Multimedia Tools and Applications (2021) 80:3275–3305
https://doi.org/10.1007/s11042-020-09549-3

An extractive text summarization approach using tagged-LDA based topic modeling

Ruby Rani[1] · **D. K. Lobiyal**[1]

Received: 18 February 2020 / Revised: 26 June 2020 / Accepted: 6 August 2020 /
Published online: 19September 2020

Abstract
Automatic text summarization is an exertion of contriving the abridged form of a text document covering salient knowledge. Numerous statistical, linguistic, rule-based, and position-based text summarization approaches have been explored for different rich-resourced languages. For under-resourced languages such as Hindi, automatic text summarization is a challenging task and still an unsolved problem. Another issue with such languages is the unavailability of corpus and the inadequacy of the processing tools. In this paper, we proposed an extractive lexical knowledge-rich topic modeling text summarization approach for Hindi novels and stories in which we implemented four independent variants using different sentence weighting schemes. We prepared a corpus of Hindi Novels and stories since the absence of a corpus. We used a smoothing technique for edifying and variety summaries followed by evaluating the efficacy of generated summaries against three metrics (gist diversity, retention ratio, and ROUGE score). The results manifest that the proposed model produces abridge, articulate and coherent summaries. To investigate the performance of the proposed model, we simulate the experiments on the English dataset as well. Further, we compare our models with the baselines and traditional topic modeling approach, where we show that the proposed model has confessed optimal results.

Keywords Topic modeling · Hindi novel · Topic diversity · Retention ratio · Tagged-LDA

1 Introduction

Due to the breakthrough in technologies like Big data, cloud computing, wireless communication, sensors, and the internet of things, a huge amount of digital data have congregated on

✉ Ruby Rani
ruby73_scs@jnu.ac.in

D. K. Lobiyal
dkl@mail.jnu.ac.in

[1] School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi, India

the internet. From this enormous digital data, the user entails only useful information instantly. Thus, it has become a challenge to excerpt indispensable information from a large corpus effectively and convincingly. One method is to condense the data without losing their proficient content. Some conventional methods need manual effort for condensing document but they demand an insignificant time. Constructing an automatic summary generation system could be effective, in terms of time and human efforts. Recently, in the last decade, a novel Automatic Text Summarization (ATS) system introduced to generate a concise and accurate form of large digital textual information by covering all required information [14]. The main objective of ATS is to collect the relevant and traceable points of a large document into a small space. Nowadays, ATS has numerous effective real-time applications, like in opinion mining [19], review mining [10, 55], and question answering systems [48].

There are two approaches to generate automatic text summaries: extractive and abstractive. The extractive methods score sentences based on their importance and select highly scored sentences with minimum redundancy. The abstractive text summarization methods extract essential information, locution and paraphrase the original text. Although, abstractive methods are more familiar to the human summary which are more accurate, rational, and articulate [1] but complex. Based on the cardinality of the documents, ATS is divided into two categories: single document and multi-document summarization. In single document summarization, the digest of one long text document is converted to a small document [9]. While multi-document summaries include unspecified themes associated with different documents [12]. In practice, text summaries are mainly evaluated based on three evaluation criteria: topics diversification (gist diversity), redundant information (retention ratio), and readability of summary (ROUGE score). However, it is arduous to maintain a balance among the three criteria [14].

1.1 Motivation and contribution

There have been discussed several types of automatic document summarizer for English [42] and other foreign languages, such as Arabic [2, 38], Chinese [53, 54], French [23], and German [22]. Some ATS systems also developed for a few Indian languages, including Gujarati [41], Punjabi [16], and Urdu [17, 41]. It is a study that nearly 260 million native speakers around the world speak and understand the Hindi language. It is well-known that Hindi is one of the two official languages of India and 4th most spoken language globally. Due to the inadequacy of Hindi tools and the deprivation of standard datasets, insubstantial research has been carried out for short length Hindi text documents only. It is challenging to construct an ATS system for lengthy Hindi text documents such as novels and stories. It is to note that a novel document has a free flow structure and an average length of around 40,000 words. While the average length of a story is around 20,000 words. Unfortunately, the existing systems are not suitable for summarizing novels. Besides the length of document, another challenge for summarizing long document is the compression ratio. The length of a text document affects the system efficiency in terms of computation cost [51]. Therefore, constructing an efficient ATS for lengthy Hindi documents is becoming a challenge for current researchers and academicians. Currently, many industries and academicians are working on Hindi text in various text mining areas such as sentiment analysis, document summarization, text categorization, etc. This motivates us to construct an automatic text summarizer for the novel documents in the Hindi language.

In this paper, we proposed a lexical feature-rich topic modeling based automatic summarizer for novel and story documents that yields coherent and diverse summaries. In summary, the proposed work has the following contributions.

- First, we propose a Parts of Speech (POS) tagger for the Hindi language, which forages the linguistic features. Further, these features assist in discovering a consequential set of topic-words from the document.
- We suggest four disparate variants of the proposed POS tagging based topic modeling system differs in sentence weighting strategies to generate a coherent, articulate, and lucid summary.
- The proposed models employ the summary smoothing technique to make generated summaries more clear, cohesive, comprehensive, error-free, and less redundant.
- We also take into consideration the topic diversity in candidate sentence selection.
- We also gauge the efficacy of generated summaries by contemplating the three summary evaluation metrics, i.e., gist diversity, retention ratio, and Recall-Oriented Understudy for Gisting Evaluation (ROUGE) score.
- The experiments of the proposed model and its variants are accomplished on two datasets (Hindi and English) and their outcomes are investigated through the three summary evaluation metrics. The results show that proposed models generate compressed, lucid, and coherent summaries of input documents.
- In performance emulation, we compare our model with the baselines approaches and identify that our models have shown significant results.
- It is a study of the effect on the selection of sentences quality by assigning different weights to sentences using different schemes.

1.2 Organization

The rest of the paper is organized as follows. Section 2 discusses the related work. We present the formal definition of the problem in section 3. In section 4, we propose a linguistic feature-rich topic model-based text summarizer and its four variants. Section 5, depicts the experimental results and comparison with baseline techniques. In the end, section 6 concludes the proposed work.

2 Related work

To date, various text summarization methods have been proposed under extractive and abstractive approaches. In this paper, we give attention to the extractive text summarization methods for multiple documents. In 2003, topic modeling came into existence and used to divide the document into topics-words clusters and enables researchers to understand the statistical relationships among topics [5]. The application of these extracted topic-words clusters is in extractive text summarization. Here, we are more focused on topic modeling based methods for novel document summarization. We review the text summarization work based on the cardinality of the documents, that is, single document and multi-document summarization.

2.1 Single document summarization

A single document can be short (e.g., short stories, reviews, and tweets), average length (e.g., news articles, editorials), and lengthy documents (e.g., books, novels, and long stories). Lui et al. [15] proposed a text summarizer for a single document, which covers two features, sentence ranking based on its relevance and selection of relevant sentences using latent semantic analysis. Nomoto and Matsumoto proposed an information-centric approach in

2003 [37]. In 2007, Patel et.al. proposed a statistical extractive text summarization approach for multilingual generic text documents for newspaper articles of normal, short, brief, and very brief sizes. In their work, they experimented on English, Hindi, Gujarati, and Urdu documents [41]. In this approach, diversity rich summary is the main concern which later measured by information retrieval tasks such as document classification and document information retrieval. In 2010, Kazantseva and Szpakowicz [24] gave an idea of short stories for summaries generation followed by evaluation of summaries has been done by the 15 judges. In [16], the author suggested a hybrid text summarization approach for Hindi and Punjabi text documents. They utilized a total of nine features for effective sentence extraction. In the end, they applied the proposed model on 30 documents with a compression ratio of 30% and measured the model's performance. In 2014, Mendoza et al. proposed a memetic algorithm in which they used guided local search for a selection of important regions [32]. In 2014, R. Mishra et al. [34] discussed several extractive summarization methods such as knowledge-rich models, statistical approaches, visualization, and machine learning methods in the biomedical domain. In 2015, Parveen and Strube introduced a graph-based extractive approach for summary generation, in which the graph-based model ranks sentences based on its importance and generate an optimized and coherent summary [40]. In 2016, Gupta and Kaur have proposed an extractive novel hybrid method for Punjabi text summarization in which they exploited statistical, linguistic, conceptual, and location-based features to compute sentence significance. The SVM (support vector machine) classifier is applied to classify sentences for summary formation. Further, the evaluation of formed summary has been done using Precision, Recall, F-score, and ROUGE-2 score metrics [17]. In 2018, Qasem and Al-Radaideh have proposed a hybrid text summarization approach for Arabic documents where they utilize various features such as domain-specific, statistical, and semantic similarity for relevant sentence extraction. These features are combined using the genetic algorithm to search for a combination of optimal sentences for an effective, systematic, and coherent summary generation [2].

2.2 Multi-document summarization

The above-discussed approaches are meant for single document-based text summarization. Here, we discuss algorithms for multi-document text summarization. In 2014, John et al. [21] proposed a population-based multi-criteria algorithm to solve the multi-objective optimization problem. Their goal was to generate a summary with maximum relevance, highly diverse, and coherent. In 2015, Litvak et al. [28] proposed a hybrid approach of evolutionary method and topic modeling to maximize the summary quality. R. Bairi et al. [3] also proposed a submodular optimization problem which depicts documents as features in a topic hierarchy. This method takes the multi-documents as input and creates a sub-module of it to form the summary with high coverage, specific in nature, more diverse, and homogeneity of topics. One other proposed optimization multi-objective Artificial Bee Colony (MOABC) algorithm solves content coverage and redundancy minimization problem in a summary generation [46].

