Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Latent Dirichlet Allocation, Vocabulary Handler, and Domain Knowledge-Based Framework for Text classification

B. Lavanya[1], U. Vageeswari[2]

[1]Associate Professor, Department of Computer Science, University of Madras, Chennai, India., lavanmu@gmail.com
[2]Research Scholar, Department of Computer Science, University of Madras, Chennai, India., uvageeswariphd@gmail.com

Due to the remarkable advancement of information technology, the amount of unordered text information in the computer database is constantly expanding, making it difficult to organise, analyse, summarise, and classify text. The process of retrieving important data from unstructured text is called text mining. The latent Dirichlet Allocation (LDA) technique which is an unsupervised machine learning technique, is frequently employed for topic modelling. The result from an LDA makes perfect sense for categorization. Domain-specific and Out of Vocabulary (OOV) terms abound in the LDA model. This research proposes an unsupervised framework for text categorization using LDA with enhanced vocabulary handling and domain knowledge. Domain-specific terms are eliminated, and the most comparable LDA Dictionary words are used in place of OOV words. Two datasets with various data categories were used in the experiment. On both datasets, the proposed model performs better than alternative models. By using the suggested framework, Accuracy, Purity, Precision, Recall, and F1- scores were all improved.

Keywords: LDA, Topic Modelling, Domain Terms, Out of Vocabulary (OOV), Text Classification.

1. Introduction

Due to the remarkable advancement of information technology, the amount of unordered text information in the computer database is constantly expanding, making it difficult to organise, analyse, summarise, and classify text [1]. The process of retrieving important data from unstructured text is called text mining. Text mining involves Organizing the input text, identifying patterns in the datasets, analysing sentiments, Named entity recognition, Relation extraction, Part-of-speech tagging, Text summarization, and then evaluating and interpreting the findings. It is possible to classify enormous quantities of textual content to assist in standardizing the service, improve search relevancy and effectiveness and show a better experience by rendering navigation easier. Considering manual classification consumes a substantial amount of labour, resource, and expense, automated fast and accurate text classification is essential. Text categorization classifies documents automatically using machine learning or other technologies. The following levels can be used for text classification [2]:

Document Level: At this level, the classification model handles the complete document as one entity.

Paragraph Level: At this level, the classification model handles the complete paragraph as one entity.

Sentence Level: At this level, the classification model handles the complete sentence as one entity.

Sub-Sentence Level: At this level, the classification model handles the complete sub-expression as one entity.

Text Classification can be done using the Supervised or Unsupervised learning method. The type of data utilised for training is the primary distinction between the supervised and unsupervised learning approaches. Supervised Learning occurs when data and its label are utilised for training. Unsupervised learning occurs when data alone is used in training. In general, supervised approaches outperform unsupervised methods in terms of accuracy. Due to the lack of a label during training, the unsupervised method underperforms when compared to the supervised method. However, because most text data is unlabelled, a better-unsupervised learning strategy for text classification is needed. This paper proposes the unsupervised framework for text classification using LDA with Domain Knowledge(LDA-DK) and Vocabulary handler(VH) with better accuracy.

1.1. Unsupervised Learning

Unsupervised learning methods are appropriate when the output variables are not supplied. It is appropriate for problems in which the algorithms identify and extract similarities between the inputs so that similar inputs can be categorized together. It enables the machine to find patterns itself in large data sets. Unsupervised learning algorithms may easily manage more complex data processing tasks than supervised learning systems. Unsupervised learning seeks to differentiate between input items using similarities discovered by the system in training data sets. The algorithms examine the underlying representation of the data sets by eliminating relevant features or characteristics from the data sets. Unsupervised machine learning can identify patterns in data that were not previously detected. Checking accuracy is difficult since there are no labelled data sets to evaluate the results. Understanding and recognising the outcomes of unsupervised learning takes time. The technique of anticipating output in unsupervised learning is not well understood. In topic modelling, an unsupervised method of machine learning known as Latent Dirichlet Allocation (LDA) is successfully applied. This linguistics algorithm’s result may be rationally applied to group documents. The LDA module contains domain-specific jargon and Out of Vocabulary (OOV) terms. This work presents a method for categorising text that makes use of a vocabulary handler(VH), domain knowledge, and latent Dirichlet allocation.

This article's remaining sections are divided into different sections: The many LDA techniques and their modifications are covered in Section 2. Section 3 explains the various methods of classification. The proposed framework is outlined in Section 4. In Section 5, the dataset examined is described, the results are tabulated, and it is shown how the suggested framework performs better than existing frameworks. Section 6 of this research paper provides the conclusion.

2. Related Works

One of the hybrid approaches [3] combines the information obtained from word embeddings with the Latent Dirichlet Allocation technique. The content of the message in subjects is expanded using Word2Vec. The hybrid technique primarily takes the Latent Dirichlet Allocation (LDA). The model words collected by Word2Vec are incorporated in the algorithm's second stage. By employing keywords that are like the words that are used the most frequently for each topic, each topic is intended to be expanded. On three of the four datasets used in the evaluation, the hybrid technique came out on top.

