Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 05 Issue 01
2023

Research Article
https://doi.org/10.56979/501/2023

Knowledge Graph Embedding Based Sentiment Analysis of Product Reviews using LSTM and Fuzzy Logic

Umaira Khurshid[1], and Imran Ihsan[1,*]

[1]Department of Creative Technologies, Air University, 44000, Pakistan.
[*]Corresponding Author: Imran Ihsan. Email: iimranihsan@gmail.com

Received: March 19, 2023 **Accepted:** May 01, 2023 **Published:** June 05, 2023

Abstract: The popularity of online buying has skyrocketed, particularly during and after the COVID-19 era. Product reviews are a huge asset when making decisions about online purchases. Product reviews can also aid in product improvement from the standpoint of retailers and producers. It takes a lot of effort and time to read through each. With the aid of sentiment analysis techniques, researches have experimented with analysing product reviews. Intelligent systems that use machine learning and deep learning algorithms are in high demand because they allow customers to quickly uncover the trend in a product rather than having to read through many evaluations. The accuracy of these intelligent systems is still debatable, though. Different researchers have modified various strategies in the past to increase accuracy. This study is another effort in that similar approach. But this study takes a somewhat different track. Product review analysis is an NLP problem, therefore textual input is prepared before being fed into a deep learning model using knowledge graph embedding. We think that using a deep learning model like LSTM in conjunction with knowledge graphs can produce the much-needed system with increased recall, precision, and accuracy. The classification of three Amazon datasets from Kaggle into the four sentiment categories of "Positive", "Negative", "Highly Positive", and "Highly Negative" is the purpose of this study. Amazon product review benchmark datasets like "Customer reviews of Amazon Products," "Amazon Cell Phones Reviews," and "Amazon Fine Food Reviews" were used to test the suggested LSTM+KGemb model. Results are achieved with accuracy levels of 97%, 98%, and 96%, respectively.

Keywords: Knowledge Graph; In-text Citation; Data Mining; Entity Recognition.

1. Introduction

News stories, blogs, reviews, and comments offer opinionated and current data about services, products, and topics for sentiment analysis on social media platforms. Due only to the Covid-19 epidemic, there were 222 million more internet users worldwide in a single year. Sentiment analysis is crucial because consumers want to know the [1] benefits and drawbacks of a product before making a purchase. Numerous items with nearly identical specifications exist, which could confuse consumers while making purchases. Customers and manufacturers both benefit from sentiment analysis. Social media customer feedback helps businesses design products that meet customer needs, and information from social media about the products available for online shopping has grown to be an important resource for helping customers make better purchasing decisions and close big deals [2]. There are several methods [3] for resolving this issue, but not enough to offer more precision. Because of this, many AI approaches using machine learning and deep learning techniques have been tested on various benchmark datasets to investigate the thoughts that customers express when using web-based entrances. Therefore, AI systems can be developed to assist businesses and consumers with the sentiment analysis of a particular product.

The proposed paradigm in this study is grounded in knowledge graphs, and knowledge graph embedding. Amazon product review benchmark datasets like "Customer reviews of Amazon Products," "Amazon Cell Phones Reviews," and "Amazon Fine Food Reviews" were used to test the suggested LSTM+KGemb model. Three Amazon product review datasets, namely, were the basis for the sentiment analysis dataset. The datasets received preprocessing, which included text normalization and tokenization procedures, before analysis. Tokenization was used to separate the text into tokens or words, and text normalization made sure the text format was consistent. After preprocessing, the datasets were prepared for additional analysis. The links between entities and attributes in the datasets for product reviews were represented using knowledge graphs. From the reviews, entities, relations, and characteristics were taken out and organized into a graph as part of the process.

The generated graphs gave the data a visual form, permitting further analysis and embedding in knowledge graphs. The TransE model was used for knowledge graph embedding to improve the representation of the knowledge graphs. By retaining the semantic connections found in the knowledge graphs, the TransE model seeks to learn vector representations of entities and relations. Using a recurrent neural network (RNN) variant called Long Short-Term Memory (LSTM), sentiment analysis was performed on the embedded knowledge graphs. LSTM models can capture long-term dependencies, making them ideal for processing sequential data. Multiple layers with hidden units made up the LSTM architecture, which has been used to learn the sentiment patterns seen in the knowledge graph embeddings. The reviews are divided into three categories by LSTM: good, negative, and neutral. Fuzzy logic was incorporated into the algorithm to improve sentiment classification even more.

2. Literature Review

Sentiment analysis and opinion mining are prominent fields of study in the modern day. The proper product can be chosen by customers using a variety of artificial intelligence techniques, however there are still numerous issues that require further development. We will talk about similar work on sentiment analysis and opinion mining in the part that follows.

The RNN-LSTM model for sentiment analysis [13] with Long short-term memory (LSTM) and a bag of words with RNN and other hyperparameters are used to limit the feature space to 5000 words. Reviews ranged from 1 to 2055 words, on average 147 words. Amazon.com's 88239 reviews used as one of the data sources for this essay. This project's main contribution is a performance comparison of models employing both their own training vectors and pre-trained GloVe vectors. Precision was 0.55%. Another model [14] combined RNN and LSTM in conjunction with Softmax Classifier because the network uses category cross entropy to determine the accuracy of both positive and negative evaluations separately. Both positive and negative accuracy were 49.31% and 92.71% respectively. 13877 tweets and datasets collected from Twitter were utilized to test and train the model.

Wei Zhao et al. [15] proposed two weakly supervised deep embedding (WDE-CNN) and weakly supervised deep embedding (WDE-LSTM) models for sentiment analysis. The initial phase in this framework is embedding space, which separates weakly and strongly labelled texts. The second phase involves adding a classification layer on top of the embedding layer. The vast dataset of Amazon reviews was used to train and test these models. 1.1 million reviews with weak labels and 11754 phrases with labels were used. WDE-CNN accuracy was 0.877 and WDE-LSTM accuracy was 0.879. Additionally, they compared the two models. James Barry[16] suggested a model using several word embedding methods. He employed LSTM GloVe, LSTM Word2vec-domain, LSTM Self Initialised, and LSTM Word2Vec. The main contribution was that they used various LSTM techniques to test and train data on various features, including Word2Vec and GloVe. Words are translated into integers with a maximum length of 1000 and the data is divided 80:20. Embedding matrices were then employed. 568,454 reviews from the Amazon.com food review datasets were used. The accuracy of LSTM Word2vec was 95.75%, that of LSTM Self Initialised was 94.86%, that of LSTM GloVe was 95.84%, and that of LSTM Word2vec-domain was 95.75%. LSTM GloVe operate effectively.