2.3 Topic modeling based summarization

Some variants of the topic modeling approach in the field of document summarization, including Bayesian sentence based topic models (BSTM) [49] and hierarchical Bayesian topic model [52] have discussed. BSTM uses the term-document and term-sentence associations to find the hidden topics and adopts these topics in a multi-document summary generation [49]. In 2013, Sanghoon et al. [25] introduced the idea of fuzzy logic with topic modeling to generate summaries for multiple documents. In their work, topic modeling extracts the latent topics while fuzzy logic solves the problem of the vagueness of sentence weights in different documents. Na et al. [30] proposed a topic-sensitive approach that discriminates between significant and insignificant topics for genuine summary and showed that not all the extracted topics identify the text theme. In other work, ~~L.~~ Na et al. [35] mixed the topics of title and content of the document into a new topic in which the summary generation algorithm learns information entropy-based weights in an adaptive asymmetric manner. However, the hierarchical Bayesian topic model embeds the n-grams into hierarchical hidden topics to find word dependencies in the local context of a word which could identify the importance of similar sentences in multi-documents [52]. To identify significant sentences, Na et al. [36] proposed a model in which they consider topics as the collection of sentences, not words. Haung et al. [20] have extended the idea of topic modeling to multilingual text summarization. In this row, hierarchical Latent Dirichlet Allocation (hLDA) used semantic information in combination with other features for sentence scoring to generate an effective and robust summary. In 2017, a merged approach of hierarchical topic modeling and the Minimal Description Length (MDL) principle proposed in which the former used to describe topics while later generated news articles summary [29]. In [18], the authors reduced the semantic redundancy using Latent Semantic Analysis (LSA) and agglomerative hierarchical clustering followed by document dimension reduction by selecting highly weighted sentences. In 2019, Roul et al. [45] proposed an LDA model based extractive summary method to generate a coherent and diverse summary.

2.4 Novel document summarization

Recall, novels have different characteristics than short documents, Ceylan [8] have proposed an extractive summary generation method for novels and considered both its length and genre in a summary generation. In [4], the authors proposed an extractive cut-paste approach to extract significant sentences from the book for the final summary generation. One more research proposed two methods where one method aligns the varied size passages from the original document into the summary sentences while the second method generalizes the HMM alignment model for long aligned documents [4]. Later, Zongda et al. [51] proposed a topic modeling-based approach in which they extract distributed topics from chapters of Novel. Extracted topics prepare a diverse and smooth summary using the LDA model. From the above discussion, it has seen that none of the algorithms is suitable for constructing an automatic text summarizer for Hindi novels and stories. Thus, it is challenging to construct an ATS system for Hindi novel and story documents.

3 Problem formulation

3.1 Notations and abbreviations

Table 1 summarizes the notations and descriptions.

3.2 Objectives

In this section, we briefly formulate the problem for the automatic summary generation, by considering the following three metrics: *Gist Diversity*, *Retention ratio,* and *ROUGE score* (Highly qualitative). Now, we discuss the intuitions that formulate the problem for generating an automatic text summary that satisfied the above-mentioned metrics.

Intuition 1: *Dimensionality Reduction.* To reduce the complexity of the lengthy documents, we first reduce the dimension of the dataset. Generally, the POS tagger marks a word in text based on its relationship with neighbor words in the given document and extracts only the most significant words with useful tags to reduce the document dimension. For our current work, we also suggested a POS tagger known as '*TnT_tagger*' to reduce the dimensionality of the Hindi dataset, as given in *Algorithm 2*.

Intuition 2: *Tagged-Topic Diffusion* (TTD). In TTD, only selected tagged terms are passed to the LDA model followed by topic-word distribution in a document that can be presented with distinct topics, as shown in Eq. (1).

$$T(D) = \left[T_k\left\{P\left(\frac{W_j}{D}\right)\right\}\right] \tag{1}$$

where, T_i denotes the k^{th} topic in document D, $W_{k,j}$ denotes j^{th} word in the k^{th} topic, for $k = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. The likelihood of occurrence of a word W in a topic T_k is indicated by$P\left(\frac{W_j}{D}\right)$.

Intuition 3: *Gist Diversity* (GD). The summary generated for the literary novel document should cover miscellaneous topics from the original document. Thus, the inclusive sentences should be rich in topics coverage. It is an incremental approach defined by using Eq. (2).

Table 1 Notations and Illustrations

Notations	Illustrations
$\boldsymbol{D}$	Document
$\boldsymbol{S_j}$	j^{th} sentence, where $S_j \in D$
$\boldsymbol{G_i}$	The gist in the i^{th} iteration.
$\boldsymbol{G_f}$	Final gist
$\boldsymbol{s}$	Sentence s, where $s \in G$
$\boldsymbol{T(D)}$	Topics in document D
$\boldsymbol{T(G)}$	Topics in gist G
$\boldsymbol{GD(G)}$	Gist diversity, which varies from 0 to 1.
$\boldsymbol{Div(S_j)}$	Topic Diversity of a sentence S_j
$\boldsymbol{CompR(G)}$	The compression ratio of a gist, which varies from 0 to 1.
$\boldsymbol{RR(S_j)}$	Redundancy rate of a sentence S_j varies from 0 to 1.
$\boldsymbol{RetenR(G)}$	Retention ratio of a gist G, which varies from 0 to1.
$\boldsymbol{S_{WT_j}}$	Weight of sentence S_j.
$\boldsymbol{TotalS_{WT}}$	Sum of the weight of all sentences in document D.
$\boldsymbol{Window_{WT_i}}$	Weight of words in the i^{th} window.
$\boldsymbol{WT_{LDA-RSW}}$	Relative sentence weight.
$\boldsymbol{WT_{LDA-ISW}}$	Integrated sentence weight.
$\boldsymbol{WT_{LDA-SWSW}}$	Sliding sentence window weight.
$\boldsymbol{WT_{LEX-LDA-SW}}$	Lexical feature-based sentence weight.
$\boldsymbol{G_{Summ\_Smooth}}$	Summary after smoothing operation.

$$GD(G) = \sum_{j=1}^{j=n} Div\left(S_j\right) \tag{2}$$

where $S_j \in G$, D, n = Number of sentences in generated gist.

Intuition 4: *Compression Ratio*. The Gist $G = \{s_1, s_2, .. s_n\}$ for document D is the subset of document D. It is represented as $G \subseteq D$. Thus, the compression ratio is defined by using Eq. (3).

$$CompR(G) = \frac{\sum_{s_g \in G} (g)}{\sum_{s_d \in D} (d)} \tag{3}$$

Intuition 5: *Redundancy Rate*. To maintain the high diversity of the topics in a gist for a given compression ratio, gist should be as least redundant as possible. For this purpose, we should remove duplicate sentences and the least relevant sentences. It is expressed by the following mathematical Eq. (4).

$$RR\left(S_j\right) = 1 - \frac{SentWT_j}{T(G_i) \cap T\left(S_j\right)} \tag{4}$$

where $S_j \in D$, $SentWT_j$=weight of j^{th} sentence.

Intuition 6: *Retention Ratio*. To ensure high quality of gist G concerning original document D. Retention Ratio can be defined as the number of common overlapping topic-words present between gist G and document D. It is defined using Eq. (5).

$$RetenR(G) = \frac{T(G) \cap T(D)}{T(D)} \tag{5}$$

Intuition 7: *Summary Generation*. The automatic extractive summary produced for input document D can be explained as a set of sentences $G = \{s_1, s_2, \ldots, s_n\}$, which are extracted from input document D. The automatic generated summary should satisfy the following criteria:

- High Compression Ratio, i.e., min$fun(len(G_n)) \leq CompR(G)$
- Low Redundancy Rate, i.e., max $fun(similarity(S_j, s))$,where $S_j \in D$ *and* $s \in G$
- Maximum Gist Diversity, i.e., max $fun(GD(G)) = T(G)$, where $0 \leq GD(G) \leq 1$
- High Gist Quality, i.e., max fun($T(G) \cap T(D)) = RetenR(G)$,where $0 \leq RetenR(G) \leq 1$

The relation among the criteria discussed above can be represented by Eq. (6).

$$CompR(G) \propto \alpha \frac{1}{RetenR(G)} = \beta \frac{1}{GD(G)} = \gamma \frac{1}{RR(G)} \tag{6}$$

We can break and rewrite the relations among different criteria, given in Eqs. (7)–(9).

$$fun(G) = \alpha\ CompR(G) + (1-\alpha) \frac{1}{RetenR(G)} \tag{7}$$

$$fun(G) = \beta\ CompR(G) + (1-\beta) \frac{1}{GD(G)} \tag{8}$$

$$fun(G) = \gamma\ CompR(G) + (1-\gamma) \frac{1}{RR(G)} \tag{9}$$

where, α, β, $\gamma \in [0, 1]$ are the parameters used to make a balance between $\{CompR, RR\}$, $\{CompR, GD\}$, and $\{CompR, RetenR\}$ respectively. The importance of different criteria is decided by the values of the parameters in such a way that greater the values of α, β, and γ more is the importance of compression ratio otherwise more important is the retention ratio, gist diversity, and redundancy rate respectively. Hence, we aim to select important and relevant sentences wisely and consciously so that the generated gist can assure the Eq. (7)–(9).

3.3 Proposed models

In this subsection, we proposed four topic model-based summary generation models based on the different sentence weighting schemes, which satisfied the objectives, defined in section 3.2. Figure 1 shows the flow graph of the proposed model. They defined as follows:

1. *Lexical LDA (LEX-LDA)*: As topic modeling is an unsupervised approach, so, topic extracted in multiple experiments may not always be the same. We assume that it is not a good idea to rely on topics extracted in a single experiment only. Hence, a certain number of experiments are performed repeatedly to fetch all possible topics discussed in the document. Here, the proposed LEX-LDA model generates a text summary based on topics that have been fetched in multiple iterations.
2. *Sliding window-based sentence weighting LDA (LDA-SWSW)*: It is a variant of LEX-LDA, wherein the sentence has assigned weights deploying the idea of Sliding Window protocol.
3. *Relative sentence weighting LDA (LDA-RSW)*: This method extracts the topics in a single run and assigns a relative weight to each sentence concerning the weights of all sentences.
4. *Integrated sentence weighting LDA (LDA-ISW)*: It is also a modification of LEX-LDA in which hidden extracted topics are integrated from the number of experiments. These integrated topics are used to assign weights to sentences.