The most pertinent category is created using the gLDA technique for topic text classification, which extends LDA by including a topic category distribution variable [4]. The model's documents are broken down into categories, and each category has a unique set of "themes." By determining the category to which it is most connected, each document is generated in the class to which it is more likely to belong. Limiting the producing scope using the topic-category distribution option, removes improper topic-word assignment.

Based on the KNearest Neighbor algorithm and the LDA topic model, an improved short-text categorization method was presented [5]. The sentences become more semantically cohesive and less sparse thanks to the probabilistic themes that are formed. Additionally, it presents a novel topic similarity metric based on the association between discriminative phrases in two short texts and a specific subject matrix. The construction of a small text dataset and experimental validation happens because of searching for articles on the Sina News website. A rational software on a generative probabilistic framework was also offered, along with a unique technique for determining how similar short texts are.

A unique method of topic modelling is offered [6] by treating the document as a group of word representations and the subjects as normal distributions in the source data. It examines several compressed sampler methodologies and creates a flexible procedure that enhances incorporation asymptotically. Current classification algorithms do not handle domain-specific sentences or uncommon words. The proposed classification approach solves these issues.

3. Methodology

3.1. Topic Modelling

A probabilistic framework is offered by Topic Models (TM), which are generative for the processing of language form and machine learning [7]. The subject methodology's output can be employed to reduce dimensionality, recommendation systems, text classification or clustering, among other NLP and information retrieval tasks. The three types of TM techniques are semi-supervised, unsupervised, and supervised.

It is possible to use both unstructured and structured data. Farming, education, medicine, social media analysis [8], production, bioinformatics, banking, reinforcement learning, and other sectors are a few of the application fields for TM. Topic modelling is the process of identifying the topics that most effectively describe a group of materials. These themes will only come up when the topic modelling process progresses.

The following fundamental premise forms the basis of all topic models:

- Each article includes a variety of topics.
- Each topic is made up of several words.

Analysing a text's topics is one of the most beneficial methods for text interpretation at the document level. Understanding, recognising, and extracting subjects from a corpus of documents is the process of topic modelling. There are several topic modelling techniques, some of which are used rather regularly.

- Latent Dirichlet Allocation (LDA)[9]
- Parallel Latent Dirichlet Allocation (PLDA)
- Pachinko Allocation Model (PAM)
- Non-Negative Matrix Factorization (NMF) [10]
- Latent Semantic Analysis (LSA) [11]

LDA is the most widely utilised TM approach among them [12].

3.2. Word Embedding

When constructing representations of words, a variety of word embedding techniques contribute to capturing the interpretation, semantic link, and context of various words. The process of word embedding produces dense vector representations of words that incorporate certain context keywords. Word embeddings use a method to create finite dense vectors and consistent vectors based on a sizable text corpus. Each word is learned and managed as a point in vector space that is positioned around the target word while maintaining semantic linkages. Words with comparable content are clustered together in a projection created by the vector space model of words.

3.3. Word2Vec

Word embedding is the most popular technique to convey document vocabulary. In addition to other things, it can identify a word's connection to other words, its perspective in a document, and its semantic and syntactic similarities. One of the most famous shallow neural network techniques for generating word embeddings is called Word2Vec. In 2013, Tomas Mikolov developed it for Google.

3.4. GloVe

Global Vectors for Word Representation GloVe is a method of unsupervised learning that creates word embedding. It is an unsupervised learning method developed by researchers at

Stanford University. The representations that come from the training reflect the major linear subsets of the word vector space and are based on metrics of global word-word co-occurrence that were gathered from a corpus. Word2vec only have local language information. That is, the words around a word have the only effect on the semantics that are learnt for that word. GloVe gathers both global and local data from a corpus to produce word vectors. The foundation of the GloVe technique is the notion that from the co-occurrence matrix, semantic links may be inferred between words.

3.5. Latent Dirichlet Allocation (LDA)

Latent techniques are hidden. Blei et al. [9] provided the first description of the Dirichlet probability distribution type. Un - supervised Latent Dirichlet Allocation gives a score to every document for each selected topic [13]. The documents may be accurately categorised in accordance with their topic using LDA results [14]. Fig (1). depicts the plate representation towards the topic model LDA Approach. The boxes serve as "plates," or substitutes for repeated or frequent things. The documents are shown on the outside plate, and the recurrent word positions, each connected with a topic and word choice, are displayed on the inside plate.

