International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Document Clustering using RoBERTa and Convolution Neural Network Model

P. Saidesh Kumar[*1], Dr. P. Vijayapal Reddy[2]

Submitted: 10/10/2023 **Revised**: 29/11/2023 **Accepted**: 10/12/2023

Abstract: Document clustering quite helpful in many applications of text mining and information retrieval. The use of cluster analysis to text texts is known as document clustering. It may be used to swiftly retrieve or filter information as well as automatically arrange papers into categories and extract themes from texts. In this study, a document clustering technique based on deep learning and lexical text feature extraction is presented. RoBERTa (Robustly Optimized BERT Pre-training Approach), a recommendation framework to extract text features where BERT (Bidirectional Encoder Representation from Transformers) receives significant hyper parameter alterations from RoBERTa. The BERT pre-next-sentence training objective is no longer used, and training in tiny batches results in significantly higher learning rates. The features are sent to CNN (Convolution Neural Networks) model containing dense and drop out layers. The proposed model obtained an accuracy of 98.3%for BBC dataset and 98.2% for News group dataset.

Keywords: *BERT, Convolution Neural Networks (CNN), Document Clustering, RoBERTa.*

1. Introduction

Clustering is one of the most significant approaches that lies at the foundation of machine learning, and it is capable of being applied to many datasets, including the document set [1-2]. Clustering is a technology for classifying data by computer. It is also a classification method for multivariate statistical analysis [3-4]. Clustering divides a data set according to a specific standard into various classes or clusters [5], and it prioritizes increasing the similarity between data objects over increasing the differences between data objects within the same cluster. The term "text clustering" refers to the organization of documents into groups. In a similar manner, data that have comparable qualities are grouped together as much as is practically feasible, whilst data that differ in significant ways are kept as far apart as is practically possible. Clustering is possible not just with text but also with anything else whose characteristics may be retrieved [6]. E-commerce websites, for instance, group products according to characteristics like price and color; app shops group apps according to the age of their users and the number of downloads; and movie websites group movies according to the topic of the films and the year they were released. Through the process of feature extraction, which takes place in the domain of mathematics, machine learning, which includes clustering, may be carried out by just transforming objects from the actual world into vectors.

One of the most effective techniques for organising, condensing, and reading textual information is text clustering, a traditional data mining technique [7]. Finding the content of a text document resource is done via text clustering. This is accomplished by classifying text texts according to several criteria of similarity. The goal is to make sure that each document type's similarity can meet a certain criteria, offering a description of the information unique to each kind. Documents with a high degree of resemblances are part of same categories and low similarities between documents that are part of other categories are one of the factors that text clustering algorithms use to choose grouping of text. There are many different types of text clustering algorithms available today. These text clustering algorithms are primarily divided up into the following four groups, according to the categorization principles that are used for clustering algorithms.

The act of assigning categories or classes to documents in order to make it simpler to manage, search, filter, or analyze the documents is referred to as document clustering [8]. In this context, the term "document" refers to a piece of information that contains information pertinent to a certain category. Emails, product images, commentary, bills, and scanned papers may all be regarded to be different types of documents. It's possible that document clustering is only one component of a much larger project known as intelligent document processing (IDP).

Natural language processing, often known as NLP, is required when working with increasingly complicated text categorization problems. NLP is a field that draws from multiple different areas of study, including linguistics, statistics, and computer science approaches that give computers the ability to comprehend human language in its

[1]*Research Scholar, University College of Engineering, Osmania University, Hyderabad, India.*
[2]*Prof., HOD CSE, Matrusri Engineering College, Hyderabad, India.*
Email: drpvijayapalreddy@gmail.com
** Corresponding Author Email: saideshp@gmail.com*

natural setting. Document classifiers can identify patterns in texts or even understand what words mean with the assistance of NLP. NLP is a machine learning technique, which means that it requires a large amount of data in order to train a model. However, it is useful for solving more complex text clustering problems, such as evaluating comments, articles, reviews, and other types of media materials.

2. Literature

Janani and Vijayarani [9] To enhance text document clustering, a brand-new methodology dubbed SCPSO (Spectral Clustering with Particle Group Optimisation) has been developed. Using both local and global optimisation functions, the randomization is carried out from the original population. This research axis aims to study the feasibility of integrating spectral clustering with swarm optimisation to handle a vast volume of text content.

Curriskis et al. [10] used several topic modelling and document clustering techniques to analyse three different datasets that were obtained from Twitter and Reddit. The performance of four different feature representations obtained from the word embedding model and inverse document frequency matrix (tf-idf) together with four clustering techniques is assessed by the authors. They also give a conceptual model of the latent Dirichlet distribution for comparison's sake. They present a discussion and suggestion for the extrinsic measures that are most suited for this activity as a result of the fact that the literature has a number of distinct assessment measures that are employed.