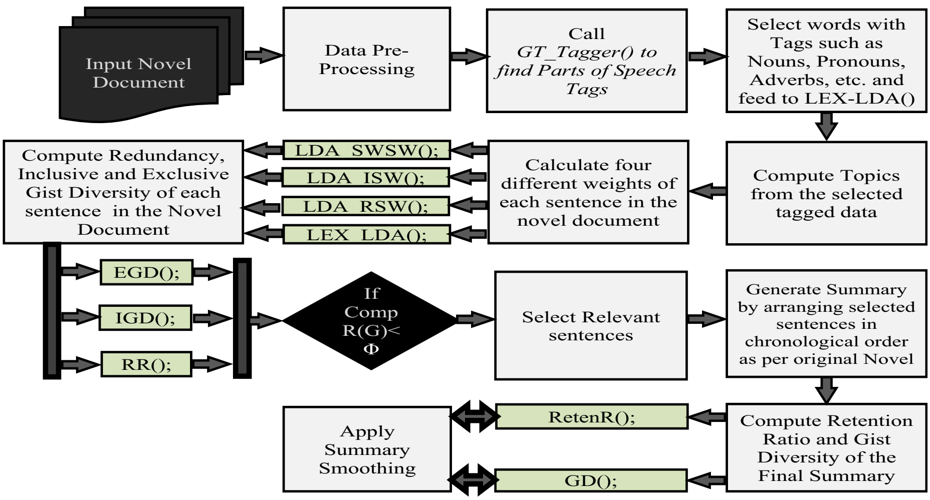


Fig. 1 Flow architecture of the proposed model, **a** steps of parts of speech tagging with example, **b** Sample input document, **c**. Different gists G_1, G_2 and G_3 generated from the input document

4 Proposed topic model-based text summarizer

In this section, we describe the proposed text summarizer, which consists of the following seven steps: data preprocessing, parts of speech tagging, linguistic feature-based topic modeling, sentence weighting, sentence selection, summary generation, and summary smoothing. We implement four variants of the proposed model in current work. Each variant of the proposed model follows all the steps except the sentence weighting phase, where they are differentiated by choosing a sentence weighting technique. The complete process is defined as follows (given in *Algorithm 1*).

Algorithm 1: Novel Gist Generator

Input: *Novel Document D*
Output: *Novel Gist G.*

1) ***begin***
2) Preprocess the novel(or story) document D using the following processes as described in section 3.1.
- *Chapter Segmentation*
- *Sentence Segmentation*
- *Tokenization*
- *Punctuation Removal*
- *Stopword Removal*
- *Stemming*

3) Call GT_Tagger() as described in Algorithm 2. *#finds tagged words.*
4) Call LEX_LDA () as explained in section 3.3 *#perceive latent topics from document D.*
5) Repeat step 5 for each sentence $S_j \in D$ *#described in section 3.4.*
- *Call LDA-SWSW ()* *#find the weight of the sentence* S_j *using a sliding window.*
- *Call LDA-RSW ()* *#find relative sentence weighting sentence* S_j.
- *Call LDA-ISW ()* *#find integrated weight of sentence* S_j.
- *Call LEX- LDA ()* *#finds weight of the lexical weight of the sentence* S_j.

 # Select contestant sentences s *based on its fitness by measuring RR, IGD, and EGD.*

6) If $G_i \neq \emptyset$ *then repeat step 7 to 10 for each sentence* $S_j \in D$
7) Call RR (S_j) *#find the redundancy rate of the sentence* S_j.
8) Call IGD (S_j) *#calculate inclusive topic diversity of sentence* S_j.
9) Call EGD (S_j) *#calculate exclusive topic diversity of sentence* S_j.
10) Add highly diverse and important sentences (low RR, high IGD, low EGD) to $G_i = G_i \cup S_j$
11) While $CompR(G) < \varphi$ *Repeat step 12* *#described in section 3.6:*
12) Select important and relevant sentences based on its weight to form final summary G_f.
13) While $i < \varphi$ *Repeat step 14:*
14) Set the sentences in G_f *into chronological order as described in section 3.6.*
15) Call RetenR(G_f*) as described in section 2.* *#calculates the retention ratio of the gist.*
16) Call GD(G_f*) as described in section 2.* *#calculates the gist diversity of the gist.*
17) Repeat step 18 and step 5 to 13 for $i = 1$ *to* n *#described in section 3.7.*
18) Call G_{Summ_Smooth} *()* *#make the final summary more smooth in nature.*
19) GoTo step 17.
20) GoTo step 18.
21) ***END***

4.1 Data preprocessing

In gist generation, before applying linguistic feature-based topic modeling, it requires to preprocess the input text document to make document noise-free and clean. It includes the following steps:

i) **Chapter Segmentation:** In general, a novel (or a story) is a collection of chapters divided by the author. Every chapter has its important topics and a combination of important topics from all the chapters are extracted to create a diverse summary.

ii) **Sentence Segmentation**: In this step, we use the NLTK sentence segmentation tool to segment each chapter into sentences based on sentence end markers such as?, |,.,!,;. After segmentation, each chapter is represented as a collection of sentences (S).

iii) **Word Tokenization**: Once sentence segmentation is done, sentence $s \in S$ is broken into words. NLTK word tokenization uses the space between two words to segment a sentence into words.

iv) **Punctuation Marks Removal**: Punctuation Marks are the conventional signs used to understand correct reading and writing of the text. Punctuation marks in Hindi are '?', '|', '!', ';', ", ',', '-','_',"" "", ':-', etc. These signs are removed from the text to make computation easy.

v) **Stop words Removal**: Stop words are the words which don't have their independent significance and help in sentence formation such as 'ke ( )', 'ka ( )', 'ne ( )', etc. We use the methodology given in [43, 44] to remove stop words from document D.

vi) **Stemming**: A word can have different forms in the singular, plural, tenses, and different parts of speech tags. For instance, going, goes, gone, and went all words have the base form 'go'. Stemming is the process to transform each word into its base form [47]. Here, we use the NLTK stemmer for stemming purposes.

4.2 Parts of speech tagging

Dimension reduction of a large document makes the processing fast and efficient. To reduce the document dimension, we first select the linguistically important features such as Nouns, Pronouns, Adverbs, Adjectives, and Verbs. However, it is a challenge to extract linguistically important terms from the document. For English text documents, we apply the NLTK POS tagger while the TnT tagger available in the NLTK library is utilized for Hindi text documents. However, it was not suitable in terms of accuracy and effectiveness for extracting linguistically important terms. Hence, we proposed a parts of speech (PoS) tagger, especially for Hindi text known as 'TnT_Tagger' as defined in *Algorithm 2*. *Algorithm 2* ensures that if the tag for word identified by TnT_tagger is unknown, Google Translator is used to translating that word into the English language. Further, the NLTK POS tagger finds the tag of the corresponding translated word and associates this new tag to the word in the source language in place of the unknown tag. Figure 2 discusses the step during tagging with an example based on *Algorithm 2*.

Algorithm 2: GT_Tagger
Input: *Sentence S* ***Output:*** *Tagged Sentence S'*
1) ***Begin*** *2) Sentence Segmentation* *3) Tokenization* *4) Train TnT Tagger* *5) Find Tags of the given input corpus.* *6) IF TAG == UNK (Unknown)* – *Call googletrans #a python library Function that translates the word into English.* – *Call NLTK POS Tagger for the translated word to find the tag.* *7) Select words tagged as Noun, Pronoun, Adverb, Adjectives, and Verbs as output to feed to the LDA model.* *8)* ***END***

4.3 Linguistic feature-based topic modeling

Here, we propose a linguistic feature-based LEX_LDA method using topic modeling. It takes the set of important related tagged (noun, pronoun, verb, etc.) words from the chapters in the input document D and gives the clusters of topic words (features). Consider, we have n documents, where D_i denotes the i^{th} the document, for $1 \leq i \leq l$, W_j denotes the j^{th} word, for $1 \leq j \leq m$ and T_k denotes the k^{th} topic, for $1 \leq k \leq n$. Table 2 gives the document-tagged_word matrix, in which integer in matrix denotes the occurrence of j^{th} word in i^{th} document.

To compute latent topics of the input document, we consider the latent Dirichlet allocation (LDA) model due to its simplicity, efficiency, and overall good performance. The main idea of LDA is to improve topic-tagged_word distribution. The LDA model takes the document-tagged_word matrix, given in Table 2 as input and gives the document-topics matrix, given in Table 3 and the topic-tagged_word matrix, given in Table 4 as output. The relation between the three matrices can be defined in Eq. (10).

$$P\left(\frac{W_j}{D_i}\right) = \sum P\left(\frac{W_j}{T_k}\right) \times P\left(\frac{T_k}{D_i}\right) \tag{10}$$

where, $P\left(\frac{W_j}{D_i}\right)$ is the probability of occurrence of a tagged word W_j in a document D_i, $P\left(\frac{W_j}{T_k}\right)$ measures the importance of W_j in the topic T_k, which is the probability of occurrence of the tagged word W_j in topic T_k and $P\left(\frac{T_k}{D_i}\right)$ measures the importance of the topic T_k, in document D_i, which is the probability of the presence of a topic T_k in the given document D_i to measure T_k.

4.4 Sentence weighting

Recall, our proposed model differs in assigning a weight to sentences. Based on assigning a weight to sentences, we have implemented the following four techniques:

1. ***Lexical Feature-based sentence weighting (LEX-LDA): In LEX-LDA***, sentence weight is the ratio of the number of topic words present in the sentence, and the total length of the sentence, is given in Eq. (11)

$$WT_{LEX-LDA-SW} = \frac{Number\ of\ topic\ words\ present\ in\ the\ sentence\ S_j}{Length\ of\ Sentence\ S_j} \tag{11}$$

2. ***LDA-Sliding Window-based sentence weighting (LDA-SWSW)***: Inspired from Luhn [31], we use a sliding window-based sentence weighting scheme, where the size of the window is the user-defined number of words. Suppose we have sentence S: ‘ ’ and the sliding window for sentence S is 8 words, i.e., “ , , , , , , , ”. The weight of i^{th} slide of sliding window $Window_{WT_i}$ can be measured by using Eq. (12)

Table 2 Document-Tagged_Word Matrix

D\W	W_1	W_2	W_3	W_m
D_1	1	0	2	1
D_2	0	1	1	4
D_3	2	1	0	1
D_l	2	3	1	0

$$Window_{WT_i} = \frac{Number\ of\ feature\ words\ covered\ in\ one\ window}{(size\ of\ winodw)} \tag{12}$$

Thus, the weight of each window in 7 windows will be 3/8, 4/8, 5/8, 4/8, 3/8, 4/8, and 3/8. From all weights, window with maximum weight will be chosen as the final sentence weight $S_{WT,}$ defined by Eq. (13).

$$WT_{LDA-SWSW} = Max\,fun(Window_{WT_1}, Window_{WT_2} \ldots.., Window_{WT_n}) \tag{13}$$

Hence, the sentence weight is the maximum of [3/8, 4/8, 5/8, 4/8, 3/8, 4/8, 3/8] i.e., 5/8.