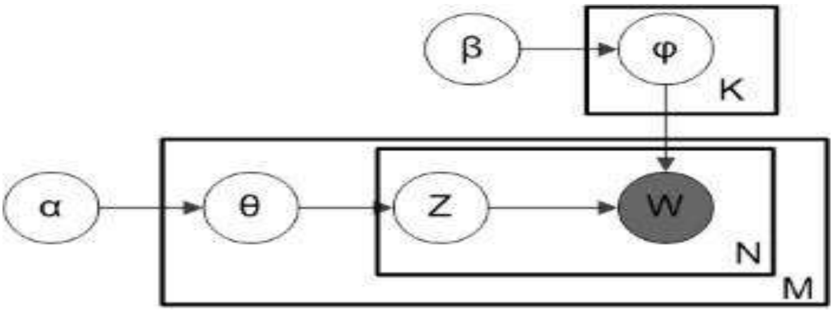


Figure 1: LDA Plate notation

The boxes stand in for repetitions or recurrent events, or "plates." The papers are displayed on the outer plate, and the repeated word placements are displayed on the inner plate; each location is associated with a topic and word choice.

M- Indicates how many documents there are.

N_i - The total number of words in each ith document.

α - The Dirichlet prior value on the topic probabilities for each document.

β - The Dirichlet prior value on the word probabilities for each topic.

K - Represents the number of topics.

θ_i -indicates the proportion of topics for the document i.

ϕ_k - indicates the proportion of words for topic k.

z_{ij} – It is the value of the topic for the jth word in document i.

W_{ij} -It is the precise word on document i in the jth position.

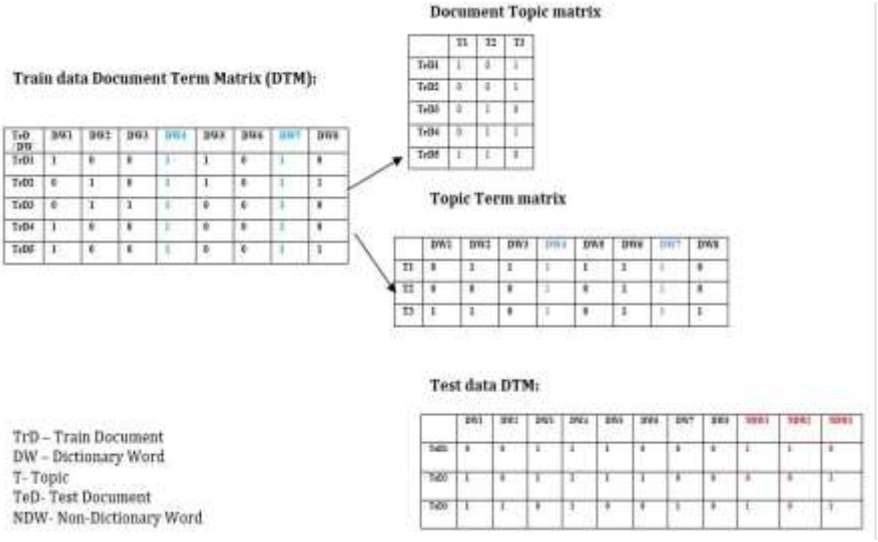
According to LDA, every other text may be represented as a probability distribution beyond a latent theme if all texts have a common Dirichlet prior. Each latent topic is further represented in the LDA model as a probabilistic distribution across words, with each topic's word distribution having a unique Dirichlet prior. Given a corpus D of M documents, each document d containing Nd words (d 1, 2, 3, … M), LDA models D in accordance with the subsequent generating process.

W is the main observable variable, while the others are latent variables. In the aforementioned generating process, terms in texts are just observable variables; additional variables (ϕ and θ) are latent variables, and (β and α) are model parameters. The formula below is used to determine the likelihood that value D will be seen for a corpus.

$$p(D|\alpha,\beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}),\beta \right) d\theta_d$$

Along with the topic of Dirichlet prior parameters, the Dirichlet distribution's result, the proportion of words across subjects, has also been defined. The topic densities at the text level are computed using the Dirichlet multinomial duos (θ, α). The topic densities at the word level are computed using the Dirichlet multinomial duos (ϕ, β). A single sample of the document-level variable D is taken for each document.

4. Proposed Classification Framework



Train data Document Term Matrix (DTM):

TrD /DW	DW1	DW2	DW3	DW4	DW5	DW6	DW7	DW8
TrD1	1	0	0	1	1	0	1	0
TrD2	0	1	0	1	1	0	1	1
TrD3	0	1	1	1	0	0	1	0
TrD4	1	0	0	1	0	0	1	0
TrD5	1	0	0	1	0	0	1	1

Document Topic matrix

	T1	T2	T3
TrD1	1	0	1
TrD2	0	0	1
TrD3	0	1	0
TrD4	0	1	1
TrD5	1	1	0

Topic Term matrix

	DW1	DW2	DW3	DW4	DW5	DW6	DW7	DW8
T1	0	1	1	1	1	1	1	0
T2	0	0	0	1	0	1	1	0
T3	1	1	0	1	0	1	1	1

Test data DTM:

	DW1	DW2	DW3	DW4	DW5	DW6	DW7	DW8	NDW1	NDW2	NDW3
TeD1	0	0	1	1	1	0	0	0	1	1	0
TeD2	1	0	1	1	1	1	0	0	0	0	1
TeD3	1	1	0	1	0	0	1	0	1	0	1

TrD – Train Document
DW – Dictionary Word
T- Topic
TeD- Test Document
NDW- Non-Dictionary Word

Figure 2: Matrix Conversion

The proposed model in matrix representation is shown in Fig (2). The Topic Term Matrix and Document Topic Matrix are created by LDA from the Document Term Matrix (DTM). In the LDA matrix conversion procedure, each document is first given a distinct ID, which is subsequently assigned to each distinct word in the document. Rows in the Document Term

Matrix (DTM) correspond to documents and columns to words.

The document is tagged as 1 if it includes the term; otherwise, it is marked as 0. Two matrices are created from a document term matrix. The first one is a topic matrix for documents, while the other one is a term matrix for topics. Matrix columns indicate words in the Topic Term, whereas rows represent subjects. A word is marked as 1 if it belongs to a subject; otherwise, it is marked as 0. Finding the most ideal representation of the Topic Term matrix and Document Topic Matrix is LDA's ultimate objective. LDA is a procedure that is iterated. It will iterate across each document and then each word. When the document-topic and topic-term matrices are produced in the most optimal way, LDA has reached its convergence point.

However, the LDA suffers from Domain-specific terms and Out of Vocabulary words. In DTM Domain specific words are shown as blue letters. These terms are not general stop words, but these are present in all documents since they are specific to the domain. Here in DTM DW4 and DW7 are Domain-Specific Terms. In test document DTM there are some terms which do not present in training. These terms are shown in red colour. In testing Non-Dictionary Words may be present. It is shown in Test Data DTM. NDW1, NDW2, and NDW3 are OOV Words it is handled in predicting the topics on the testing document.

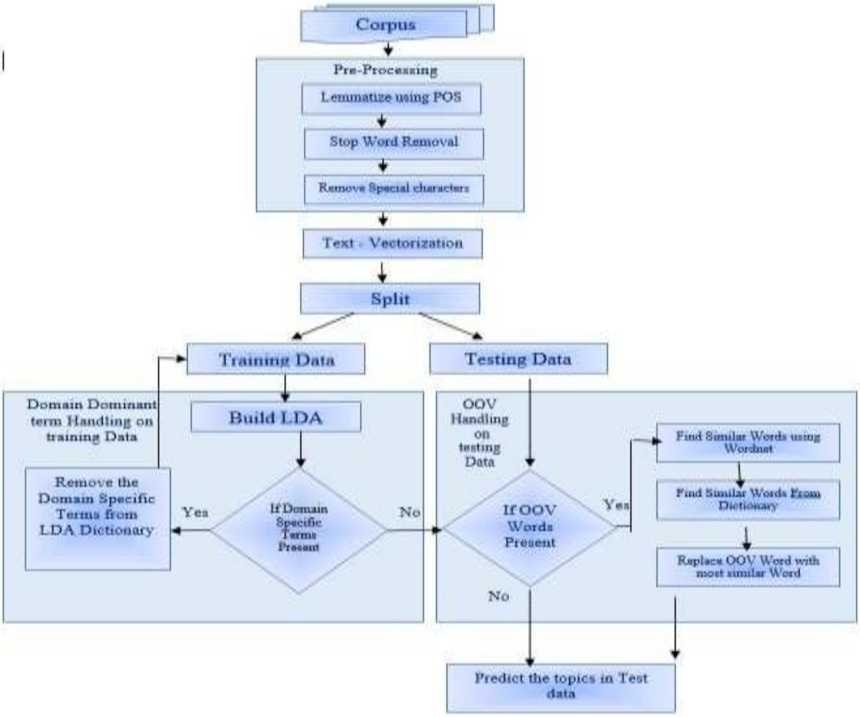

Corpus
Pre-Processing
Lemmatize using POS
Stop Word Removal
Remove Special characters
Text - Vectorization
Split
Training Data
Testing Data
Domain Dominant term Handling on training Data
Build LDA
Remove the Domain Specific Terms from LDA Dictionary
Yes
If Domain Specific Terms Present
No
OOV Handling on testing Data
If OOV Words Present
Yes
No
Find Similar Words using Wordnet
Find Similar Words From Dictionary
Replace OOV Word with most similar Word
Predict the topics in Test data

Figure 3: Proposed Document Classification FrameWork

The domain-specific terms that have a greater impact on the LDA model than other subject words are eliminated using the proposed framework. Domain-specific terms will not be removed by the generic stop word removal. More crucially, only the terms that appear in all subjects should be disregarded in the topic identification process rather than all domain-specific words. The domain-specific word's weight is now distributed across other terms. A fuzzy set is the model's output. The topic of the document is regarded as having great weight.