Dogan and Birant [11] offered an overview of how machine learning methods might be employed to realise manufacturing mechanisms with intelligent behaviours. A complete literature review has been provided to do so. In addition to this, it draws attention to a number of key research issues that have been raised in recent literature with the same aim but have not yet been addressed. Through this study, the authors want to provide scholars a good grasp of the primary methodologies and algorithms that have been employed to enhance manufacturing processes during the last twenty years. It does this by organising the earlier ML research and more contemporary manufacturing advancements into four primary categories: scheduling, monitoring, quality, and failure.

Fard et al. [12] provided k-means clustering dependent on continuous reparametrization of objective functions resulting injoint solutions. The method's behaviour is exhibited using a variety of datasets, demonstrating its effectiveness in learning representations for objects while grouping those items together.

Huang et al. [13] examined hierarchical semantics of each input data layer, a novel deep multi-view clustering model was presented. A novel collaborative deep matrix decay architecture is used to learn hidden representations in a range of features. The hierarchical semantics that are gained by each layer may be learned in a cooperative manner by the model that has been suggested. In the low-dimensional space, the instances from the same class are driven to be closer to one another when one layer is added on top of the previous one, which is useful for the future clustering operation. In addition, an idea weight is mechanically allocated to each view, as opposed to the practise of the earlier techniques, which included the introduction of an additional hyperparameter. An effective technique for iteratively updating the model has been provided, and its convergence has also been theoretically guaranteed. This will allow us to tackle the optimization challenge that the model presents.

Ren et al. [14] offered a novel approach to semi-supervised deep embedded clustering (SDEC), in order to circumvent this restriction, to be more specific, SDEC is capable of learning feature representations that are favourable to the clustering tasks and concurrently performing clustering assignments. In contrast to DEC, SDEC includes pairwise constraints as part of the process of learning new features. These constraints ensure that data samples that belong to the same cluster are located relatively close to one another in the learned feature space, while data samples that belong to different clusters are located relatively far apart from one another.

Elnagar et al. [15] introducing entirely new datasets for single-label (SANAD) and multiple-label (NADiA) Arabic text categorization tasks that are insightful and impartial. The two stated repositories are open to the scientific community engaged in Arabic computational linguistics. Furthermore, the authors offer a thorough comparison of various deep learning (DL) models for Arabic text categorization in order to assess the performance of such models on SANAD and NADiA. What distinguishes the proposed effort from other existing initiatives of a similar kind is the absence of any preprocessing procedures and the exclusive reliance on deep learning models.

Kumar et al. [16] introduced a four-part contextual model called ConvNet-SVMBoVW . These include a decision module, a discrete module, a text analysis module, and a module for analysing images. Multimodal text, represented by the symbol m (which stands for "text, image, infographic"), is the input that the model accepts. To distinguish text from photos, the discrete module makes advantage of Google Lens. The modules in charge of text analysis and image analysis, respectively, receive the text and images after they have been processed as separate entities. The text analysis module uses SentiCircle's contextual semantics in conjunction with a convolutional

neural network (ConvNet) modified to identify the mood of the input text. Here, a synthetic approach for calculating hybrid polarisation is described.

Muller et al. [17] evaluated how beneficial an automated clustering approach called Lingo3G is for classifying studies in a simplified quick review, and then to compare the performance of this method to the performance of human classification in terms of accuracy and recall. The authors used a random assignment process to decide whether each of the 128 papers in the review would be coded by a human researcher who was blind to the cluster assignment or by a human researcher who was not blind to the cluster assignment.

Khan et al. [18] placed a greater emphasis on Non-Negative Matrix FactorizationNMF clustering approach for multi-view data that incorporates manifold regularization. It is possible to maintain the locally geometrical structure of the data space by using the manifold regularization factor, which also provides an extensively common clustering solution when viewed from many perspectives. The phrase "weight control" has been coined in order to manage the distribution of the weight of each view.

3. Proposed Framework

The vocabulary model RoBERTa is used for the purpose of extracting text features by the proposed framework is shown in fig. 1. Adjustments of vital hyperparameters are made by RoBERTa to the representations of BERT. These benefits include removing the next-sentence pretraining objective from BERT and increasing the size of both the mini-batches and the learning rates of the training sessions. The features are then passed to a model of a Convolutional Neural Network (CNN) that includes dense and drop out layers.