3. ***LDA-Relative Sentence Weighting (LDA-RSW)***: It is defined as the weight of a sentence, S_{WT_j} defined in Eq. (15) w.r.t. to the total weight of all present sentences in the input document, given in Eq. (14)

$$WT_{LDA-RSW} = \frac{S_{WT_j}}{\sum_{j=1}^{j=n} S_{WT_j}} \tag{14}$$

Where,

$$S_{WT_j} = \frac{\sum(topic\text{–}word\ Frequency(S_j)) \times (Weight\ of\ each\ W\ in\ topic(P(W_j)))}{Length\ of\ Sentence\ S_j} \tag{15}$$

4. ***LDA-Integrated sentence weight (LDA-ISW)***: This method integrates the topic-words TW^i extracted using topic modeling function that is executed 'n' number of times, defined in Eq. (16)

Table 3 Document-Topic Matrix

D/T	T_1	T_2	T_3	T_n
D_1	1	0	0	1
D_2	0	1	1	0
D_3	1	1	0	1
D_l	1	0	1	0

Table 4 Topic-Tagged_Word Matrix

T/W	W_1	W_2	W_3	W_m
T_1	1	0	1	1
T_2	0	1	1	0
T_3	1	1	0	1
T_n	0	1	0	1

$$TW = \overset{i=n}{\underset{i=1}{\cup}} TW^i \tag{16}$$

We then utilize TW in topic-words frequency in Eq. (15) to find sentence weight and Eq. (14) to find the $WT_{LDA-ISW}$.

4.5 Sentence selection

After assigning a weight to each sentence in document D, in this section, we discuss the strategy of selection of a subset of candidate sentences to generate gist G, where $G \subseteq D$. The prepared gist G should have high topic diversity within the given compression ratio. To select a sentence from the document, we use Eq. (8). Due to the exhaustive search nature of Eq. (8), it takes more time for large document sizes. Therefore, we convert this exhaustive search problem into a single objective optimization problem using a heuristic search approach, defined in Eq. (17).

$$maxfun(G) = GD\,(G), \text{and } CompR(G) > \varphi \tag{17}$$

where φ is the user-defined compression ratio. It is well-known for the candidate sentence of gist, the fitness of each sentence should be high, which relies on topic diversity, retention ratio, and redundancy rate of the sentences. A good summary is the collection of most fit sentences. Hence, we have seen observations on different metrics such as compression rate, diversity, retention ratio, and redundancy to choose a set of most fit sentences.

> ***Observation 1.*** *(Inclusive Sentence Diversity (ISD)).* The final gist should cover as maximum as possible topics from the parent document. Thus, sentences with a variety of topics should be included in the gist. To understand this concept, we have considered the following intuitions:

Intuition 8: (High Topic Diverse Sentence): To accrue the gist diversity, the sentences with more topics should be included in the gist. This can be explained using two sentences S_1 and S_2 in document D such that S_1 covers the number of high probability topic words than the sentence S_2, and so it is kept in the gist, defined in Eq. (18).

Table 5 Topic-Terms Correlation in sample input document D

Topic	Terms: Probability		
1	Health:0.168	Sugar:0.083	Bad: 0.072
2	consume:0.061	Drive:0.050	Sister:0.050
3	Pressure: 0.049	Father:0.049	Sister:0.050

Table 6 Dataset Characteristics

Nature of Dataset	Hindi-Dataset Munshi Prem Chand Novels & Stories	English-Dataset Translated Novels & Stories
#of Novel Documents	114 (14 Novels and 100 stories)	114 (14 Novels and 100 stories)
#of sentences	599,647	599,647
# of Words	4,825,983	4,826,234
Total Size of Novels	64 MB	64 MB
Average # of words per document	29,200	29,278

$$maxfun(WT(S_1)) = \max\left\{\left(P(TW_i)|TW_i{\in}T, TW_i{\in}S_1\right\}\right. \quad (18)$$

For example, let us consider document D stated in Fig. 3, which includes three topics, as described in Table 5. Let us suppose a sentence from document D as " My sister likes to have sugar, but not my father", where "sister", "sugar" and "father" are considered the three topic words with probabilities are 2*(0.050 + 0.050) = 0.20, 1*(0.083) = 0.083, and 1*(0.049) = 0.049. Thus, the topic diversity of the sentence is (0.20 + 0.083 + 0.049) = 0.332, which is highly diverse.

> ***Observation 2.*** (Low Topic Diverse Sentence). A sentence that is less diverse in topics coverage, such sentence should assign less weight and is excluded from being the candidate sentence of the summary.

For example, let us inspect the sentence, "Sugar is bad to consume", where "sugar" and "consume" are the topic terms with probabilities 1*(0.083) = 0.083 and 1*(0.061) =0.061 evaluated from Eq. (19) and (20). Hence, topic diversity of given sentence is (0.083 + 0.061) = 0.144. The sentence has low diversity than the sentence explained in intuition 1. Hence, sentence 1 can be contemplated as the candidate sentence for summary genesis.

Intuition 9: (Low Topic Diverse Sentence).The diversity of a sentence $Div(S_j)$ is determined based on the prominence of each topic-word in the sentence S_j using Eq. (20) while the prominence of the topic word TW_i in all topics T_n is the sum of weight (probability) of the word TW_i in individualistic topics, given in Eq. (19).

Table 7 Performance of average gist (summary) quality for compression ratio at 10%, 20%, and 30% respectively with proposed algorithms

Average Gist Quality	CR = 10%			CR = 20%			CR = 30%		
Models || # Topics	***20***	***40***	***60***	***20***	***40***	***60***	***20***	***40***	***60***
LDA	0.12	0.06	0.08	0.06	0.04	0.03	0.04	0.04	0.03
LDA − 1	0.34	0.36	0.45	0.2	0.34	0.48	0.38	0.39	0.41
LEX-LDA	0.90	0.43	0.55	0.39	0.39	0.57	0.45	0.43	0.53
LDA-SWSW	0.48	0.47	0.64	0.30	0.49	0.60	0.44	0.52	0.57
LDA- ISW	0.76	0.37	0.48	0.40	0.38	0.44	0.43	0.43	0.44
LDA- RSW	0.81	0.39	0.48	0.41	0.37	0.48	0.46	0.44	0.48

Input: वहीं पूवक्मुख्यमंत्री दिगंबर कामत ने भाजपा में जाने क अटकलों पर दवरामलगा दयिा।
Output: [('वहीं', 'NLOC'), ('पूवष़ 'JJ'), ('मुख्यमंत्री', 'NNC'), ('दिगंबर', 'NN'), ('कामत', 'NNP'), ('ने', 'PREP'), ('भाजपा', 'NNP'), ('में', 'PREP'), ('जाने', 'VNN'), ('क ', 'PREP'), ('अटकलों', 'NN'), ('पर', 'PREP'), ('दवराम़ 'NN'), ('लगा', 'VFM'), ('दयिा', 'NN'), ('।', 'SYM')]

Step 1: Tokenize the sentence
['वहीं', 'पूवव, 'मुख्यमंत्री', 'दिगंबर', 'कामत', 'ने', 'भाजपा', 'में', 'जाने', 'क ', 'अटकलों', 'पर', 'दवराम, 'लगा', 'दयिा।']
Step 2: Apply TnT Tagger on tokenized text.
[('वहीं', 'NLOC'), ('पूवष़ 'JJ'), ('मुख्यमंत्री', 'NNC'), ('दिगंबर', 'Unk'), ('कामत', 'Unk'), ('ने', 'PREP'), ('भाजपा', 'NNP'), ('में', 'PREP'), ('जाने', 'VNN'), ('क ', 'PREP'), ('अटकलों', 'Unk'), ('पर', 'PREP'), ('दवराम़ 'NN'), ('लगा', 'VFM'), ('दयिा', 'Unk'), ('।', 'SYM')]
Step 3: For UNK tags, find the English translation of the respective word. For example,
'दिगंबर', 'Unk' = Digambar
'दयिा', 'Unk' = Gave
Step 4: Apply again the NLTK tagger on the translated word. E.g.
'दिगंबर', 'NN'
'दयिा', 'NN'
Step 5: Reform the sentence again
[('वहीं', 'NLOC'), ('पूवष़ 'JJ'), ('मुख्यमंत्री', 'NNC'), ('दिगंबर', 'NN'), ('कामत', 'NNP'), ('ने', 'PREP'), ('भाजपा', 'NNP'), ('में', 'PREP'), ('जाने', 'VNN'), ('क ', 'PREP'), ('अटकलों', 'NN'), ('पर', 'PREP'), ('दवराम़ 'NN'), ('लगा', 'VFM'), ('दयिा', 'NN'), ('।', 'SYM')]

Fig. 2 steps of parts of speech tagging with example

$$\mathrm{P}(\mathrm{TW_i}) = \mathrm{TW_{iCount}} \times \sum \mathrm{P}\left(\frac{\mathrm{TW_i}}{\mathrm{T_n}}\right) \tag{19}$$

$$Div(S_j) = \sum P(TW_i) \tag{20}$$

where, $\mathrm{TW_{iCount}}$ and $P\left(\frac{W_i}{}T_j\right)$ denote the distribution of the word TW_i in the extracted topics and probability of an i^{th} word in j^{th} topic respectively.

> ***Observation 3.*** *(Excluded Sentence Diversity (ESD)).* For a given sentence $S_j \in D$, if S_j has high similitude with the already included sentences in gist then the sentence should not be included in the gist.