Three phases constitute the recommended text classification framework. The pre-processing stage is step one. The inappropriate stem word recognition problem is overcome in this step. The training phase is phase two. In this step, the domain-specific word difficulty is resolved. The Out of Vocabulary issue is resolved in Phase 3, which is also known as the Testing Phase.

Algorithm1: Find POS

```
Input: W
    /* W - word                                                  *
                                                                 /
    Output: T
    /* T – tag                                                   *
                                                                 /
Function get POS(W )
    tag ← nltk.pos tag(W)
    tag dict ←{ J: wordnet.ADJ,
                V: wordnet.VERB,
                N: wordnet.NOUN,
                R: wordnet.ADV  }
    T ← tag dict(tag)
    return T
```

Algorithm2: Find Root Word

```
Input: W
/* W - word                                                      */
Output: R
/* R - root word                                                 */
Function Lemmatize ( W )
    token pos ← CALLget POS(W)
    R ← Wordnet.Lemmatize(W,token pos)
    return R
```

Phase 1: The pre-processing procedures for this system include lowercase conversion, tokenization, stop word removal, stemming, special character removal, and lemmatization using part of speech. The nltk POS technique is used by the Find POS algorithm (1) to determine the Part of Speech of a given word. The Find Root Word algorithm (2) finds the root word of the given word using Lemmatize method from wordnet and POS. Algorithms (1) and (2) are used in both training and testing documents. Lemmatizing with a tag improves the accuracy of predicting root words. On the Lemmatization process also here, we improved using the POS tag.

Algorithm 3: DST: Domain-specific Terms

```
Input: SWL
    /* SWL - Subject Words List                              */
    Output: List of Domain-Specific Words-DSW

1 Function DSW (SWL)
    //  SOL - Subject overlap
    $SOL \leftarrow$ SWL [0]
    $DSW \leftarrow \emptyset$
    foreach $WL \in SWL$ do
        $DSW \leftarrow WL \cap SOL$
        $SOL \leftarrow DSW$
    return DSW
```

Phase 2: The LDA model is generated during training, and any domain-specific terms are discovered and deleted using the DST technique. The LDA Model is then built again, and the procedure is repeated until a stable LDA model is created. The LDA Classification model was built by employing train data. The proposed technique looks for the upper n domain-specific keywords in every topic set of words. Whenever there are no synonyms relevant to a certain place, use the system to anticipate test data topics. Get rid of any domain-specific terms from the vocabulary before rebuilding the LDA model. Using this method, domain-specific terms from the LDA lexicon that significantly affect topic allocation are found and removed. The LDA using the Domain Knowledge Framework is shown in Figure 3. The framework receives its input from the corpus, which is a collection of texts that will be classified by topic. Domain-specific words in training data are handled by the DS algorithm (3). The algorithm shows where to search for domain-specific phrases. The LDA model provides input to the algorithm in the form of a number of domain word lists. The algorithm searches for words that appear in all topic word lists and are more likely to increase the topic's overall likelihood. The Loop searches through each subject word list repeatedly to identify the domain-specific terms that carry a significant amount of weight. During training, the LDA classification model is created, and any domain-specific words are found and removed via the DST algorithm. The LDA

Model is then built again, and the procedure is repeated until a stable LDA model is created. The testing component receives this model as input.

Algorithm4: FindSynonyms

```
Input: W
    /* W - Word                                                    */
    Output: S
    /* S - List of Synonyms words                                  */
Function get Synonyms (W )
    S ← []
    foreach synset ∈ wordnet. synsets(W) do
        foreach lemma ∈ Synset. Lemmas() do
            S ← S+lemma.name ()
    return S
```

Algorithm 5: Find Most Similar Word Input: W,S

```
    /* W-Word,S-list of synonyms words                             */
    Output: MSW
    /* MSW - most similar word                                     */
Function get Similar (W, S)
    word-score ←{}
    x ← glove[W]
    foreach i ∈ S do
        y ← glove[i]
        score ← torch.cosine -similarity (x,y )
        word-score[i] ← score
    MSW ← max(word score)
    return MSW
```

Phase 3: The model and test documents are used as input in testing. The OOV handler is called if the testing document contains OOV terms. The Handler seeks similar words to the OOV term first, then the most similar one, and then all occurrences of the OOV word are replaced with the Most Similar word in the last phase. For all OOV terms, the same procedure is followed.

The topic of the test document is predicted using the LDA Model. The tested data is utilized to identify the topics using the trained LDA model. And in the classification task, it is put into a certain class according to the topic. After training, the PPTD algorithm is used to process the test papers (6). The Major issue in testing documents is OOV words. Out Of Vocabulary Words are not handled in the testing data in the normal classification model. The proposed model identifies OOV words and finds the most similar word to the OOV Word using Wordnet and Glove models. On Predicting the topic in the test document first normal pre-processing is performed. Then the OOV Words are identified. If there are any OOV words present in the test document similar words are found Using Wordnet using the find synonyms algorithm (4).

