>
<td>combines CNN and bi-directional long short-term memory (Bi-LSTM) to detect Adverse Drug Event</td>
<td>accuracy 85.67 , F-score 85.57%, Precision 86.88%, Recall 84.16%</td>
</tr>
<tr>
<td>Proposed approach</td>
<td>WebMD, AskaPatient, DrugBank, and Twitter Posts</td>
<td>Ontology with Word2vec and LDA to extract feature and bi-LSTM to classify data</td>
<td>accuracy of 96% and AUC score of 93 and F-score of 0.94 for used datasets</td>
</tr>
</tbody>
</table>
5.4. Discussion

5.4.1. Results analysis and Implications

As stated before, the proposed approach aims to extract the most valuable information from social data and helps users to make decisions, and detects knowledge from text to provide word semantic meaning. The proposed framework modeled with (Ontology with BERT and LDA and Bi-LSTM) obtained a better result than existing methods. The proposed approach achieved higher accuracy in terms of defining features in the content on social media compared with the extraction approaches. The existing LDA method lacks useful features during topic generation due to the limited dataset. Therefore, the proposed approach combines LDA with sentence embedding and Ontology for feature extraction. Based on the obtained results, it is observed that the performance of proposed systems increases when using Ontology with word embeddings and LDA topic modeling as feature extraction and Bi-LSTM as a classifier. The proposed approach achieved an accuracy of 98%

Up to our knowledge, this is the first paper applying various experiments on conservative social media datasets like Twitter and social media forums datasets about Drugs to evaluate the effectiveness of the proposed approach. In contrast to previous research, the proposed approach uses a hybrid feature extraction with deep learning methods that detect and improve word value in the document that makes better guessing of context words, and improves the extracted feature. Also, Bi-LSTM is used to perform text feature learning, through updating and providing sufficient results of sentiment classification. Finally, The proposed approach introduces the use of Fuzzy Ontology and LDA with BERT sentence embeddings that are capable of collecting greater elements of sentiment and semantic in comparison to other systems. The proposed approach and its results can be extended and spread through several activities for information extraction. It will solve any other named entity recognition problems and determining the capability of LDA with BERT and the use of a fuzzy ontology to enhance current models.

5.4.2. Limitations

A limitation is that Sentiment Analysis and detection can be considered across languages, and a potential multilingual model can be created, but it was not available during training and data collection for the proposed approach where there are restrictions on appropriately label datasets. Another limitation is the widespread use of knowledge extracted from Ontology. A large number of concepts and their relationships may lead to a large extent of semantic knowledge, which makes classification methods complex.

6. Conclusions and Future Work

In this paper, a sentiment classification approach based on an Ontology with the word, sentence embeddings (BERT), and LDA topic modeling is proposed to enhance the document representation performance. The proposed approach has many of the key issues involving the use of valuable information, converting the extracted data into useful knowledge, and generating the topics and features using Ontology with BERT and LDA. The proposed approach presents a sentiment classification system that establishes the most relevant texts in social media and analyzes them. It will not only improve the performance of the feature extraction but also outperform the methods of representation of documents with different datasets. It combines the lexicon with a pre-trained BERT embeddings system, which improves the sentiment classification accuracy. The Ontology with BERT and LDA is used to extract information on drug interactions of content via the internet and can be used to determine the most common cause. Based on the obtained results, it is observed that the performance of the proposed approach increases when using Ontology with BERT and LDA topic modeling as feature extraction and LSTM as a classifier. The proposed approach achieved an accuracy of 98%.

In future research work, the classification performance will be improved by analyzing different content and new feature extraction approaches should be utilized.

References