Intuition 10: If the sentence $similarity(S, S_j) = 1$, where $S \in G_i$, then sentence S_j will not be included in G_i.

Intuition 11: If sentence S_j is rich in topics but all of its topics have been covered, then S_j should not be included in G_i.

Table 8 Summary evaluation using average retention ratio for 10%, 20% and 30% compression ratio

Average retention ratio	CR = 10%			CR = 20%			CR = 30%		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.53	0.31	0.22	0.34	0.31	0.22	0.56	0.30	0.21
LDA − 1	0.54	0.52	0.53	0.63	0.60	0.58	0.67	0.60	0.58
LEX-LDA	0.59	0.55	0.55	0.66	0.67	0.65	0.77	0.67	0.65
LDA-SWSW	0.64	0.59	0.59	0.68	0.66	0.64	0.80	0.65	0.62
LDA- ISW	0.59	0.55	0.55	0.65	0.61	0.64	0.76	0.62	0.61
LDA- RSW	0.60	0.54	0.56	0.66	0.62	0.59	0.75	0.61	0.60

Sugar is bad to consume. My sister likes to have sugar, but not my father. My father spends a lot of time driving my sister around to dance practice. Doctors suggest that driving may cause increased stress and blood pressure. Sometimes, I feel pressure to perform well at school, but my father never seems to drive my sister to do better. Health experts say that Sugar is not good for your lifestyle.

Fig. 3 Different gists G_1, G_2 and G_3 generated from the input document

> ***Observation 4.*** *(Redundant Information Removal).* The importance of a sentence is influenced by two parameters: the manifestation of topic words and stop words in the sentence discretely. The appearance of less number of stop words and the high number of topic words in gist G leads to high weight. We revise Eq. (9) to Eq. (21) to elucidate the required redundancy rate.

Intuition 12. For a given sentence $S \in G$ and $S_j \in D$, if the number of topics in both S_j and S are same but the redundancy rate is such that, $S_j < S$, then S_j is added in the final gist G, otherwise, terminate the operation. Consider a sentence "*Sometimes, I feel the pressure to perform well at school, but my father never seems to drive my sister to do better*." To compute the redundancy of the given sentence, we reckon the sentence weight i.e., 0.049. We suppose the number of topic terms present in some gist G is 2. Using Eq. (4), we have RR of the given sentence is 0.9 which is enumerated as highly redundant and hence cannot be included in the final summary.

$$minfun(S_j) = \left\{ RR\ (S_j) \middle| \ S \in G, S_j \in D \right\} s.t. G \subseteq D \text{ and } CompR(G) > \varphi \tag{21}$$

4.6 Summary generation

In this step, we construct the final gist by rearranging the selected sentences in chronological order.

> ***Observation 5.*** *(Highly Retention Gist).* We know that $G \subseteq D$, such that compressed gist G should cover all attainable topics of D or we can say that the outcome gist should be thoroughly newsy. High retention gist refers to the intense sentence quality and is defined as the number of common topic-words present in both summaries as well as in document D.

Table 9 Rouge Score-1 values with the change in the number of topics as 20, 40, and 60 respectively with the compression rate at 10%

Compression Ratio = 10%	Rouge Score-1 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.30	0.28	0.29	0.31	0.35	0.38	0.45	0.41	0.42
LDA − 1	0.47	0.37	0.46	0.40	0.41	0.38	0.50	0.45	0.41
LEX-LDA	0.61	0.52	0.49	0.43	0.42	0.47	0.53	0.51	0.48
LDA-SWSW	0.69	0.66	0.59	0.44	0.44	0.49	0.51	0.50	0.47
LDA- ISW	0.67	0.54	0.51	0.65	0.66	0.63	0.55	0.47	0.46
LDA- RSW	0.64	0.57	0.53	0.43	0.44	0.48	0.52	0.50	0.51

Table 10 Rouge Score-1 behavior with the change in the number of topics as 20, 40, and 60 respectively for given compression rate at 20%

Compression Ratio = 20%	Rouge- Score-1 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.32	0.35	0.33	0.34	0.30	0.31	0.42	0.41	0.44
LDA − 1	0.53	0.50	0.47	0.35	0.31	0.34	0.44	0.42	0.40
LEX-LDA	0.63	0.56	0.53	o.53	0.35	0.38	0.48	0.46	0.48
LDA-SWSW	0.70	0.63	0.60	0.38	0.36	0.39	0.45	0.47	0.48
LDA- ISW	0.62	0.51	0.59	0.47	0.45	0.46	0.53	0.52	0.51
LDA- RSW	0.64	0.60	0.57	0.39	0.36	0.39	0.49	0.50	0.51

Intuition 13. Suppose three different gists are G_1, G_2, G_3 are prepared from input document D as given in Fig 4, where G_1 covers the two topics {"sister, father", "consume, pressure"}, G_2 also covers the two topics {"health, pressure", "bad, drive"} and G_3 covers the three topics {"health, sugar", "pressure", "sister, father"}. The retention ratio of G_1, G_2, G_3 are 4/9, 4/9, and 5/9 respectively. Thus, G_3 is considered to be more informative than G_1 and G_2. Hence, all sentences from G_3 are kept in the final summary and most informative sentences from G_1 and G_2 those have not been covered in G_3 are also added in the final summary.

$$maxfun(G) = RetenR\ (G), \text{and } CompR(G) > \varphi \tag{22}$$

The critical threshold values as φ in equation (17), (21), and (22) are defined by the user manually based on his requirement. It is decided based on the size of the summary a particular user needed. E.g., the title of the document, summary length in terms of percentage of the original document length, and limit machine-generated summary in the form of the number of words. In our proposed work, we restrict the summary length in the form of the percentage of the original document length.

Time Complexity From the *Algorithm1*, we observe that the time taken by the algorithm can be divided into three parts: i) data preprocessing (Line 2); ii) topic modeling (Lines 3–4); iii) sentence weight calculation (Line 5); iv) subset of candidate sentences selection (Lines 6–10); v) calculate summary quality. If we ignore the data preprocessing and topic modeling time, the

Table 11 Rouge Score-1 with the change in the number of topics as 20, 40, and 60 topics respectively for given compression rate at 30%

Compression Ratio = 30%	Rouge Score-1 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.32	0.36	0.37	0.30	0.36	0.33	0.42	0.45	0.46
LDA − 1	0.34	0.31	0.36	0.39	0.32	0.34	0.39	0.31	0.50
LEX-LDA	0.63	0.59	0.53	0.37	0.46	0.42	0.47	0.49	0.48
LDA-SWSW	0.70	0.65	0.62	0.32	0.38	0.37	0.44	0.47	0.47
LDA- ISW	0.57	0.55	0.52	0.47	0.51	0.56	0.52	0.53	0.54
LDA- RSW	0.64	0.61	0.56	0.39	0.42	0.44	0.49	0.51	0.55

Table 12 Change in behavior of Rouge Score-2 of summary with topics as 20, 40, and 60 respectively for 10% compression ratio

Compression Ratio = 10%	Rouge Score-2 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.02	0.18	0.02	0.07	0.06	0.05	0.05	0.04	0.04
LDA − 1	0.11	0.10	0.14	0.09	0.12	0.14	0.16	0.13	0.14
LEX-LDA	0.19	0.18	0.16	0.14	0.15	0.14	0.16	0.17	0.15
LDA-SWSW	0.29	0.25	0.16	0.20	0.19	0.14	0.23	0.22	0.15
LDA- ISW	0.13	0.11	0.18	0.12	0.15	0.16	0.20	0.16	0.11
LDA- RSW	0.21	0.20	0.17	0.14	0.16	0.14	0.17	0.18	0.16

remaining processing time depends on Line 5–10 in weight calculation of sentences and selection of the important sentences in the document. Step 5 takes *O(n.m)* time, where n is the number of sentences in the input literary novel and we assume *O(m)* is the time complexity of the called function within the loop. Similarly, in steps 6–10, the time complexity is again O(n.m), thus, the total time complexity is *O(n.m)* + *O(n.m)* = *O(n.m)*. Hence, the time complexity of Algorithm 1 is *O(n.m)*.

4.7 Summary smoothing

Usually, the final summary should be intelligible, readable, noise-free, least redundant, and error-free. To accomplish these tasks, we amalgamate the summaries G_i derived from the i^{th} iteration with the removal of duplicate sentences and insertion of unambiguous salient sentences to form an intermediate set of sentences G_f, defined in Eq. (23). Further, we afresh rank those concerning their weights; pick the high weighted sentences and rearrange in the chronological order to form the final summary.

$$G_{Summ\_Smooth} = \bigcup_{i=1}^{i=n} G_i \tag{23}$$

Table 13 Change in behavior of Rouge Score-2 of summary with topics as 20, 40, and 60 respectively for 20% compression ratio

Compression Ratio = 20%	Rouge- Score-2 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.02	0.02	0.01	0.07	0.08	0.05	0.04	0.05	0.03
LDA − 1	0.11	0.10	0.107	0.03	0.04	0.09	0.09	0.07	0.05
LEX-LDA	0.20	0.19	0.15	0.12	0.13	0.13	0.15	0.16	0.14
LDA-SWSW	0.25	0.20	0.19	0.14	0.14	0.15	0.18	0.16	0.17
LDA- ISW	0.14	0.15	0.12	0.13	0.14	0.13	0.14	0.15	0.12
LDA- RSW	0.21	0.20	0.17	0.13	0.14	0.14	0.16	0.17	0.15

Table 14 Change in behavior of Rouge Score-2 of summary with topics as 20, 40, and 60 respectively for 30% compression ratio

Compression Ratio = 30%	Rouge Score-2 for Hindi Dataset								
Summary accuracy Measures	Precision			Recall			F-Score		
Models || # Topics	**20**	**40**	**60**	**20**	**40**	**60**	**20**	**40**	**60**
LDA	0.03	0.02	0.03	0.08	0.07	0.06	0.04	0.04	0.04
LDA − 1	0.10	0.09	0.11	0.07	0.08	0.06	0.10	0.07	0.11
LEX-LDA	0.20	0.18	0.19	0.12	0.14	0.16	0.15	0.16	0.18
LDA-SWSW	0.23	0.21	0.15	0.23	0.20	0.18	0.13	0.14	0.12
LDA- ISW	0.13	0.12	0.12	0.16	0.15	0.16	0.14	0.15	0.16
LDA- RSW	0.20	0.20	0.21	0.13	0.15	0.17	0.16	0.17	0.19

5 Experimental and result analysis

In this section, we assemble the summaries on Hindi and English novels using our proposed approaches. For experiments simulation, here, we first define the characteristics of prerequisite datasets and manual summaries. We concisely elaborate on our proposed models and differentiation models to contrast them. Further, we evaluate the above summaries generated from the proposed models with reference (manual) summaries.