Convolution-EMB and LSTM-EMB [17] perform character level embedding using various hidden layers and embedding sizes. performed sentiment analysis across languages. During training, the model only learns each language once. Model testing and model training have been done using a larger dataset of 1.6 million tweets. Convolution-EMB's accuracy was 0.714 and LSTM-EMB's accuracy was 0.713. LSTM with

Token2Vac and a sizable dataset of 7.8 million Amazon.com reviews were employed by Achira Jeewaka Shamal et al. in their study [18]. Accuracy was 88.20%, and the system used pre-trained RNN model with LSTM cells and two matrices for input and output. We used 500000 unlabeled user reviews. Emojis, acronyms, and phrases were embedded in one vector space using Token2Vec, which was a significant contribution.

The Open Information Extraction (OpenIE) uses binary relations between the authors to generate KGs [13]. Instead of manual curation, the OpenIE automatically extracts all semantic relational tuples from source text [17]. To achieve this goal, they have developed methods for enhancing the extraction and connection of named items with KG persons, as well as their association with grammatical units that result in the generation of more cohesive facts. Furthermore, they have provided choices to construct potentially helpful RDF triples for the KG. Results have shown that using grammatical structures together with information extraction units improves comprehension of the proposition-based representations in the building of KGs. Another proposed CNN-LSTM model [19] uses a skip-gram data sample that is divided into a 75:25 ratio to perform word embedding. Binary cross entropy was employed for the cost function. Accuracy and re-views on Amazon.com were 40.8% and 41.6%, respectively. The comparison of Naive Bayes, SVM, and CNN-LSTM on two datasets is a key addition. Sentiment analysis was carried out by Hanxue Ji et al [20] using LSTM. A single vector divides the entire document. Larger unlabeled data were used for training. Using attention-based LSTM framework. Framework that is semi-supervised was suggested. Unsupervised encoder-decoder structures were utilized in the first stage to convert documents into vectors, and supervised LSTM-based networks with SoftMax layers were then used for feature extraction and classification. Model developed and trained using 607,551 Amazon reviews with accuracy of 85%.

LSTM and CNN were suggested by Momina et al [12] for sentiment classification. Reviews were classified using unigram classifiers. Confusion measures were applied to assess the classifiers' performance. used 400,000 reviews from the dataset on Amazon.com The model is tested using various classifiers, and results are provided for accuracy, F1, recall, and precision for each classifier. LSTM accuracy is 73.7%, whereas CNN accuracy is 77.5%. LSTM and LSVM were proposed by Emilie Coyne et al[22] and utilized on a sizable dataset of 3.94 million reviews from amazon.com. After preprocessing, feature extraction approach was used to convert text data to numerical data. Compare MNB, LSVM, and LSTM before assessing LSTM's performance on a different dataset. LSTM and LSVM accuracy are 86% and 90%, respectively. There are four models that Mehmet Umut et al[23] offered, including CNN, Bi-LSTM, and CNN-Bi-LSTM. The proposed model made use of trees. In this first branch, the components such words, emojis, numbers, and symbols were finely tuned, and the second branch would perform the classification. The SoftMax layer is the root classification layer. 17,289 tweets were used as datasets, with accuracy ranging from 75.67% to 80.44% to 82.14%. By utilising the CNN extract feature of 3-grams, 4-grams, and 5-grams with fixed length, Kiran R. a. et al. [24] proposed CNN-BiLSTM. After combining various models, the model's outputs are fed into the following method, which then captures long-term dependencies and combines the CNN and LSTM features with a single optimizer into a single architecture. This model is examined using six Amazon.com datasets. The model's accuracy is 96.9%.

The LSTM model was proposed by Amol C. Adamuthe[25] and the skip gramme model is utilized as its input. Wor2Vec is employed for vector space. On seven full datasets of Amazon reviews, six experiments are conducted. The model's accuracy is 87.3%. Sentenced level sentiment classification was carried out using stacked neural networks, as proposed by Ganesh More[26]. Based on ranking, a model divides reviews into good and negative categories. For vector conversion, Word2Vec is utilized. Performance of LSTM, GRU, and LSTM-GRU are compared. 5067,073 reviews on Amazon.com were used to train the model, and its accuracy was 74%.

By integrating the syntax dependency of the text that can be used as input to extract information, Zuhua Dai[34] proposed GCN and Knowledge graph. Aspect-level sentiment analysis was performed. The accuracy for the SemEval2014 dataset was 79.54%. On Amazon.com, Julio Vizcarra[35] employed knowledge graph embedding reviews. Disambiguation is used to improve knowledge graphs created after data collection. Semantic processing employs knowledge graphs, semantic similarities, disambiguation, and graph theory algorithms. Wen Chen[36] used knowledge graphs and support vector machines to analyse data from TripAdvisor. From online travel sites, keywords are taken, and KG is created. Word2vac was employed to extract. There have been two different sampling techniques utilized. Modelling Syntax and Knowledge were implemented by Jie Zhou[37] combined with a graph convolution network. Commonsense knowledge and the syntactic dependency tree are employed by SK-GCN1 and SK-GCN2. 5 datasets were used to apply pre-trained BERT. Model accuracy was 75% after being tested and trained on five Twitter datasets. Knowledge graph embedding was applied by Abhishek Kumar[19]. Combining two Layered Attention Networks results in a supervised model called support vector regression.

3. Methodology

The opinionated data created by the user has a double-edged sword in that it is more challenging to extract relevant information the more data there is. Even while data is readily available, it can be challenging for businesses and consumers to assess how the public feels about a given product. To accomplish accurate sentiment classification with contextual understanding, a comprehensive framework for sentiment analysis using LSTM Models, knowledge graphs, and knowledge graph embedding is being developed. Figure 1 depicts the suggested methods for this objective.