Finding the word that is the most comparable is necessary since there are many words that are like the OOV term. Using the Find most similar word algorithm (5) Most Similar Word is identified and replaced in the testing document. The Algorithm uses Glove Word2Vec and Cosine Similarity methods to find the most similar Words.

Algorithm 6: PPTD: Pre-process Test Data for OOV Words

```
Input: Text, LDA Dict, SW, DST
/* Text - Text to be checked Topic, LDA Dict -Dictionary from
Train Data , SW - General Stop Words of Language, DST -
Words Specific to Domain                                    */
Output: Words to predict
/* Words to predict - Words in Test document to predict Topic */
```

```
Function Pre-process Test Data (Text, LDA Dict, SW, DST)
    $Words\ to\ predict \leftarrow$ []
    foreach $token \in Text$ do
        if $token \notin SW$ then
            $Word \leftarrow$ CALL Lemmatize(token)
            if $Word \notin DST$ then
                if $Word \notin LDA\ \ Dict$ then
                    $Synonyms \leftarrow$ CALL gets  Synonyms (Word)
                    $MSW \leftarrow$ CALL gets  Similar (Word, Synonyms)
                    $Words\ to\ predict \leftarrow$ MSW
                else
                    $Words\ to\ predict \leftarrow$ Word
    return  Words to predict
```

5. Results and Discussion

5.1. System Implementation

The proposed model has been tested using the Microsoft Windows 10 OS, an Intel Core i3 processor from the 10th generation with an integrated GPU, 4GB of DDR4-2667 memory, and a 256GB PCIe NVMe SSD. The models are run in Jupyter Notebook with python. The model utilized the following python libraries nltk for Part of Speech identification, and Lemmatization, torch and torch text libraries for Glove word2voc representation.

5.2. NEWS Dataset

The BBC News dataset, which contains five types of data—sport, business, politics, technology, and amusement employed for our experiment. There are 2225 NEWS documents in the collection. Table 1 illustrates the volume of data from every category. This data set is a labelled dataset, but no label is used in training. This dataset has been chosen because evaluation labels are required.

Table 1: The NEWS Dataset's number of documents in every category

Document Category	Documents Count
Sport	511
Business	510
Politics	417
Technology	401
Entertainment	386

5.3. Abstract Dataset

The second dataset that has been used for the experiment is the abstract dataset. The dataset has four categories of data namely Computer Science, Maths, Statistics, and Physics. This dataset contains 4000 documents. It is also a labelled dataset. The quantity of documents in each category is shown in Table 2.

Table 2: The Abstract Dataset's number of documents in every category

Document Category	Documents Count
Computer Science	1000
Maths	1000
Statistics	1000
Physics	1000

5.4. Result discussion

The suggested framework is evaluated in comparison against other LDA methods. Table (3) below lists the top 5 words from the generic LDA technique in the Abstract and NEWS dataset for all disciplines. The DST algorithm identifies the bold words as domain-specific terms that have a greater effect on the model and provide biased results. The top 5 terms across all subjects from the proposed framework in the NEWS and Abstract dataset are displayed in Table (4).

"SAY" and "YEAR" are two of the top five terms in the NEWS dataset that have been recognised as domain words. Since the term "say" appears in nearly all media articles in some variation, including "say," "said," and "saying." And the year word appears in every news item. The top five terms in the Abstract dataset's identified domain words are "MODEL" and "PAPER." Almost all abstractions contain the word "model" in some form, such as model, modelling, or model. And the word paper is present in almost all abstracts.

Table 3: Topic words from Standard LDA in NEWS and Abstract Dataset

NEWS Dataset					Abstract Dataset			
Theme 0	Theme 1	Theme 2	Theme 3	Theme 4	Theme 0	Theme 1	Theme 2	Theme 3
say	say	say	game	say	field	model	paper	test
govern	year	year	Say	company	base	algebra	model	paper
labour	people	best	england	year	model	group	electron	model
elect	mobile	play	Play	firm	paper	result	space	propose
year	service	film	year	sale	network	paper	problem	point

Table 4: Topic words from proposed framework knowledge in NEWS and Abstract Dataset

NEWS Dataset					Abstract Dataset			
Theme 0	Theme 1	Theme 2	Theme 3	Theme 4	Theme 0	Theme 1	Theme 2	Theme 3
labour	company	govern	Play	film	algorithm	network	time	estimate
elect	market	country	Time	people	space	user	energy	algebra
party	rise	people	Player	music	time	provide	perform	distribute
people	firm	world	England	best	prove	present	different	sample
blair	bank	nation	Go	mobile	case	system	approach	group

The topic word list weighting for the NEWS dataset's standard LDA Model is displayed in Table (5). The subject word list weighting for the NEWS dataset's conventional LDA Model is displayed in Table (6).