5.1 Dataset characteristics

It has been discerned that the corpus of text summarization for different languages is accessible, for example, Arabic, Czech English, Greek, Hebrew, Chinese, Romanian, and Spanish [11, 26]. To the best of our knowledge, no corpus and reference summaries are at hand for the Hindi language due to which, we first construct the Hindi corpus by accumulating the novels and stories written by famous Indian writer Munshi Premchand (see in [50]). Here, we have stockpiled around 114 Hindi novels including short stories from 'Munshi Premchand's stories' blog [6]. To standardize the results, we have also simulated our proposed model on the English version of 'Munshi Premchand's stories [50]. Table 6 summarizes the characteristics of the required dataset.

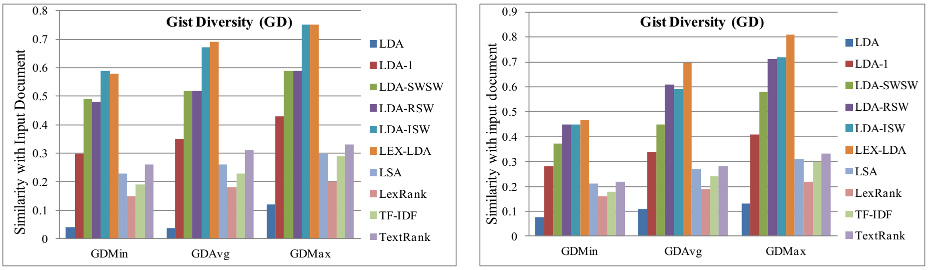
To estimate the performance of machine-generated summaries, we need reference (manual) summaries. Consequently, we collected the reference summaries from various online blogs including discussions and reviews (see in [7]). The online accrued blogs have user-shared manual summaries and their corresponding reviews on novels of well-known Hindi writer, Munshi Premchand.

Gist G_1: Sugar is bad to consume. My sister likes to have sugar, but not my father. Sometimes, I feel pressure to perform wee at school, but my father never seems to drive my sister to do better.

Gist G_2: Sugar us bad to consume. Sometimes, I feel pressure to perform wee at school, but my father never seems to drive my sister to do better. Health experts say that Sugar is not good for your lifestyle.

Gist G_3: Sometimes, I feel pressure to perform wee at school, but my father never seems to drive my sister to do better. Health experts say that Sugar is not good for your lifestyle.

Fig. 4 Different gists G_1, G_2 and G_3 generated from the input document



(a). The similarity of Gist Diversity (GD) with the manual summary on Hindi dataset using proposed models and baseline models

(b). The similarity of Gist Diversity (GD) with the manual summary on English dataset using proposed models and baseline models

Fig. 5 **a**. The similarity of Gist Diversity (GD) with the manual summary on Hindi dataset using proposed models and baseline models, **b**. The similarity of Gist Diversity (GD) with the manual summary on English dataset using proposed models and baseline models

5.2 Comparison models

Here, we briefly explore the comparison models, usually known as baselines (candidate) models.

1. *Traditional LDA*: It identifies the latent topics in a given document and uses topic words as features for sentence weighting.
2. *LDA-1*: LDA-1 is a tweak of proposed LDA that runs only for one iteration to fetch the latent topics of the given document. For make different with generic LDA models, the authors abbreviate it as LDA-1.
3. *TF-IDF*: TF-IDF is an idea to select sentences based on its weight to generate a summary. We have considered this approach as the minimum limit for summary evaluation.
4. *TextRank*: TextRank is an unsupervised and undirected graph-based extractive method that exploits the document structure to find important topics where it constructs a graph, in which vertices are text units and edges are built on the lexical similarity between two vertices. For a summary generation, it ranks each vertex using the random walk algorithm based on the eigenvector [33].
5. *Latent Semantic Analysis (LSA)*: LSA extracts the semantically important sentences employing singular vector decomposition (SVD). It mainly works to cover the wider aspect of a document with less redundancy [39].

Table 15 Paired t-test p values for Gist Diversity of proposed techniques against baseline algorithms

Gist Diversity(GD) Score		ENGLISH			HINDI			
Baselines|| Ours	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA
LDA	0.008201	0.007653	0.008617	0.001165	0.000162	0.000992	0.000531	0.000766
LDA-1	0.001974	0.012228	0.001021	0.001848	0.001454	0.001389	0.003633	0.004446
LSA	0.001291	0.009902	0.001593	0.001689	0.001347	0.006138	0.002742	0.003578
LexRank	0.001237	0.000396	0.001668	0.001233	0.00097	0.004472	0.00251	0.003149
TF-IDF	0.007091	0.006928	0.007987	0.00133	0.001014	0.003356	0.001353	0.001854
TextRank	0.001446	0.009618	0.001107	0.001696	0.002151	0.008384	0.00341	0.004417

Table 16 Paired t-test p-values for the Retention Ratio of proposed techniques against baseline algorithms

Retention Ratio (RR)		Hindi		English				
Baselines|| Ours	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA
LDA	0.001935	0.007945	0.004042	0.006606	0.004082	0.001743	0.005406	0.007819
LDA-1	0.000102	0.004926	0.002584	0.002595	0.004681	0.002607	0.00411	0.002933
LSA	0.001136	0.004566	0.001942	0.001942	0.001764	0.002218	0.001453	0.00667
LexRank	0.003814	0.005896	0.00247	0.004653	0.00443	0.001744	0.004322	0.003345
TF-IDF	0.001719	0.002883	0.000739	0.001735	0.003179	0.000806	0.00298	0.002165
TextRank	0.001715	0.00079	0.00079	0.001804	0.001795	0.000294	0.001638	0.000986

6. *LexRank*: LexRank is also a graph-based unsupervised approach that scores sentences based on graph centrality. The recommendation principle evaluates the graph centrality, in which one sentence recommends the other sentences, and the sentence with higher recommendation is ranked as important [13].

5.3 Summary evaluation

Here, we evaluate the effectiveness of generated summaries using three metrics: gist diversity, retention ratio (RetenR), and ROUGE-N score. We estimate and analyze each metric on two different parameters: the number of topics and the compression ratio. For evaluation purposes, we simulate all the proposed and baseline models on 10%, 20%, and 30% compression ratios. Moreover, we analyze the performance of the proposed model and the baselines on a different set of topics, 20, 40, and 60. Further, we evaluate and compare the performance of the baseline algorithms with our four proposed models under three metrics (for maximum, average, and minimum metric value).

Here, we investigate the gist diversity (GD), retention ratio, and ROUGE-N score for examining the performance of proposed models.

Table 17 Paired t-test p-values for ROUGE-1 F-scores of proposed techniques against baseline algorithms

ROUGE-1 Scores		English		Hindi				
Baselines|| Ours	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA
LDA	0.001187	0.001401	0.000858	0.001272	0.000403	0.001941	0.002013	0.000021
LDA-1	0.002489	0.003364	0.003492	0.009052	0.003931	0.003955	0.002033	0.003293
LSA	0.001993	0.00047	0.00025	0.000002	0.00023	0.001665	0.000003	0.002756
LexRank	0.003908	0.004226	0.00246	0.005498	0.003997	0.002757	0.000924	0.001691
TF-IDF	0.001097	0.001633	0.001776	0.00151	0.001071	0.001363	0.000131	0.001895
TextRank	0.005092	0.007768	0.000472	0.005212	0.000941	0.001816	0.002746	0.003244

Table 18 Paired t-test p-values for ROUGE-2 F-scores of proposed techniques against baseline algorithms

ROUGE-2 scores	English				Hindi			
Baselines|| Ours	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA	LDA-SWSW	LDA-RSW	LDA-ISW	LEX-LDA
LDA	0.002752	0.000531	0.002447	0.004372	0.006629	0.001161	0.001277	0.003622
LDA-1	0.00795	0.002362	0.000414	0.031531	0.004806	0.0004	0.001438	0.000444
LSA	0.003581	0.001072	0.004337	0.005441	0.00457	0.000007	0.000971	0.000007
LexRank	0.001943	0.000234	0.001013	0.003142	0.004347	0.000008	0.000966	0.000008
TF-IDF	0.008795	0.004767	0.023416	0.012911	0.003446	0.000001	0.00021	0.000001
TextRank	0.003958	0.000905	0.006081	0.006459	0.006498	0.00057	0.002012	0.000651

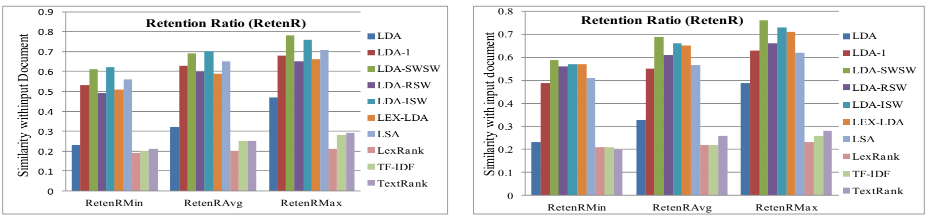
1. **Gist Diversity (GD)**: The maximum, average and minimum gist diversity for the i^{th} algorithm is defined in Eqs. (24)–(26) respectively, where 'c' represented as compression rate and 'T' constituted as the number of topics 'T' such that $c \in \{10, 20, 30\}$ and $T \in \{20, 40, 60\}$

$$GD_{Max}\left(G_f\right) = MaxDiv\left({G_i}^{c,T}\right) \tag{24}$$

$$GD_{Avg}\left(G_f\right) = \frac{\sum\limits_{G_i} Div\left({G_i}^{c,T}\right)}{\sum\limits_{G_i} Length\left({G_i}^{c}\right)}\Bigg) \tag{25}$$