Figure 1. Proposed Methodology of In-Text Citation Knowledge Graph Generation.

3.1. Dataset

3.1.1. Customer Reviews of Amazon Products (CRAP)

There are 34,000 reviews in this dataset. The devices being reviewed by Datafiniti include the Kindle, Fire TV stick, and many others. This collection includes basic data like review and rating text.

3.1.2. Amazon Cell Phones Review (ACPR)

The reviews of several mobile phone manufacturers, including Samsung, Nokia, Huawei, Sony, ASUS, Google, Motorola, OnePlus, and Xiaomi, are included in this dataset. There are 67,986 reviews in the ACPR, each of which includes a review paragraph and the date it was posted.

3.1.3. Amazon Fine Food Reviews (AFFR)

The AFFR dataset, which has 568,454 reviews, is about reviews of fine food. Similar to that, this information also includes the review's paragraph and time of posting.

3.2. Data Preprocessing

Pre-processing is the cleansing and elimination of noise or extraneous data [1]. A reviewer's name, reviewer ID, time, rating, Summary, review text, and category are all included in all three datasets. With the use of a lexicon, non-opinionated and unnecessary data will be removed by labelling the subjective sentences. Text normalization activities include dictionary mapping, spelling checks, and machine translation. Tokenization [1], converting text into lower tokens, text normalization such as spelling correction, lemmatization [1], and breaking up big phrases are only a few of the jobs that make up pre-processing. Tokenization is the process of breaking the sentence up into words. Sentences will then be changed to lowercase. Afterward, spelling checks and dictionary mapping will be completed. Lemmatization is another name for dictionary mapping. Long sentences will be broken up into manageable chunks.

3.2.1. Tokenization

The process of tokenization is the division of words into discrete units called tokens [1]. In addition, the text is free of numerals and punctuation. Filters are used to get rid of numerals, punctuation, and shorter words.

3.2.2. Lower Case

In this pre-processing stage, all words, or sentences [1] are changed to lower case. The extraction of tokens includes both long and brief words.

3.2.3. Normalization

Text normalization is the process of preparing text for automated processing. The Stop-Word removal allows for POS labelling and stemming. For the algorithm to recognize each word in the phrase, the spelling of the words could [1] be checked and fixed before moving on. The method of condensing the word "spatiality" is called lemmatization [1]. For instance, the word "Dark" is used to refer to phrases like dark, darker, and darken.

3.2.4. Sentence Splitting

Long sentences will be broken up into meaningful chunks.

3.3. Knowledge Graph Construction

Unstructured data is modelled using knowledge graphs (KG), which link data in meaningful ways. Knowledge graphs, in their most basic sense, are technologies for storing complex structured and unstructured data or information [5]. It is sometimes referred to as a network since it has nodes and edges, where edges stand in for the connections between entities and nodes for the actual entities themselves.

3.3.1. Segmentation

Data from product reviews may contain a variety of feature descriptions, so data will be translated into several sentences during sentence segmentation [39][40]. Extracted information will be evaluated collectively. In this step, the subject and the object will both be evaluated.

3.3.2. Entity Extraction

Entity extraction is a method that may be used to reveal the primary subjects of interest quickly and readily to gain some understanding of unfamiliar data sets. One advantage that analysts gain from having access to such a structured corpus as a starting point for future research and analysis is the ability to view all entity kinds (such as names, companies, and nations) in one place [39].

3.3.3. Relation Extraction

Relationship extraction is the technique of determining the informal relationships between words in a text. The extracted relationships can be categorized using a variety of semantic categories, but they all involve two or more entities of the same type (such as people, organizations, or actual locations), such as married to, employed by, or residing in [39].

3.3.4. Knowledge Graph Building

The final stage is to construct the knowledge graph following phrase segmentation, entity extraction, and relationship extraction. Nodes, edges, and the connections between them are contained in a knowledge graph [40]. Algorithms based on graph theory are utilized for knowledge reasoning.

3.4. Sentiment Analysis

3.4.1. Knowledge Graph Embedding

Since knowledge graphs and knowledge graph embedding may be used to many different fields, including artificial intelligence, machine learning, deep learning, and natural language processing, they have become quite popular among researchers. By analyzing graph data and enabling the effective, knowledge graph embedding represents knowledge graph as continuous vectors in a high dimensional space. Knowledge graph embedding is the process of converting items and relationships into low-dimensional vector spaces [41-50]. Sentiment analysis is one of several activities that uses knowledge graph embedding. The knowledge graph triplets are adjusted based on idea graph information items. Information in the form of a head entity, relation, and tail entity is called a triplet. Many knowledge graphs embedding techniques, including TransE, pTranSe, and TransH, are employed for embedding.

TransE is one of the most well-known and often used models in knowledge graph embedding. In essence, translation-based embedding is what it is. It is used to record the semantics of relationships between things in knowledge graphs. In the TransE paradigm, each entity is represented as a continuous vector. A translation from the head entity to tail entity vector is what the relationship vector in TransE looks like. As this embedding model seeks to identify the vector representation for each triple in entity, relationship, and tail in k knowledge graph, translation from head entity vector comes closer to the tail entity vector by relationship vector. TransE model has demonstrated promising outcomes in knowledge graphs by capturing the structural regularities and semantic patterns.

3.4.2. LSTM Model

Long Short-Term Memory, or LSTM, is frequently employed for sentiment classification. The vanishing gradient issue in standard RNN can be handled with LSTM by catching the long-term dependencies. In essence, it is made up of three gates: input, forget, and output. Input 0 indicates that all the information has been entirely dumped, and output 1 indicates that all of the information has been totally kept. Forget gate filter the information to dump from cell by using previous and current values. New vector values produced by the tanh layer with new Cti values after updating the data from the previous two steps. The output gate is formed by executing the sigmoid layer after the new state has been updated by multiplying with the forget gate it*Cti. After knowledge graph embedding, the LSTM model will be used to classify sentiment and provide sentiment scores. The LSTM model will be used to generate sentiment scores that are positive, negative, and neutral. The RNN version known as LSTM uses prior output as input. A call state vector value is used by LSTM at each phase. Binary gates, input gates, forget gates, and output gates are all present [31].