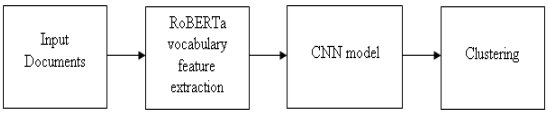

Fig. 1. Proposed Model

3.1. RoBERTa

RoBERTa was developed by researchers from Facebook and Washington University. The purpose of creating this model is to optimise the BERT architecture's training process in order to shorten the pre-training phase. RoBERTa's architecture is quite similar to BERT's.; however, in order to enhance the results obtained by BERT architecture, the authors made some simple modifications to the architecture of RoBERTa as well as to the training technique. The functions of RoBERTa include:

- **Getting rid of the Next Sentence Prediction (NSP) objective**: The next sentence prediction feature (NSP) is used in next sentence prediction to train the model to determine whether the observed document segments are from the same document or from other documents. Auxiliary. To accomplish the next sentence prediction (NSP) objective, this is done. The authors found that deleting NSP loss is equivalent to or slightly increases the performance of the writers after conducting a series of experiments in which they added or removed NSP loss to various versions. upstream support.
- **Training with increased batch sizes and longer sequences**: Using batches of 256 sequences each, BERT was initially trained for 1 million steps. In this work, the researchers used 31,000 steps with a batch size of 8,000 sequences and 125 steps with 2,000 sequences to train the model. This has two benefits: one, it raises the end task's accuracy, and second, it makes the goal of obfuscated language modelling more challenging. Additionally, distributed parallel training makes it simpler to parallelize large batches..
- **Altering the masking pattern dynamically**: In this concept, a single static mask is produced by doing the masking only once, during the data preparation stage as the BERT design, makes it feasible. Using a different masking method over more than forty epochs, the training data was duplicated and masked 10 times. Only four epochs use the same mask as a result of this, which prevents the use of a single static mask. This method is thought to be more adaptable than dynamic masking, which creates a new mask for each new dataset added to the model.

The RoBERTa model is trained on the following datasets:

- BOOK CORPUS and English Wikipedia dataset: 16GB of text.
- CC-NEWS. 63 million English news articles, 76 GB of text.
- OPENWEBTEXT: 38 GB of web text documents.
- STORIES: 31 GB of text.

3.2. Working of RoBERTa

The fundamental BERT model consists of 12 layers with a total of 110 million parameters, 768 hidden and equal integration layers, and 768 layers. The creation process necessitates extensive computing because of its enormous scale. Since Bert's unique words are encoded, prefixing such words with special letters is not necessary during sentence tokenization. If, on the other hand, it is a compound word, the word is broken up into many chunks of subwords, all of which, with the exception of the first subword, have the prefix "##" added to them. The notation ## in the Bert tokenizer indicates that the current subword should be joined with the one that came before it in order to form a single word.

During the tokenization process in Roberta, the initial word of the phrase does not have any kind of prefix or special character added to it. However, the prefix "" is added to all of the words in the sentence with the exception of the first word. It is important to take note that if a single word is broken up into multiple subwords, the initial subword is given the "" prefix, while the remaining subwords do not receive any additional prefixes or other special characters. In the Roberta tokenizer, a subword that does not have the "" character after it is an indication that the current subword should be joined with the one that came before it to form a single word.

3.3. CNN

A deep learning model called CNN is used to handle data with a grid pattern, such photographs [19-25]. This model was developed to automatically and adaptively learn spatial hierarchies of features, progressing from low-level models to high-level models. It was inspired by the structure of the visual brain of animals. higher order image. The proposed CNN model is shown in fig. 2.

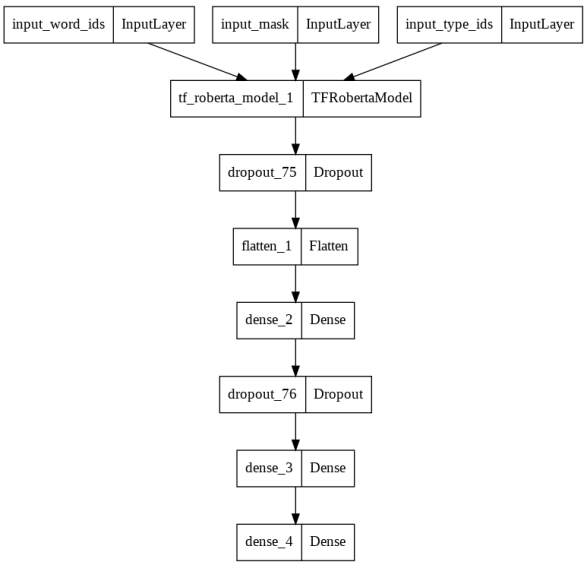

Fig. 2. Proposed CNN Model

The RoBERTa layers are added first added to the model that extracts features from the text like a convolution layer. These features are then sent to the subsequent layers.