Table 5: Standard LDA topic words and weightage in the dataset

Theme 0	0.028*”say” + 0.018*”peopl” + 0.018*”govern” + 0.015*”minist” + 0.005*”elect”
Theme 1	0.018*”say” + 0.059*”club” + 0.0058*”game” + 0.0059*”play” + 0.0049*”time”

Theme 2	0.023*”say” + 0.017*”year” + 0.0079*”market” + 0.0052*”rise” + 0.0049*”price”
Theme 3	0.0119*”film” + 0.009*”say” + 0.009*”year” + 0.008*”best” + 0.0079*”play”
Theme 4	0.0191*”say” + 0.009*”game” + 0.007*”year” + 0.007*”music” + 0.006*”peopl”

Table 6: Standard LDA topic word and weightage in the Abstract dataset

Theme 0	0.006*”field” + 0.005*”base” + 0.005*” model ” + 0.005*” paper ” + 0.005*” network ”
Theme 1	0.011*” model ” + 0.009*” algebra ” + 0.008*” group ” + 0.008*” result ” + 0.007*” paper ”
Theme 2	0.007*” paper ” + 0.007*” model ” + 0.005*” electron ” + 0.005*” space ” + 0.005*” problem ”
Theme 3	0.017*” test ” + 0.011*” paper ” + 0.011*” model ” + 0.010*” propose ” + 0.008*” point ”

The subject word list weighting in the proposed framework in the NEWS dataset is shown in Table (7). The subject word list weighting in the proposed framework in the Abstract dataset is shown in Table (8).

Table 7: Proposed framework’s topic word and weightage in News dataset

Theme 0	0.019*”compani” + 0.018*”firm” + 0.017*”phone” + 0.016*”mobil” + 0.015*”peopl”
Theme 1	0.0159*”rise” + 0.016*”market” + 0.016*”world” + 0.017*”month” + 0.015*”price”
Theme 2	0.014*”film” + 0.0059*”peopl” + 0.006*”music” + 0.005*”best” + 0.005*”star”
Theme 3	0.0079*”govern” + 0.0077*”peopl” + 0.0065*”elect” + 0.0069*”labour” + 0.0065*”parti”
Theme 4	0.0075*”play” + 0.017*”england” + 0.016*”time” + 0.0049*”best” + 0.015*”player”

Table 8: Proposed framework’s topic word and weightage in Abstract dataset

Theme 0	0.011*”algorithm” + 0.017*”space” + 0.016*”time” + 0.006*”prove” + 0.015*”case”
Theme 1	0.009*”network” + 0.007*” user ” + 0.005* ” provide ” + 0.015*” present ” + 0.015*” system”
Theme 2	0.006*”time” + 0.005*”energy” + 0.005*”perform” + 0.005*”different” + 0.005*”approach”
Theme 3	0.013*”estimate” + 0.009*”algebra” + 0.008*”distribute” + 0.006*”sample” + 0.006*”group”

Results for the NEWS and Abstract datasets are shown in Tables (9) and (10) respectively. Precision, Recall, Accuracy, Purity, and F1-Score are used to evaluate the suggested framework. Accuracy, Purity, Precision, Recall, and F1 score increased by 20%,21%,22%, 20%, and 15%, respectively, as a result of the domain knowledge framework and OOV handler.

Table 9: Results for NEWS dataset

Classification Model	Accuracy	Purity	F1-Score	Precision	Recall
LDA	0.52817	0.57746	0.49535	0.53199	0.51173
LDA + TF-IDF	0.53756	0.53756	0.44774	0.53362	0.54855
LDA-DK	0.71586	0.71594	0.70510	0.71162	0.71066
LDA-DK +VH	0.72300	0.72300	0.71700	0.73268	0.72300

Table 10: Results for Abstract dataset

Classification model	Accuracy	Purity	F1 Score	Precision	Recall
LDA	0.65369	0.63552	0.58633	0.64626	0.63521
LDA + TF-IDF	0.3558	0.35751	0.11740	0.48209	0.25583
LDA-DK	0.63916	0.659160	0.78621	0.74615	0.73200
LDA-DK + VH	0.79164	0.7516	0.8161	0.78615	0.75200

Table (11) lists a few OOV terms along with the MSW words that the model suggests should replace them.

5.5. Statistical Test

To find out if there is a statistically significant distinction between the two classifiers applied the paired t-test. N separate test sets must be employed in this paired t-test in order to evaluate each classifier. Fortunately, we can perform k-fold cross-validation instead of truly requiring N test sets. This methodology allows you to test one model on several test sets to get more robust results even when you have limited data.