3.5. Fuzzy Logic

Sentiment score generation will be followed by phrase polarity recognition using a lexicon-based method. Fuzzy logic is utilized to categorize the subsequent score generation into four categories: very positive, highly negative, negative, and positive as the language could not be entirely identified as either positive or negative. In the fuzzification module, the emotion score values of each word are transformed into a fuzzy set. In which a fuzzy set is used to represent a member function or linguistic variable. The sentences are categorized into positive, negative, extremely positive, and highly negative using a rule-based method. Defuzzification process resultant fuzzy set converted set of values [1].

4. Experiments

The implementation, experimental findings, and discussion of the results for sentiment analysis of three datasets of Amazon product reviews using LSTM, Knowledge graphs, Knowledge graph embedding, and Fuzzy Logic are the main topics of this section. The detailed process for preprocessing the datasets, creating knowledge graphs, performing knowledge graph embedding with the TransE model, applying LSTM for classification into three sentiment classes (positive, negative, and neutral), and incorporating fuzzy logic for additional classification into four sentiment classes (positive, negative, highly positive, and highly negative) is presented in this section.

Utilizing Google Colab and the most recent Python version, the implementation was completed. Three Amazon product review datasets, namely, were the basis for the sentiment analysis dataset. The datasets received preprocessing, which included text normalization and tokenization procedures, before analysis. Tokenization was used to separate the text into tokens or words, and text normalization made sure the text format was consistent. To complete these tasks, Python's libraries like NLTK or SpaCy were used. After preprocessing, the datasets were prepared for additional analysis.

The links between entities and attributes in the datasets for product reviews were represented using knowledge graphs. From the reviews, entities, relations, and characteristics were taken out and organized into a graph as part of the process. The generated graphs gave the data a visual form, permitting further analysis and embedding in knowledge graphs. The TransE model was used for knowledge graph embedding to improve the representation of the knowledge graphs. By retaining the semantic connections found in the knowledge graphs, the TransE model seeks to learn vector representations of entities and relations. TensorFlow or PyTorch were used as frameworks for the Python implementation.

To enhance the representations of entities and relations within the knowledge graphs, hyperparameters were tweaked during training. Using a recurrent neural network (RNN) variant called Long Short-Term Memory (LSTM), sentiment analysis was performed on the embedded knowledge graphs. LSTM models can capture long-term dependencies, making them ideal for processing sequential data. Multiple layers with hidden units made up the LSTM architecture, which was used to learn the sentiment patterns seen in the knowledge graph embeddings. On the preprocessed datasets, the LSTM model was implemented and trained using Python tools like TensorFlow or PyTorch.

To evaluate the effectiveness of the implemented system, training and evaluation were conducted. A training set, a validation set, and a testing set were created from the dataset. Utilizing the proper loss functions and techniques, the LSTM model and fuzzy logic system were optimized during training. The system's performance was gauged using evaluation measures such F1 score, recall, accuracy, and precision. The evaluation's findings revealed information on the sentiment analysis system's precision and efficiency.

4.1. Experimental Setup

4.1.1. Hyper Parameters

The proposed model's performance is improved by changing various hyperparameters. Hyper settings are configurable and directly control the training process. Before the training process starts, they are set. Hyperparameters have a significant impact on the proposed model's learning, stability, and accuracy.

4.1.2. LSTM Layers

All three datasets underwent preprocessing to build knowledge graphs, which were then used to perform feature extraction using the knowledge graph embedding model TransE. Dataset was divided into three categories using LSTM (Positive, Negative, and Neutral). Three layers make up LSTM: the LSTM layer, the input layer, and the output layer. To analyse the performance after training the embedding layer, 128 LSTM layers were used. Three nodes were selected to provide sentiment ratings since we employed three classes (Positive, Negative, and Neutral). We used 100 memory units in our suggested model, which uses memory units to help people remember the words from sentences in dataset reviews.

4.1.3. Activation Function

Sentiment analysis using a knowledge network is a multi-label classification problem. We employed the SoftMax activation function, which is the most appropriate activation function, to handle the multi-label categorization.

4.1.4. Epochs

On 100 epochs, the proposed model was tested. The quantity of epochs generally refers to the quantity of LSTM model iterations on the input dataset. When there were 100 epochs, the suggested model had good accuracy. On 100 epochs, the proposed model performed well. The model failed to recognize extremely positive sentences on the fuzzy logic phase after training on 10, 30, and 50 epochs. On 100 training epochs, the suggested model correctly identified reviews that were extremely favorable.

4.1.5. Batch Size

On each of the three benchmark datasets of Amazon product evaluations, the proposed model experimented with batch sizes of 32. The batch size determines how many sentences from reviews the LSTM network will process.

4.1.6. Optimizer

The LSTM network's learning rate optimizer is tuned. We applied the Adam optimizer to the model we proposed. The key parameters and experimental setting used in the implementation process are detailed in the table below.

4.2. Knowledge Graph

Knowledge graphs were produced for each benchmark dataset of Amazon product reviews following data preprocessing. First, dependencies for each sentence in the reviews text of each dataset were identified throughout the knowledge graph construction process. To extract entities from the review text, we subdivided each sentence in the second phase. When extracting entities, predefined rules are applied. Relationship was derived from the reviews' content using a specified pattern after entity extraction. The pattern-matching words have been retrieved as relationships. For the dataset CRAP, there were 344 relationships for the relationship that "is great," 1061 relationships for "recommend," and 1161 relationships for "love." For each dataset of Amazon product reviews, knowledge graphs were constructed utilizing Source, Target, and Edge in the final stage of the process. Figure shows visualization of knowledge graphs.