- **Dense Layer**: Since each neuron in this layer receives information from every neuron in the layer below it, the dense layer is a layer of neurons. The dense layer's classification process is supported by the output of the convolutional layers. a single neuron's activity These neurons can be found in large numbers within a layer. The dot product process, which the dense layer is in charge of, requires more computing than the selection operation, which the embedding layer is in charge of. As a result, training proceeds significantly more quickly.
- **Flatten**: A flattening layer is used to reduce the input's spatial dimension to fit the channel size. The output of the layer after flattening is an array of size (H*W* C) x N x S, for instance, if the input of the layer is an array of size H x W x C x N x S (image sequence).
- **Dropout Layer**: The dropout method The dropout approach is a training tactic in which a predefined number of neurons are arbitrarily skipped. They are "dropped" in a random order. This means that during the forward transition, their contribution to the activation of downstream neurons will be temporarily erased, and during the reverse transition, any weight changes won't be applied to the neuron. Additionally, this indicates that no weight adjustments will be applied while forwarding.

Loss Function

Sparse categorical crossentropies: For multi-class classification models where the output label is provided an integer value (for example, 0, 1, 2, 3), this loss function is used. In terms of its mathematical form, this loss function is equivalent to the categorical cross entropy.

Algorithm: Proposed Mode (RoBERTa with CNN)

Input: BBC or Newgroup (dataset)

Output: Clustered documents, Clustering time, Accuracy, Parameters (Precision, recall, F1-score)

Step 1: IMPORT PACKAGES

In the first step, all of the python packages are imported.

Step 2: SET UP THE CONFIGURATION

The vast majority of modifications need to be made in this section of 'config.' In specifically, these are the model's hyperparameters, as well as the path to the files and the names of the columns.

Step 3: Read Input dataset with n documents.

Step 4: Determine categorizes and read all documents from the dataset

Step 5: SET UP TOKENIZER

- Construct fastai tokenizer for roberta
- Create fastai vocabulary for roberta
- Setting up pre-processors
- Create fastai tokenizer for roberta

Step 6: SET UP DATABUNCH

- Creating a DataBunch class that is special to Roberta

- loading the dataset
- loading the tokenizer and vocab processors
- removing files that aren't essential

Step 7: TRAINING AND EVALUATION

- designing the architecture of the model
- train the data with the model

Step 8: PREDICTION

- Once the model is trained, new data can be tested.

Step 9: CNN

- The RoBERTa layers are added to the model that extracts features from the text like a convolution layer
- The dropout approach, a training tactic that includes eliminating a predefined number of random neurons, receives features.
- Using a flattening layer, the input's spatial dimension is condensed into the channel size.
- The output of the convolutional layers is used as the foundation for the classification carried out by the dense layer, which is a layer of neurons, and the feature is transmitted to this layer, which is composed of all the neurons in the layer underneath it.

4. Experimental Results

This section presents the experimental analysis carried out to validate the proposed model.

Dataset 1: BBC News Dataset

This work's 2,225 documents that make up the BBC News dataset were taken from the BBC News website and relate to content that was published in five distinct topic areas between 2004 and 2005. There are five class tags. Table 1 displays the document label and quantity.

Table 1. Dataset-1 details

Labels	***Document count***
sports	511
businesses	510
politics	417
technical	401
entertainments	386

The distribution of texts is shown in fig. 3.

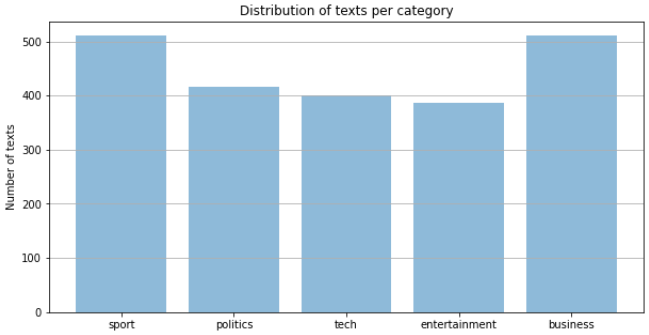

Fig. 3. Distribution of Texts Per Category (Dataset-1)

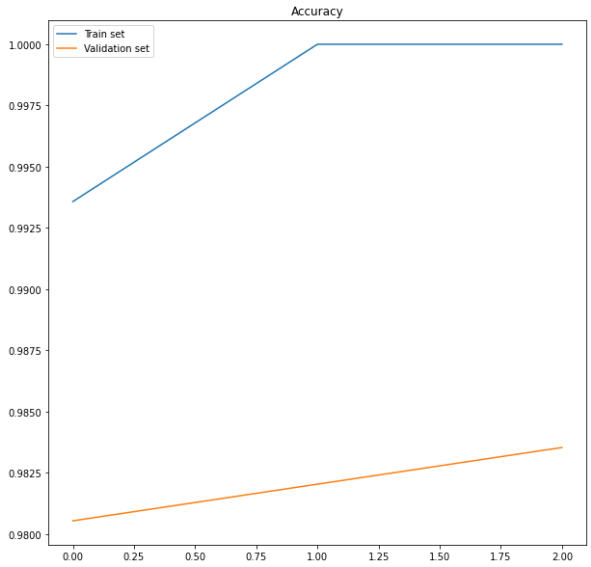

Fig. 4. Training and Validation Accuracy (Dataset-1)

Training and Validation Accuracy (Dataset-1) is shown in fig. 4.