Table 11: OOV Words & Replaced Words IN NEWS and Abstract Dataset

NEWS Dataset		Abstract Dataset	
OOV Word	Most Similar Word	OOV word	Most Similar Word
accredit	Recognise	demerit	fault
sofa	Couch	pretty	fairly
substantiate	Corroborate	detachment	separation
decamp	Skip	alters	alter
supplementary	Auxiliary	actualize	realise
obese	Weighty	unwanted	undesirable
requisite	Requirement	judgment	assessment
aurora	Morning	renowned	famous
totality	Entirety	unusual	strange
mesmerise	Hypnotise	fruitfully	profitably
intuition	Suspicion	fraud	fake
symbolise	Represent	argues	debate
raring	Impatient	prolong	extend
cowardly	Fearful	unattackable	strong
luminosity	Light	Trick	magic

NULL Hypothesis H0: The Proposed classification framework with LDA does not improve classification accuracy significantly when compared with Standard LDA classification.

Alternate Hypothesis H1: The Proposed classification framework with LDA improves classification accuracy significantly when compared with Standard LDA classification.

The test statistic T for the NEWS dataset is equal to 3.7251, which is beyond the 95% acceptability range [-2.7764, 2.7764]. There is a 0.009974 p-value. H0 is disproved since the p-value is less than 0.05. This implies that the likelihood of a type I mistake is minimal (rejecting a valid H0).

The test statistic T for the Abstract dataset is equivalent to 4.084821, which is beyond the accepted 95% range of [-2.7764, 2.7764]. There is a 0.007549 p-value. H0 is disproved since the p-value is less than 0.05. This implies that the likelihood of a type I mistake is minimal (rejecting a valid H0).

The proposed classification framework with LDA's accuracy is higher on average than the accuracy of the Standard LDA classification. In other words, the Standard LDA and Proposed Framework averages differ statistically significantly from one another. It is large enough for the Proposed Framework with LDA to be statistically significant.

6. Conclusion

A vital part of text mining is performed by topic models. In computer science, there are several text classification models. Since LDA is a probability-based method, it has been selected. The Inputs to the LDA model are document collection, the number of topics and there are two hyperparameters alpha and beta. In comparison to prior cutting-edge LDA models, the suggested LDA with domain knowledge and Vocabulary handler structure produces superior results. The top five words are tested from the vocabulary for domain-specific words; however, the number can be altered based on the size of the samples and the vocabulary. Accuracy, Purity Precision, Recall, and F1 scores were all increased by 20%, 21%, 22%, 20%, and 15%, respectively, using the Proposed Framework. By lowering dimensionality during the training phase, this work can be further expanded in the future.

Acknowledgement

Under the Junior Research Fellowship programme, University Grants Commission is sponsoring this research (JRF).

References

1. B. Lavanya, U. Vageeswari, An assortment of query-based summarization technique (qbs)–a study, in: Proceedings of the International Conference on Innovative Computing and Communication (ICICC), http://dx.doi.org/10.2139/ssrn.3833295, 2021.
2. K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text classification algorithms: A survey, Information 10 (4) (2019) 150.
3. J. Jedrzejowicz, M. Zakrzewska, Text classification using a lda-w2v hybrid algorithm, in Intelligent Decision Technologies 2019, Springer, 2020, pp. 227–237.
4. D. Zhao, J. He, J. Liu, An improved LDA algorithm for text classification, in: 2014 International Conference on Information Science, Electronics and Electrical Engineering, Vol. 1, IEEE, 2014, pp. 217–221.
5. Q. Chen, L. Yao, J. Yang, Short text classification based on LDA topic model, in: 2016 International Conference on Audio, Language and Image Processing (ICALIP), IEEE, 2016, pp. 749–753.
6. R. Das, M. Zaheer, C. Dyer, Gaussian lda for topic models with word embeddings, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 795– 804.
7. S.-B. Kim, K.-S. Han, H.-C. Rim, S. H. Myaeng, Some effective techniques for naive Bayes text classification, IEEE transactions on knowledge and data engineering 18 (11) (2006) 1457–1466.
8. O. Aborisade, M. Anwar, Classification for authorship of tweets by comparing logistic regression and naive Bayes classifiers, in: 2018 IEEE International Conference on Information Reuse and Integration (IRI), IEEE, 2018, pp. 269–276.
9. D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, the Journal of machine Learning research 3 (2003) 993–1022.
10. M. W. Berry, M. Browne, Email surveillance using non-negative matrix factorization, Computational & Mathematical Organization Theory 11 (3) (2005) 249–264.
11. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R. Harshman, Indexing by latent semantic analysis, Journal of the American society for information science 41 (6) (1990) 391–407.
12. G. N. Gopal, B. C. Kovoor, U. Mini, Keyword template based semisupervised topic modelling in tweets, in: International Conference on Innovative Computing and Communications, Springer, 2021, pp. 659–666.
13. B. Lavanya, U. Vageeswari, A machine learning framework for document classification by topic recognition using latent dirichlet allocation and domain knowledge, in: International Conference on Innovative Computing and Communications, Springer, 2023, pp. 509–520.
14. M. Pavlinek, V. Podgorelec, Text classification method based on selftraining and lda topic models, Expert Systems with Applications 80 (2017) 83–93.