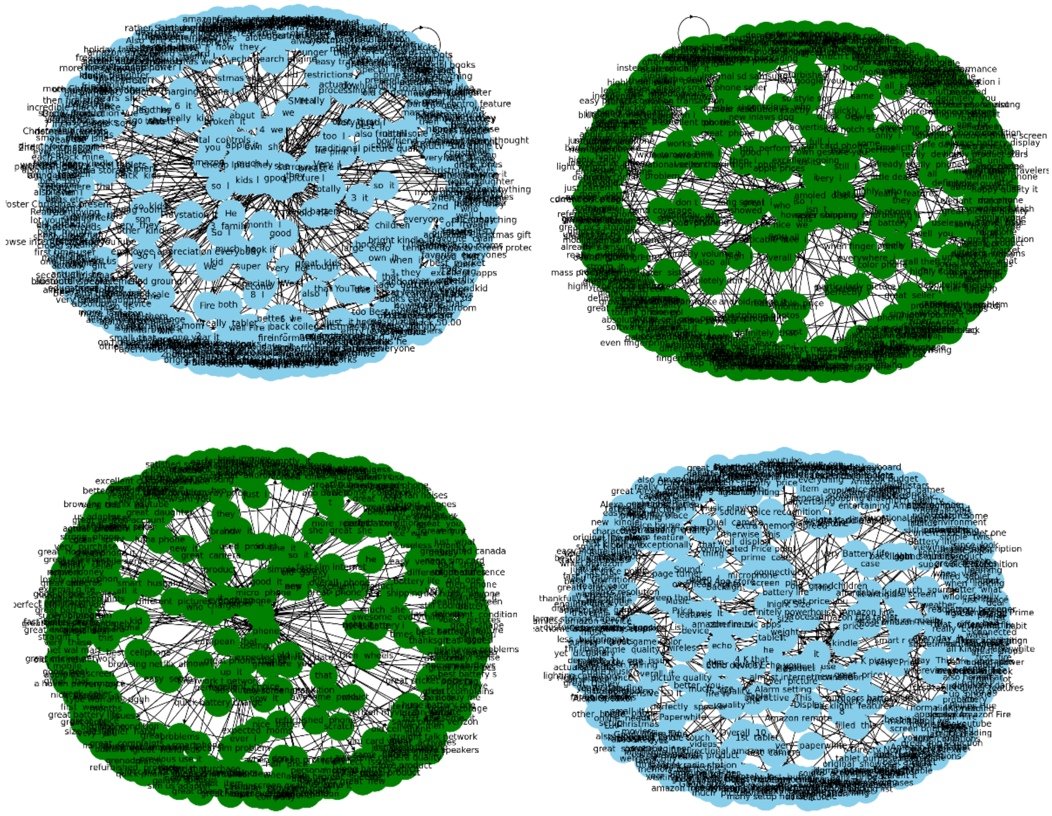

Figure 2. Entity Extraction from ACL Dataset.

4.3. Fuzzy Logic

After using fuzzy logic to divide the dataset into four categories—positive, negative, very positive, and highly negative—LSTM was used to divide the Amazon reviews dataset into three categories: positive, negative, and neutral. The major goal of this fuzzy logic classification is to detect those sentences and categorize them as positive and extremely bad reviews because some sentences cannot be classified as positive or negative.

5. Results

5.1. Training Accuracy and Training Loss

Three different datasets of Amazon product reviews were used to test and train the proposed LSTM and knowledge graph transE. 34000 reviews have been submitted for the CRAP dataset, 67986 for ACPR, and 568454 for AFFR. According to the table and fig:- dataset CRAP's initial accuracy was 0.93; it increased to 1.00 after 10 epochs and stayed there for 100 epochs. For the CRAP review dataset, the initial training loss was 0.2605. Training loss starts to increase after 9 epochs and then starts to reduce again after 100 epochs. For the ACPR dataset, the initial accuracy was 0.95. Up to 69 epochs, the accuracy was becoming better. It stayed at 1.00 after 70 epochs until 100 epochs. When accuracy hit 1.00, the training loss started to increase after declining for 69 epochs as shown in Figure.

	CRAP		ACPR		AFFR	
Epochs	**Training Loss**	**Training Accuracy**	**Training Loss**	**Training Accuracy**	**Training Loss**	**Training Accuracy**
Initial	0.2605	0.9349	0.1912	0.9553	0.1082	0.9578
3	0.0339	0.9912	0.036	0.9875	0.0368	0.9855
5	0.0139	0.9978	0.0241	0.9919	0.0220	0.9917
10	9.1497	1.000	0.0241	0.9978	0.0124	0.9955
50	1.9839	1.000	0.0068	0.9982	0.0039	0.9987
70	2.4089	1.000	4.5402	1.000	0.0026	0.9991
100	7.7528	1.000	3.2447	1.000	0.0014	0.9995