The table 2 shows the results of dataset-1. The performance is evaluated using different parameters namely Precision, recall, f1-score and support.

Table 2. Validation parameters (Dataset-1)

	Precision	***Recall***	***f1-score***
Business	0.99	1.00	1.00
Politics	0.97	0.97	0.97
Entertainment	0.97	0.99	0.98
Technology	1.00	1.00	1.00
Sport	0.98	0.96	0.97

The Precision for different categories business, politics, entertainment, technology and sport is 0.99, 0.97, 0.97, 1.00 and 0.98 respectively. The Recall for different categories business, politics, entertainment, technology and sport is 1.00, 0.97, 0.99, 1.00 and 0.96 respectively. The f1-score for different categories business, politics, entertainment, technology and sport is 1.00, 0.97, 0.98, 1.00 and 0.97 respectively. The Support for different

categories business, politics, entertainment, technology and sport is 155, 124, 107, 123 and 159 respectively.

Table 3. Accuracy and averages (Dataset-1)

	Accuracy	***Macro Avg***	***Weighted Avg***
Dataset-1	0.98	0.98	0.98

The Accuracy, Macro Avg and Weighted Avg for dataset-1 are 0.98, 0.98 and 0.98 respectively is shown in table 3.

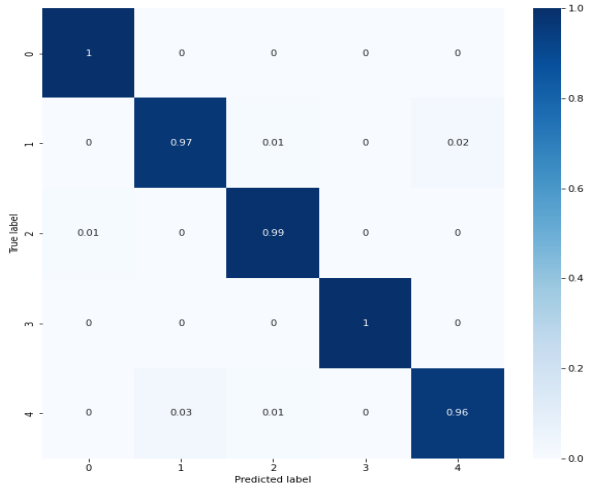

Fig. 5. Confusion Matrix (Dataset-1)

Time taken for proposed method to run on dataset-1 is 72ms is shown in table 4.

Table 4. Execution time

	Execution Time (milli seconds)
GloVe with LSTM [19, 20, 21]	76ms
Spacy with CNN [22]	87ms
NLTK with CNN [23]	91ms
GloVe with CNN [24]	143ms
BERT with CNN [25]	84ms
Proposed model (RoBERTa with CNN)	72ms

The time taken to run for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 76ms, 48ms, 11ms, 143ms, 84ms and 72ms respectively.

This table 5 shows the comparative analysis of GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN andBERT with CNN. The performance is evaluated using different parameters namely Accuracy, Precision, Recall and f1-score.

Table 5. Comparative analysis

Algorithm Name	***Accuracy (%)***	***Precision***	***Recall***	***f1-score***
GloVe with LSTM [19, 20, 21]	67.64%	0.732	0.714	0.660
Spacy with CNN [22]	81.63%	0.812	0.804	0.804
NLTK with CNN [23]	79.18%	0.804	0.772	0.768
GloVe with CNN [24]	85.39%	0.872	0.842	0.848
BERT with CNN [25]	92.13%	0.930	0.922	0.925
Proposed model(RoBERTa with CNN)	98.30%	0.982	0.984	0.984

The Accuracy for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 67.64%, 81.63%, 79.18%, 85.39%, 92.13% and 98.30% respectively.

The Precision for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.732, 0.812, 0.804, 0.872, 0.930, 0.82 respectively.

The Recall for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.714, 0.804, 0.772, 0.842, 0.922, 0.984 respectively.

The F1-score for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.660, 0.804, 0.768, 0.848, 0.925, 0.984 respectively.

Dataset 2: News Groups

This dataset is a compilation of documents from several newsgroups. The collection of 10 newsgroups has been a common choice for use as a data set for research involving the application of machine learning algorithms to text, namely text clustering and text classification [26-29].