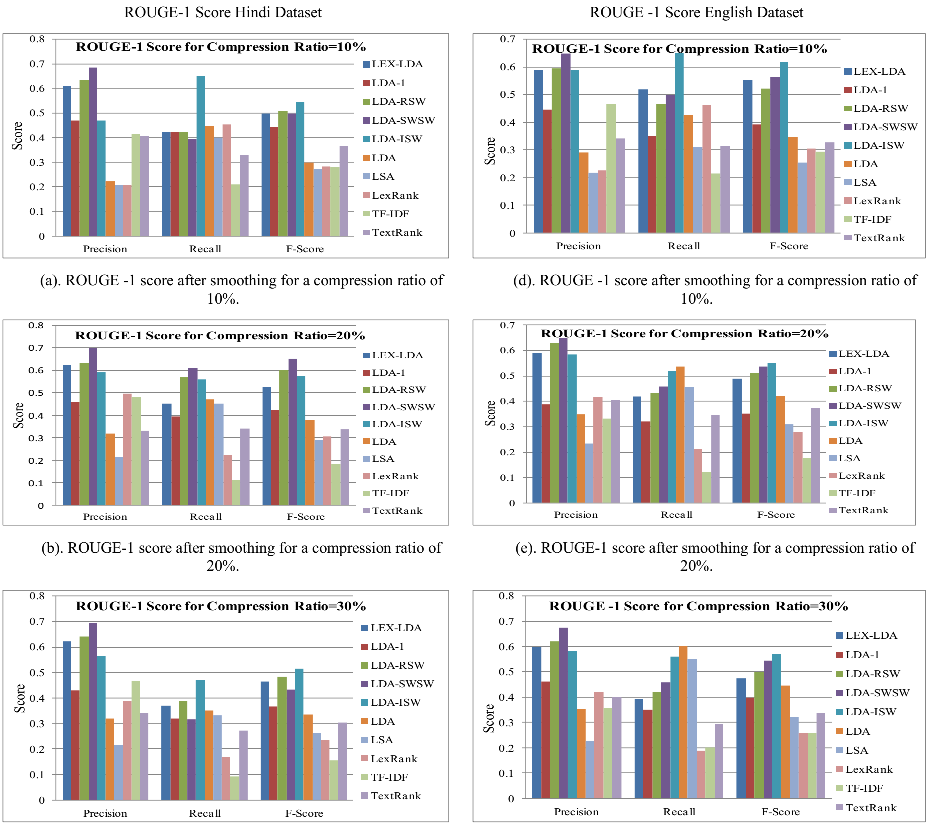
$$GD_{Min}\left(G_f\right) = MinDiv\left({G_i}^{c,T}\right) \tag{26}$$

Table 7 summarizes the difference of average gist quality score obtained from our proposed models (LEX-LDA, LDA-SWSW, LDA-RSW, and LDA-ISW) with generic LDA models, where, we presume the compressed data as 10%, 20%, and 30% and the number of topics is 20, 40 and 60 topics. It is to note that the deviation in the conduct of the proposed algorithm is due to the changes ponder in choosing the number of topics and compression ratio. Since due to the unsupervised behavior of proposed models, they don't follow a particular behavior. From Table 7, we observe that the more compressed data covered fewer topics have good gist summary value and gradually decrease while covering more topics. For instance, LEX-LDA and LDA-RSW models have 0.90 and 0.81 gist diversity scores respectively on 20 topics, 10% of the compression ratio and LEX-LDA and LDA-RSW models have 0.55 and 0.48 gist summary scores respectively on 60 topics and 10% of compressed data. It is also discerned that



(a). Retention Ratio of summary on Hindi Dataset. (b). Retention Ratio of summary on English Dataset.

Fig. 6 **a**. Retention Ratio of summary on Hindi Dataset, **b**. Retention Ratio of summary on English Dataset



(a). ROUGE -1 score after smoothing for a compression ratio of 10%.

(d). ROUGE -1 score after smoothing for a compression ratio of 10%.

(b). ROUGE-1 score after smoothing for a compression ratio of 20%.

(e). ROUGE-1 score after smoothing for a compression ratio of 20%.

(c). ROUGE-1 score after smoothing for a compression ratio of 30%.

(f). ROUGE-1 score after smoothing for a compression ratio of 30%.

Fig. 7 **a**. ROUGE −1 score after smoothing for a compression ratio of 10%, **b**. ROUGE-1 score after smoothing for a compression ratio of 20%, **c**. ROUGE-1 score after smoothing for a compression ratio of 30%, **d**. ROUGE −1 score after smoothing for a compression ratio of 10%, **e**. ROUGE-1 score after smoothing for a compression ratio of 20%, **f**. ROUGE-1 score after smoothing for a compression ratio of 30%

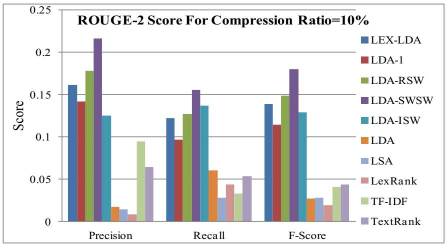
models give less gist diversity scores for less compressed data and fewer topics. Table 7 summarizes that the proposed models have a high gist score on contrasting with the generic LDA model.

Graphically, Fig. 2a and b shows the comparison of minimum, average, and maximum gist diversity of proposed approaches (LEX-LDA, LDA-SWSW, LDA-RSW, and LDA-ISW) and baseline algorithms (LDA, LDA-1, LSA, LexRank, TF-IDF, and TextRank) for Hindi dataset and English dataset respectively. Here, minimum, average, and maximum gist diversity is evaluated using Eqs. (25)–(27) respectively. From Fig. 2a and b, it is clear that the proposed models have superior gist diversity scores when comparing with the baseline models.

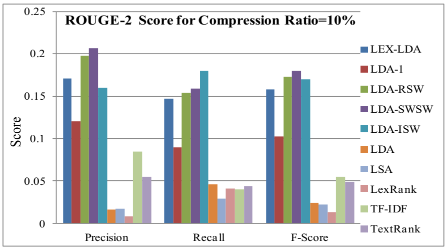
2. **Retention Ratio (RetenR):** Similar to GD, the maximum, average, and minimum retention ratio for the i^{th} algorithm is elucidated in Eqs. (27)–(29) respectively, where compression rate 'c' and the number of topics 'T' such that $c \in \{10, 20, 30\}$ and $T \in \{20, 40, 60\}$

ROUGE-2 score on Hindi dataset

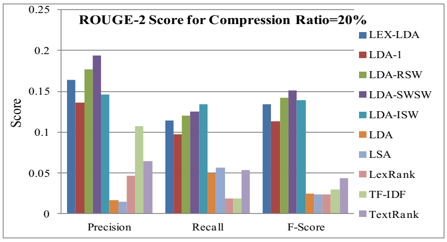
ROUGE-2 Score on English dataset



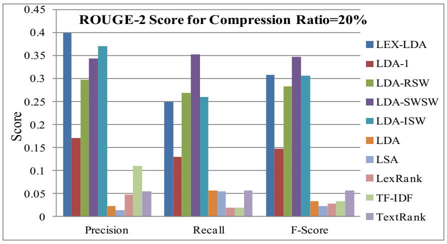
(a). ROUGE-2 score after smoothing for a compression ratio of 10%.



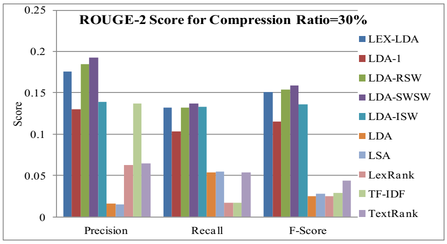
(d). ROUGE-2 score after smoothing for a compression ratio of 10%.



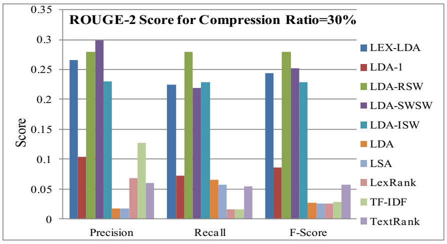
(b). ROUGE-2 score after smoothing for a compression ratio of 20%.



(e). ROUGE-2 score after smoothing fora compression ratioof20%.



(c). ROUGE-2 score after smoothing for a compression ratio of 30%.



(f). ROUGE-2 score after smoothing for a compression ratio of 30%.

Fig. 8 **a**. ROUGE-2 score after smoothing for a compression ratio of 10%, **b**. ROUGE-2 score after smoothing for a compression ratio of 20%, **c**. ROUGE-2 score after smoothing for a compression ratio of 30%, **d**. ROUGE-2 score after smoothing for a compression ratio of 10%, **e**. ROUGE-2 score after smoothing for a compression ratio of 20%, **f**. ROUGE-2 score after smoothing for a compression ratio of 30%

$$RetenR_{Max}\left(G_f\right) = Max\ RetenR\ \left({G_i}^{c,T}\right) \tag{27}$$

$$RetenR_{Avg}\left(G_f\right) = \frac{\sum_{G_i} RetenR\ \left({G_i}^{c,T}\right)}{\sum_{G_i} Length\left({G_i}^{c}\right)} \tag{28}$$

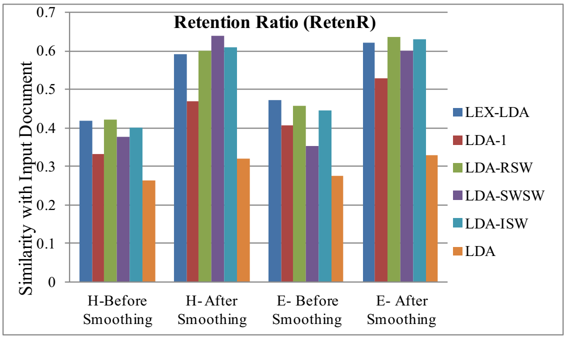


Fig. 10 Comparison of RetenR on Hindi and English dataset before and after summary smoothing

$$RetenR_{Min}\left(G_f\right) = MinRetenR\ \left({G_i}^{c,T}\right) \tag{29}$$

Table 8 compares the average retention ratio values of our proposed models (LEX-LDA, LDA-SWSW, LDA-RSW, and LDA-ISW) and generic LDA models, where, we consider the compressed data as 10%, 20%, and 30% and the number of topics is 20, 40 and 60 topics. From Table 8, we can perceive that the LDA-SWSW model under a 10% compression ratio covers the maximum topics. The deviation in the behavior of the proposed algorithm is due to the changes considered in choosing the number of topics and compression ratio. Fig. 3a and b indicates the difference of minimum, average and maximum retention ratio value of proposed approaches (LEX-LDA, LDA-SWSW, LDA-RSW, and LDA-ISW) and it's transfigurations with the baseline algorithms (LDA, LDA-1, LSA, LexRank, TF-IDF and TextRank) for Hindi dataset and English dataset respectively. Here, minimum, average, and maximum gist diversity is evaluated using Eqs. (27)–(29) respectively. From Fig. 3a and b, it has been seen clearly that proposed models have better gist diversity scores on comparing with the baselines.