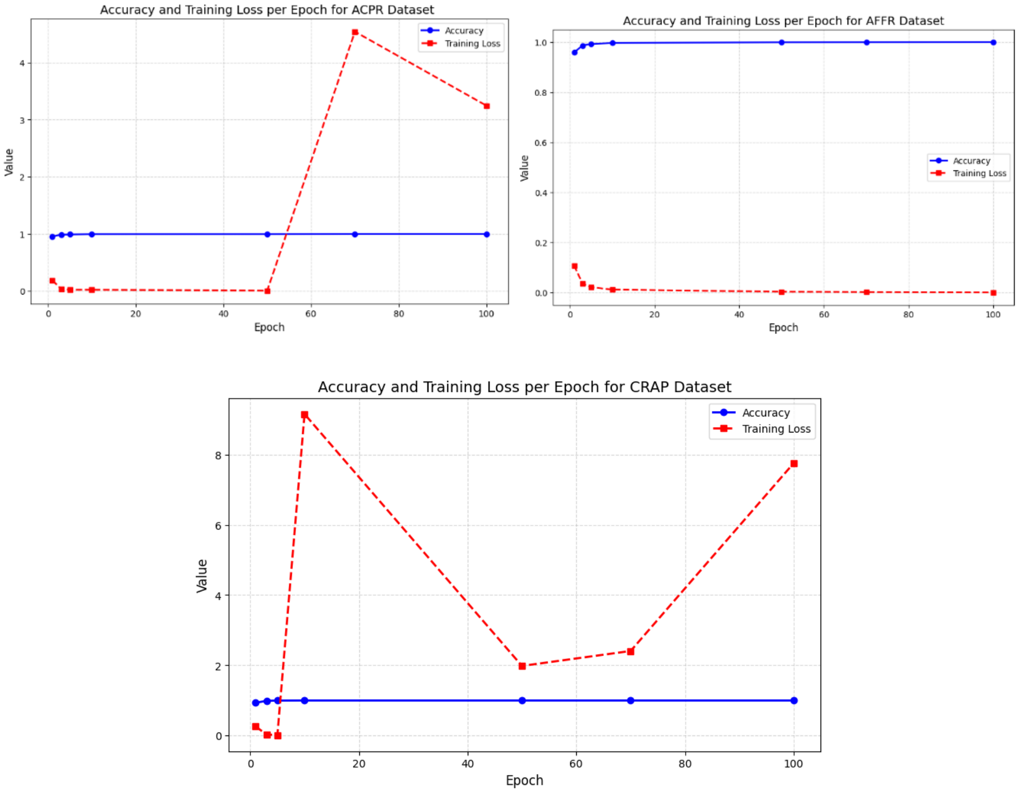

Figure 3. Various Relation Extracted from the Dataset.

5.2. Test Accuracy and Test Loss

As shown in figure, the test accuracy for the datasets CRAP, ACPR, and AFFR was 0.97, 0.98, and 0.96, respectively, while the test loss for each was 0.10, 0.079, and 0.17.

	Test Loss	Test Accuracy
CRAP	0.1010	0.9763
ACPR	0.079	0.9827
AFFR	0.1773	0.9690

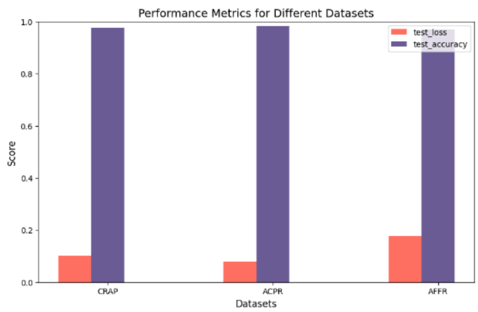

Figure 4. Various Relation Extracted from the Dataset.

5.3. Accuracy

The proposed LSTM+KG embedding model's average accuracy after training and testing on the three datasets of Amazon product reviews (CRAP, ACPR, AFFR) was 97%, 98%, and 96%, respectively.

Accuracy		
Dataset	**Model**	**Accuracy**
CRAP	TransE+LSTM	**97%**
ACPR	TransE+LSTM	98%
AFFR	TransE+LSTM	96%

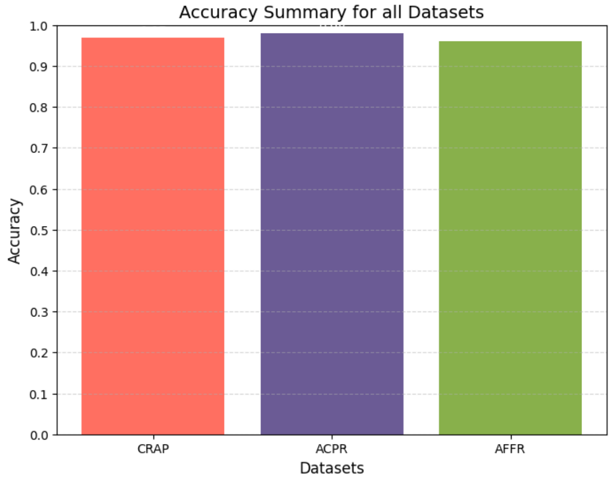

Figure 5. Relation Frequency Count.

5.4. Precision, Recall and F1 scores

On three benchmark datasets, CRAP, ACPR, and AFFR, which are all available for public use, the performance of the suggested LSTM+KGemb model was assessed. For CRAP, ACPR, and AFFR, the precision was 96%, 97%, and 97%, respectively, while the recall was 97%, 98%, and 97%. The F1 score was likewise 95%, 97%, and 96%, respectively.

Precision, Recall, F1-score Summary				
		CRAP	**ACPR**	**AFFR**
Precision	TransE+LSTM	96%	97%	97%
Recall	TransE+LSTM	97%	98%	97%
F1-score	TransE+LSTM	95%	97%	96%

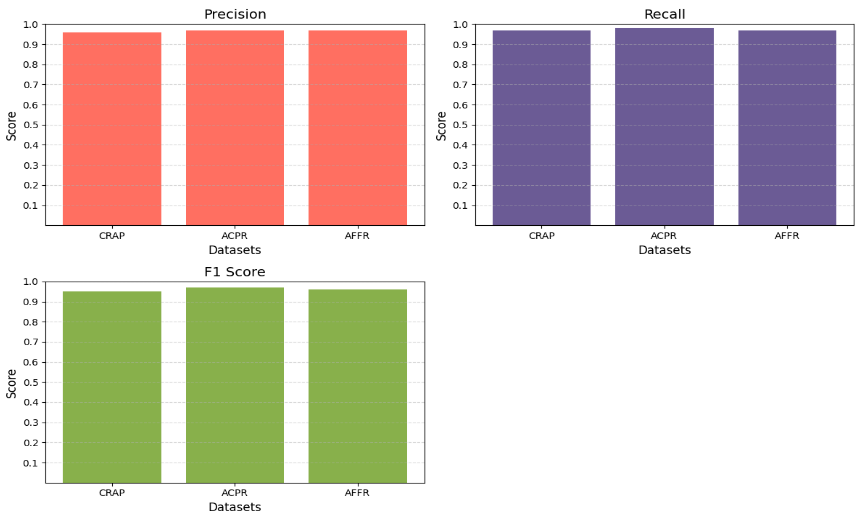

Figure 6. Relation Frequency Count.