Table 6. Dataset-2 details

Label	***Document count***
Food	100
Graphics	100
Medical	100

Space	100
Historical	100
Sport	100
Entertainment	100
Politics	100
Business	100
Technology	100

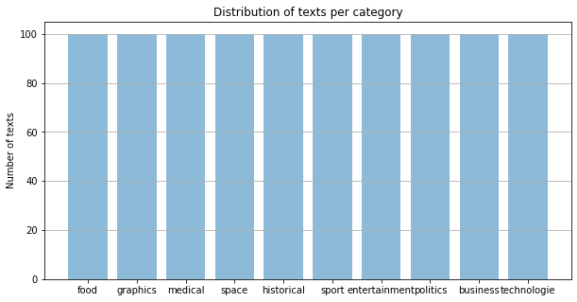

Fig. 6. Distribution of Texts Per Category (Dataset-2)

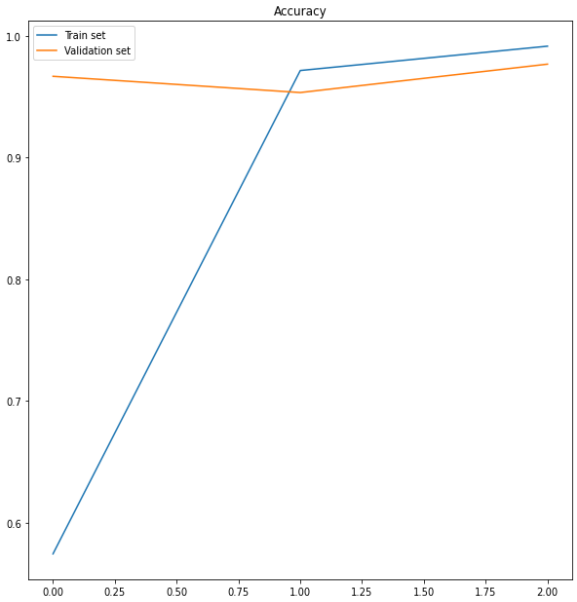

Fig. 7. Training and Validation Accuracy (Dataset-2)

The table 7 shows the results of dataset 2. The performance is evaluated using different parameters namely Precision,recall, f1-score and support.

Table 7. Validation parameters (Dataset-2)

	Precision	***Recall***	***f1-score***
Historical	1.00	1.00	1.00
Politics	1.00	0.96	0.98
Entertainment	0.97	1.00	0.99
Space	0.96	0.96	0.96
Sport	1.00	1.00	1.00
Graphics	1.00	1.00	1.00
Business	0.97	1.00	0.98
Technology	0.93	0.96	0.95
Food	0.93	0.93	0.93
Medical	1.00	0.93	0.96

The Precision for different categories Historical, Politics, Entertainment, Space, Sport, Graphics, Business, Technology, Food and Medical is 1.00, 1.00, 0.97, 0.96, 1.00, 1.00, 0.97, 0.93, 0.93 and 1.00 respectively. The Recall for different categories Historical, Politics, Entertainment, Space, Sport, Graphics, Business, Technology, Food and Medical is 1.00, 0.96, 1.00, 0.96, 1.00, 1.00, 1.00, 0.96, 0.93 and 0.93 respectively. The f1-score for different categories Historical, Politics, Entertainment, Space, Sport, Graphics, Business, Technology, Food and Medical is 1.00, 0.98, 0.99, 0.96, 1.00, 1.00, 0.98, 0.95, 0.93 and 0.96 respectively. The Support for different categories Historical, Politics, Entertainment, Space, Sport, Graphics, Business, Technology, Food and Medical is 37, 28, 33, 26, 33, 28, 32, 28, 27 and 28 respectively.

Table 8. Accuracy and averages (Dataset-2)

	Accuracy	***Macro Avg***	***Weighted Avg***
Dataset-2	0.98	0.98	0.98

The Accuracy, Macro Avg and Weighted Avg for dataset-1 are 0.98, 0.98 and 0.98 respectively is shown in table 8.

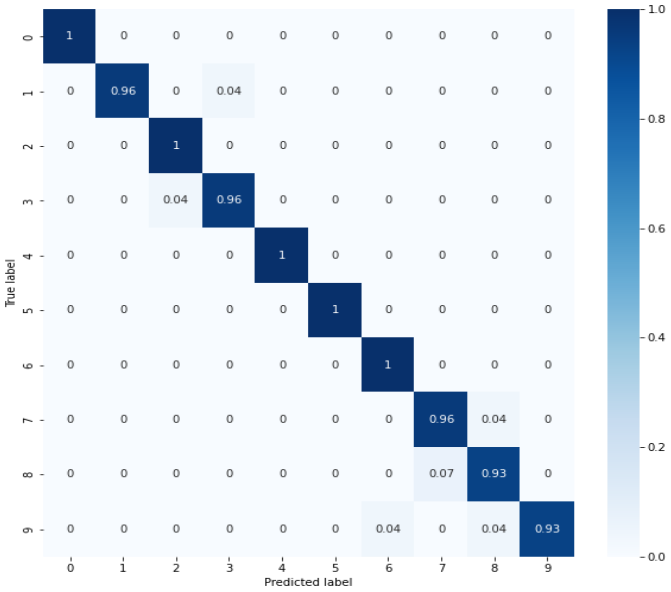

Fig. 8. Confusion Matrix (Dataset-2)

Table 9. Execution time

	Execution Time (milli seconds)
GloVe with LSTM [19, 20, 21]	83ms
Spacy with CNN [22]	89ms
NLTK with CNN [23]	95ms
GloVe with CNN [24]	121ms
BERT with CNN [25]	88ms
Proposed model (RoBERTa with CNN)	75ms

The time taken to run for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN,GloVe with CNN, BERT with CNN and RoBERTa with CNN is 83ms, 89ms, 95ms, 121ms, 88ms and 75ms respectively.

This table10 shows the comparative analysis of GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN and BERT with CNN. The performance is evaluated using different parameters namely Precision, Recall and f1-score.

Table 10. Comparative analysis

Algorithm Name	***Accuracy (%)***	***Precision***	***Recall***	***f1-score***
GloVe with LSTM [19, 20, 21]	61.50%	0.631	0.616	0.583
Spacy with CNN [22]	80.24%	0.802	0.794	0.814
NLTK with CNN [23]	79.18%	0.793	0.772	0.788
GloVe with CNN [24]	87.00%	0.874	0.882	0.872
BERT with CNN [25]	91.02%	0.920	0.900	0.912
Proposed model (RoBERTa with CNN)	98.2%	0.976	0.974	0.975

The Accuracy for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 61.50, 80.24%, 79.18%, 87%, 91.02% and 98.2% respectively.

The Precision for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.631, 0802, 0.793, 0.874, 0.920 and 0.976 respectively.

The Recall for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.616, 0.794, 0.772, 0.882, 0.900 and 0.974 respectively.

The F1-score for different methodologies GloVe with LSTM, Spacy with CNN, NLTK with CNN, GloVe with CNN, BERT with CNN and RoBERTa with CNN is 0.583, 0.814, 0.788, 0.872, 0.912 and 0.975 respectively.

5. Conclusion

The suggested architecture makes use of a RoBERTa-based lexical model for text feature extraction. RoBERTa (BERT) made significant hyperparametric changes to the bidirectional encoder representations from the processor. These advantages include the capacity to train with tiny batches and a significantly higher learning rate, as well as the removal of BERT's pre-training aim for the subsequent phrase. In addition to the rejected layers, the features fed into the CNN model also include dense layers. For the BBC dataset and the Newsgroup dataset, the suggested model's accuracy is 98.3% and 98.2%, respectively.

References

[1] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine learning*, vol. 109, no. 2, pp. 373-440, 2020.

[2] G. Baryannis, S. Dani and G. Antoniou, "Predicting supply chain risks using machine learning: The trade-off between performance and interpretability," *Future Generation Computer Systems*, vol. 101, pp. 993-1004, 2019.

[3] C. Maione, F. Barbosa Jr and R. M. Barbosa, "Predicting the botanical and geographical origin of honey with multivariate data analysis and machine learning techniques: A review," *Computers and Electronics in Agriculture*, vol. 157, pp. 436-446, 2019.

[4] A. Khraisat, I. Gondal, P. Vamplew and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1-22, 2019.

[5] R. Yan, J. Liao, J. Yang, W. Sun, M. Nong and F. Li, "Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering," *Expert Systems with Applications*, vol. 169, pp. 1-15, 2021.

[6] A. Al-Subaihin, F. Sarro, S. Black and L. Capra, "Empirical comparison of text-based mobile apps

similarity measurement techniques," *Empirical Software Engineering*, vol. 24, pp. 3290-3315, 2019.

[7] A. Onan and M. A. Toçoğlu, "Weighted word embeddings and clustering-based identification of question topics in MOOC discussion forum posts," *Computer Applications in Engineering Education*, vol. 29, no. 4, pp. 675-689, 2021.

[8] F. J. Arenas-Márquez, R. Martinez-Torres and S. Toral, "Convolutional neural encoding of online reviews for the identification of travel group type topics on TripAdvisor," *Information Processing & Management*, vol. 58, no. 5, pp. 1-16, 2021.

[9] R. Janani and S. Vijayarani, "Text document clustering using spectral clustering algorithm with particle swarm optimization," *Expert Systems with Applications*, vol. 134, pp. 192-200, 2019.

[10] S. A. Curiskis, B. Drake, T. R. Osborn and P. J. Kennedy, "An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit," *Information Processing & Management*, vol. 57, no. 2, pp. 1-50, 2020.