6. Conclusion

Product reviews are a huge asset when making decisions about online purchases. Product reviews can also aid in product improvement from the standpoint of retailers and producers. It takes a lot of effort and time to read through each review because there are so many comparable products and reviews. The popularity of online buying has skyrocketed, particularly during and after the COVID-19 era. 2.44 billion individuals visited Amazon in September 2020, and 79% of those visitors read product evaluations before making a purchase, according to statistics from Statista (A Portal for Market Data). n Sentiment analysis is crucial because consumers want to know the [1] benefits and drawbacks of a product before making a purchase. Numerous items with nearly identical specifications exist, which could confuse consumers while making purchases. Manufacturers and customers benefit from sentiment analysis. Social media customer feedback helps businesses design products that meet customer needs, and information from social media about the products available for online shopping has grown to be an important resource for helping customers make better purchasing decisions and close big deals. There are several methods [3] for resolving this issue, but not enough to offer more precision. Because of this, many AI approaches using machine learning and deep learning techniques have been tested on various benchmark datasets to investigate the thoughts that customers express when using web-based entrances.

The proposed model for sentiment analysis classifies sentiment into three categories: positive, negative, and neutral, using LSTM for feature extraction and knowledge graph embedding for knowledge graphs. We used three publicly accessible benchmark datasets containing 34,000, 67,986 and 568,454 reviews each: "Customer reviews of Amazon Products," "Amazon Cell Phones Reviews," and "Amazon Fine Food Reviews." Text normalization was done as part of preprocessing tokenization to get rid of noise. Knowledge graphs were constructed after preprocessing, enabling knowledge graph embedding for feature extraction. The suggested model employed LSTM to classify sentiment. Instead of just categorizing reviews as positive or negative, we apply fuzzy logic to divide them into four categories: extremely positive, very negative, neutral, and neutral. The constructed model was tested on three separate datasets that were made accessible to the public, and it provides accuracy ratings of 97%, 98%, and 96%. Future aspect-based sentiment analysis experiments on Twitter datasets will use the established model.

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

References

1. M Sivakumar and Srinivasulu Reddy Uyyala. “Aspect-based sentiment analysis of mobile phone reviews using LSTM and fuzzy logic”. In: International Journal of Data Science and Analytics 12.4 (2021), pp. 355–367.
2. Chinmay Milind Suryavanshi. “Sentiment Analysis of Product Reviews: Opinion Extraction and Display”. PhD thesis. California State University, Northridge, 2021.
3. Ranjan Kumar Behera et al. “Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data”. In: Information Processing & Management 58.1 (2021), p. 102435.
4. Doaa Mohey El-Din Mohamed Hussein. “Analyzing scientific papers based on sentiment analy- sis”. In: Information System Department Faculty of Computers and Information Cairo University, Egypt (2016).
5. Fernando Andres Lovera, Yudith Coromoto Cardinale, and Masun Nabhan Homsi. “Sentiment Analysis in Twitter Based on Knowledge Graph and Deep Learning Classification”. In: Electronics 10.22 (2021), p. 2739.
6. Arwa SM AlQahtani. “Product Sentiment Analysis for Amazon Reviews”. In: International Journal of Computer Science & Information Technology (IJCSIT) Vol 13 (2021).
7. Emma Haddi. “Sentiment analysis: text, pre-processing, reader views and cross domains”. PhD thesis. Brunel University London, 2015.
8. Ansari Fatima Anees et al. “Survey paper on sentiment analysis: Techniques and challenges”. In: EasyChair2516-2314 (2020).
9. Muhammad Umer et al. “Sentiment analysis of tweets using a unified convolutional neural network- long short-term memory network model”. In: Computational Intelligence 37.1 (2021), pp. 409–434.
10. Mohammad Tubishat, Norisma Idris, and Mohammad AM Abushariah. “Implicit aspect extrac- tion in sentiment analysis: Review, taxonomy, oppportunities, and open challenges”. In: Informa- tion Processing & Management 54.4 (2018), pp. 545–563.
11. Ashima Yadav and Dinesh Kumar Vishwakarma. “Sentiment analysis using deep learning archi- tectures: a review”. In: Artificial Intelligence Review 53.6 (2020), pp. 4335–4385.
12. Momina Shaheen et al. “Sentiment analysis on mobile phone reviews using supervised learning techniques.” In: International Journal of Modern Education & Computer Science 11.7 (2019).
13. Bobby Nguy. “Evaluate helpfulness in amazon reviews using deep learning”. In: Stanford Univer- sity. 2016.
14. Madhuri Dadhich, Mausumi Goswami, and BS Purkayastha. “GLOBAL JOURNAL OF ENGI- NEERING SCIENCE AND RESEARCHES”. In: ().
15. Wei Zhao et al. “Weakly-supervised deep embedding for product review sentiment analysis”. In: IEEE Transactions on Knowledge and Data Engineering 30.1 (2017), pp. 185–197.
16. James Barry. “Sentiment Analysis of Online Reviews Using Bag-of-Words and LSTM Approaches.” In: AICS. 2017, pp. 272–274.
17. Michele Di Capua, Emanuel Di Nardo, and Alfredo Petrosino. “An architecture for sentiment analysis in twitter”. In: Proceed. of Int. Conference on E-learning, Germany. 2015, pp. 1–10.
18. Achira Jeewaka Shamal et al. “Sentiment analysis using Token2Vec and LSTMs: user review an- alyzing module”. In: 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer). IEEE. 2018, pp. 48–53.
19. Abhishek Kumar, Daisuke Kawahara, and Sadao Kurohashi. “Knowledge-enriched two-layered attention network for sentiment analysis”. In: arXiv preprint arXiv:1805.07819 (2018).
20. Hanxue Ji et al. “LSTM based semi-supervised attention framework for sentiment analysis”. In: 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scal- able Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE. 2019, pp. 1170–1177.
21. Ali Feizollah et al. “Halal products on Twitter: Data extraction and sentiment analysis using stack of deep learning algorithms”. In: IEEE Access 7 (2019), pp. 83354–83362.
22. Wanliang Tan, Xinyu Wang, and Xinyu Xu. “Sentiment analysis for Amazon reviews”. In: Interna- tional Conference. 2018, pp. 1–5.
23. Mehmet Umut Salur and Ilhan Aydin. “A novel hybrid deep learning model for sentiment classi- fication”. In: IEEE Access 8 (2020), pp. 58080–58093.
24. R Kiran, Pradeep Kumar, and Bharat Bhasker. “OSLCFit (organic simultaneous LSTM and CNN Fit): a novel deep learning based solution for sentiment polarity classification of reviews”. In: Expert Systems with Applications 157 (2020), p. 113488.
25. Amol C Adamuthe. “Improved Text Classification using Long Short-Term Memory and Word Embedding Technique”. In: International Journal of Hybrid Information Technology 13.1 (2020), pp. 19– 32.
26. Ganesh More, Harish Behara, and Apoorva Mysore Suresha. “Sentiment Analysis on Amazon Product Reviews with Stacked Neural Networks”. In: no. October (2020).
27. Ishaani Priyadarshini and Chase Cotton. “A novel LSTM–CNN–grid search-based deep neural network for sentiment analysis”. In: The Journal of Supercomputing 77.12 (2021), pp. 13911–13932.
28. Zubair Nawaz et al. “Role of Artificial Neural Networks Techniques in Development of Market Intelligence: A Study of Sentiment Analysis of eWOM of a Women’s Clothing Company”. In: Journal of Theoretical and Applied Electronic Commerce Research 16.5 (2021), pp. 1862–1876.

29. Sakirin Tam, Rachid Ben Said, and Ö Özgür Tanriöver. "A ConvBiLSTM deep learning model- based approach for Twitter sentiment classification". In: IEEE Access 9 (2021), pp. 41283–41293.
30. Hafed Zarzour, Mahmoud Al-Ayyoub, Yaser Jararweh, et al. "Sentiment Analysis Based on Deep Learning Methods for Explainable Recommendations with Reviews". In: 2021 12th International Conference on Information and Communication Systems (ICICS). IEEE. 2021, pp. 452–456.
31. N Aishwarya, LS Bhuvana, and N Kayarvizhy. "Summarization and Prioritization of Amazon Re- views based on multi-level credibility attributes". In: 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT). IEEE. 2021, pp. 5–9.
32. Xuechun Li et al. "Explainable Sentence-Level Sentiment Analysis for Amazon Product Reviews". In: 2021 5th International Conference on Imaging, Signal Processing and Communications (ICISPC). IEEE. 2021, pp. 88–94.
33. Yan Li and Nianfeng Li. "Sentiment Analysis of Weibo Comments Based on Graph Neural Net- work". In: IEEE Access 10 (2022), pp. 23497–23510.
34. Zuhua Dai et al. "Aspect-level sentiment analysis merged with knowledge graph and graph con- volutional neural network". In: Journal of Physics: Conference Series. Vol. 2083. 4. IOP Publishing. 2021, p. 042044.
35. Julio Vizcarra et al. "Knowledge-based sentiment analysis and visualization on social networks". In: New Generation Computing 39.1 (2021), pp. 199–229.
36. Wen Chen et al. "Research on sentiment classification of online travel review text". In: Applied Sciences 10.15 (2020), p. 5275.
37. Jie Zhou et al. "Sk-gcn: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification". In: Knowledge-Based Systems 205 (2020), p. 106292.
38. Jinlong Ji et al. "Cross-domain sentiment classification via a bifurcated-LSTM". In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer. 2018, pp. 681–693.
39. Xiaoyan Yan, Fanghong Jian, and Bo Sun. "SAKG-BERT: Enabling Language Representation With Knowledge Graphs for Chinese Sentiment Analysis". In: IEEE Access 9 (2021), pp. 101695–101701.
40. Manuela Nayantara Jeyaraj. "Conceptualizing the Knowledge Graph Construction Pipeline". In: Towards Data Science (2019).
41. Niannian Guan, Dandan Song, and Lejian Liao. "Knowledge graph embedding with concepts". In: Knowledge-Based Systems 164 (2019), pp. 38–44.
42. I. Ihsan, H. Rahman, A. Sheikh, A. Sulaiman, K. Rajab and A. Rajab, "Improving in-text citation reason extraction and classification using supervised machine learning techniques," in Computer Speech & Language, 82, 101526 (2023). https://doi.org/10.1016/j.csl.2023.10152.
43. I. Ihsan and M. A. Qadir, An NLP-based citation reason analysis using CCRO. Scientometrics 126, 4769–4791 (2021). https://doi.org/10.1007/s11192-021-03955-6.
44. Li, J., Hu, Q., Imran, A., Zhang, L., Yang, J. J., & Wang, Q. (2018, July). Vessel recognition of retinal fundus images based on fully convolutional network. In 2018 IEEE 42nd annual computer software and applications conference (COMPSAC) (Vol. 2, pp. 413-418). IEEE.
45. Akhtar, F., Li, J., Guan, Y., Imran, A., & Azeem, M. (2019). Monitoring bio-chemical indicators using machine learning techniques for an effective large for gestational age prediction model with reduced computational overhead. In Frontier Computing: Theory, Technologies and Applications (FC 2018) 7 (pp. 130-137). Springer Singapore.
46. Wajahat, A., Imran, A., Latif, J., Nazir, A., & Bilal, A. (2019, January). A Novel Approach of Unprivileged Keylogger Detection. In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-6). IEEE.
47. Bilal, A., Sun, G., Mazhar, S., Imran, A., & Latif, J. (2022). A Transfer Learning and U-Net-based automatic detection of diabetic retinopathy from fundus images. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 10(6), 663-674.
48. Imran, A., Li, J., Pei, Y., Mokbal, F. M., Yang, J. J., & Wang, Q. (2020). Enhanced intelligence using collective data augmentation for CNN based cataract detection. In Frontier Computing: Theory, Technologies and Applications (FC 2019) 8 (pp. 148-160). Springer Singapore.
49. Ali, S., Rehman, S. U., Imran, A., Adeem, G., Iqbal, Z., & Kim, K. I. (2022). Comparative Evaluation of AI-Based Techniques for Zero-Day Attacks Detection. Electronics, 11(23), 3934.
50. Ashfaq, A., Imran, A., Ullah, I., Alzahrani, A., Alheeti, K. M. A., & Yasin, A. (2022, October). Multi-model Ensemble Based Approach for Heart Disease Diagnosis. In 2022 International Conference on Recent Advances in Electrical Engineering & Computer Sciences (RAEE & CS) (pp. 1-8). IEEE.