[11] A. Dogan and D. Birant, "Machine learning and data mining in manufacturing," *Expert Systems with Applications*, vol. 166, pp.1-44, 2021.

[12] M. M. Fard, T. Thonet and E. Gaussier, "Deep k-means: Jointly clustering with k-means and learning representations," *Pattern Recognition Letters*, vol. 138, pp. 185-192, 2020.

[13] S. Huang, Z. Kang and Z. Xu, "Auto-weighted multi-view clustering via deep matrix decomposition," *Pattern Recognition*, vol. 97, pp. 1-11, 2020.

[14] Y. Ren, K. Hu, X. Dai, L. Pan, S. C. Hoi and Z. Xu, "Semi-supervised deep embedded clustering," *Neurocomputing*, vol. 325, pp. 121-130, 2019.

[15] A. Elnagar, R. Al-Debsi and O. Einea, "Arabic text classification using deep learning models," *Information Processing & Management*, vol. 57, no. 1, pp. 1-17, 2020.

[16] A. Kumar, K. Srinivasan, W. H. Cheng and A. Y. Zomaya, "Hybrid context enriched deep learning model for fine-grained sentiment analysis in textual and visual semiotic modality social data," *Information Processing & Management*, vol. 57, no. 1, pp. 1-34, 2020.

[17] A. E. Muller, H. M. R. Ames, P. S. J. Jardim and C. J. Rose, "Machine learning in systematic reviews: Comparing automated text clustering with Lingo3G and human researcher categorization in a rapid review," *Research Synthesis Methods*, vol. 13, no. 2, pp. 229-241, 2022.

[18] G. A. Khan, J. Hu, T. Li, B. Diallo and H. Wang, "Multi-view data clustering via non-negative matrix factorization with manifold regularization," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 3, pp. 677-689, 2022.

[19] A. Mahmoud and M. Zrigui, "Deep neural network models for paraphrased text classification in the Arabic language," *Natural Language Processing and Information Systems: 24th International Conference on Applications of Natural Language to Information Systems, NLDB 2019, Salford, UK, June 26–28, 2019, Proceedings 24*, pp. 3-16.

[20] K. Chen, R. J. Mahfoud, Y. Sun, D. Nan, K. Wang, H. Haes Alhelou and P. Siano, "Defect texts mining of secondary device in smart substation with GloVe and attention-based bidirectional LSTM," *Energies*, vol. 13, no. 17, pp. 1-17, 2020.

[21] A. Alsharef, K. Aggarwal, D. Koundal, H. Alyami and D. Ameyed, "An automated toxicity classification on social media using LSTM and word embedding," *Computational Intelligence and Neuroscience*, pp. 1-8, 2022.

[22] Z. Wen, J. Phengsuwan, N. B. Thekkummal, R. Sun, P. jamathi-Chidananda, T. Shah, P. James and R. Ranjan, "Active Hazard Observation via Human in the Loop Social Media Analytics System," *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, October 2020, pp. 3469-3472.

[23] S. H. Park, B. C. Bae and Y. G. Cheong, "Emotion recognition from text stories using an emotion embedding model," *IEEE international conference on big data and smart computing (BigComp)*, February 2020, pp. 579-583.

[24] M. R. Hossain and M. M. Hoque, "Covtexminer: Covid text mining using cnn with domain-specific glove embedding," *International Conference on Intelligent Computing & Optimization*, October. 2022, pp. 65-74.

[25] C. R. Rahman, M. D. Rahman, S. Zakir, M. Rafsan and M. E. Ali, "BSpell: A CNN-blended BERT Based Bengali Spell Checker," *arXiv preprint arXiv:2208.09709*, pp. 1-14, 2022.

[26] M. Yaseen, H. S. Salih, M. Aljanabi, A. H. Ali and S. A. Abed, "Improving Process Efficiency in Iraqi universities: a proposed management information system," *Iraqi Journal For Computer Science and Mathematics*, vol. 4, no. 1, pp. 211-219, 2023.

[27] M. Aljanabi and S. Y. Mohammed, “Metaverse: open possibilities,” *Iraqi Journal For Computer Science and Mathematics*, vol. 4, no. 3, pp. 79-86, 2023.

[28] A. S. Shaker, O. F. Youssif, M. Aljanabi, Z. Abbood and M.S. Mahdi, “SEEK Mobility Adaptive Protocol Destination Seeker Media Access Control Protocol for Mobile WSNs,” *Iraqi Journal For Computer Science and Mathematics*, vol. 4, no. 1, pp. 130-145, 2023.

[29] H. S. Salih, M. Ghazi and M. Aljanabi, “Implementing an Automated Inventory Management System for Small and Medium-sized Enterprises,” *Iraqi Journal For Computer Science and Mathematics*, vol. 4, no. 2, pp. 238-244, 2023.