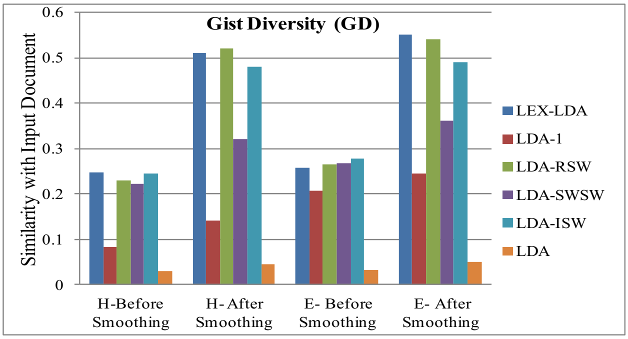
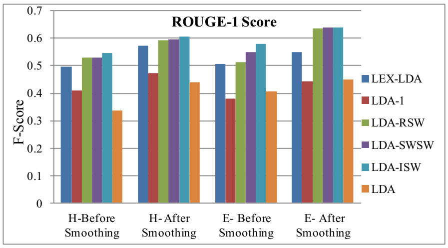
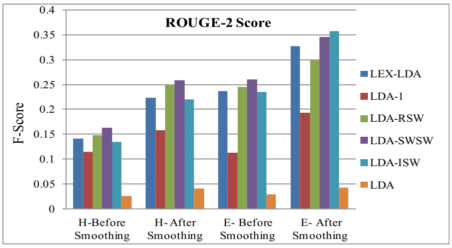


Fig. 9 Comparison of GD on Hindi and English dataset before and after summary smoothing



(a). Comparison of ROUGE-1 score on Hindi and English Dataset before and after gist smoothing.



(b). Comparison of ROUGE-2 score on Hindi and English Dataset before and after gist smoothing.

Fig. 11 **a**. Comparison of ROUGE-1 score on Hindi and English Dataset before and after gist smoothing, **b**. Comparison of ROUGE-2 score on Hindi and English Dataset before and after gist smoothing

3. **ROUGE Score:** ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation, which is considered to be the gold standard for summary evaluation [27]. It is based on the principle of overlapping words in both manual summaries as well as automated summaries. Under the ROUGE score, we have evaluated the ROUGE-N metrics, as given in Eq. (30)

$$ROUGE{-}N = \frac{\sum_{G}\left(\sum_{n} CountMatch(n)\right)}{\sum_{G}\left(\sum_{n} Count(n)\right)} \quad where, G \in Man\_Summ \tag{30}$$

Where n is depicted as the length of n-grams, Man_Summ represents the manual summary, $CountMatch(n)$ is the number of matching words between manual summary and automated summary and $Count(n)$ denotes the maximum number of n-grams in the manual summary. Tables 9, 10 and 11 show ROUGE-1 Score of summaries of proposed models under distinct topics for 10%, 20%, and 30% compression rates respectively. Tables 12, 13 and 14 show Rouge Score-2 of summaries of the proposed model for distinct topics under 10%, 20%, and 30% compression rates respectively. Tables 9, 10 and 11 show that the proposed algorithm and its sentence selection based variants beat the traditional LDA models. Figure 4 shows the ROUGE-1 score comparison of proposed models with baseline models on Hindi and English dataset. Similarly, Fig. 5 shows the comparison of the ROUGE-2 scores of proposed models with baseline models on Hindi and English dataset. We also notice that F-score values for the proposed algorithms win over baseline algorithms.

Statistical testing results We also performed statistical paired samples t-test to examine the significant difference between the proposed and baselines methods. The paired t-test is a statistical test carried out for small size samples under which observation of one sample pairs with the specific observation of the other sample in a significant way. From the above experiments, we see that the proposed model with four different sentence selection transfigurations effectuates better than the baseline algorithms both on Hindi and English datasets. To confirm whether outcomes of our proposed model is a coincidence, or it is statistically significant over the

Table 19 Sample Hindi Novel Document and summaries at a 10% compression ratio for proposed and baseline algorithms

Sample Document	नादरिशाह क सेना में दिल्ली के कत्लेआम कर रखा है। गलियों मे खून क नदियां बह रह हैं। ारि तरफ हाहाकार मा ि हुआ है। बाजार बंद है। दिल्ली के ल ग घरों के द्वार बंद दकयेजान क खैर मना रहे है। दकस क जान सलामत नहीं है। कहीं घरों में आग लग हुई है, कहीं बाजार लुट रहा है; क ई दकस क फररया ि नहीं सुनता। रईसों क बेगमें महल से दनकाल जा रह है और उनक बेहुरती क जात है। ईरान सिपाहियों क रक्त दपपासादकस तरह नहीं बुझत । मानव हृदया क क्रूरता, कठ रता और पैशादकिता अपना दवकरालतमरूप धारण दकयेहुए है। इस समया नादरि शाह ने बाशिाह महल में प्रवेश दकया।दिल्ली उन दिनों भ ग-दवलासक केंद्र बन हुई थ । सजावट और तकल्लुफ के सामानों से रईसों के भवन भरे रहते थे। स्त्रियों क बनाव-सिगांर के दसवा क ई काम न था। पुरूषों क सुख-भ ग के दसवा और क ई चिन्ता न थ । राज नदतका स्थान शेर -शायर ने ले दलयाथा। समस्त प्रन्तो से धन खिंच-खिंच कर दिल्ली आता था। और पान क भांति बहाया जाता था। वेश्याओं क चादीं थ । कहीं तीतरों के जोड़ ह ते थे, कहीं बटेर और बुलबुलों क पलियां ठनत थीं। सारा नगर दवलास–निद्रा में मग्न था। नादरिशाह शाह महल में पहुंचा त वहां का सामान ेखिकर उसक आंखें खुल गयीं। उसका जन्म दरिद्र-घर में हुआ था। उसका समसत ज वन रणभूदम में ह कटा था। भ ग दवलासका उसे सिका न लगा था। कहां रण-क्षेत्र के कष्ट और कहां यह सुख-साम्राज्य। दजधरआंख उठत थ , उधर से हटने का नाम न लेत थ । संध्या ह गय थ । नादरिशाह अपने सरदारों के साथ महल क सैर करता और अपन पसंद क चीजों क बट रता हुआ ि विाने -खास में आकर कारबि मसनि पर बैठ गया, सरदारों क वहां से लि जाने का हुक्म ेि दयिा, अपने सबहदथयाररख दयि और महल के रि गा क बुलाकर हुक्म दयिा–मै शाह बेगमों का ना ि ेखिना ाहिता हूं। तुम इस वक्त उनक सुंदर वस्त्राभूषणों से सजाकर मेरे सामने लाओं खबरारि , जरा भ ेरि न ह ! मै क ई उज्र या इनकार नहीं सुन सकता।
LEX-LDA	नादरिशाह क सेना में दिल्ली के कत्लेआम कर रखा है। मानव हृदया क क्रूरता, कठ रता और पैशादकिता अपना दवकरालतमरूप धारण दकयेहुए है। दिल्ली उन दिनों भ ग-दवलासक केंद्र बन हुई थ । राज नदतका स्थान शेर -शायर ने ले दलयाथा।
LDA-1	स्त्रियों क बनाव-सिगांर के दसवाक ई काम न था। राज नदतका स्थान शेर -शायर ने ले दलया था। समस्त प्रन्तो से धन खिंच-खिंच कर दिल्ली आता था। उसका समसत ज वन रणभूदम में ह कटा था
LDA-SWSW	नादरिशाह क सेना में दिल्ली के कत्लेआम कर रखा है। मानव हृदया क क्रूरता, कठ रता और पैशादकिता अपना दवकरालतम रूप धारण दकये हुए है। तुम इस वक्त उनक सुंदर वस्त्राभूषणों से सजाकर मेरे सामने लाओं खबरारि , जरा भ ेरि न ह ! मै क ई उज्र या इनकार नहीं सुन सकता।
LDA-ISW	इस समया नादरि शाह ने बाशिाह महल में प्रवेश दकया।दिल्ली उन दिनों भ ग-दवलासक केंद्र बन हुई थ । राज नदतका स्थान शेर -शायर ने ले दलयाथा। उसका समसत ज वन रणभूदम में ह कटा था
LDA-RSW	नादरिशाह क सेना में दिल्ली के कत्लेआम कर रखा है। मानव हृदया क क्रूरता, कठ रता और पैशादकिता अपना दवकरालतमरूप धारण दकयेहुए है। दजधरआंख उठत थ , उधर से हटने का नाम न लेत थ । तुम इस वक्त उनक सुंदर वस्त्राभूषणों से सजाकर मेरे सामने लाओं खबरारि , जरा भ ेरि न ह !
Traditional LDA	बाजार बंद है। इस समया नादरि शाह ने बाशिाह महल में प्रवेश दकया।स्त्रियों क बनाव-सिगांर के दसवाक ई काम न था। वेश्याओं क चादीं थ । कहां रण-क्षेत्र के कष्ट और कहां यह सुख-साम्राज्य।
TF-IDF	कहीं घरों में आग लग हुई है, कहीं बाजार लुट रहा है; क ई दकस क फररया ि नहीं सुनता। रईसों क बेगमें महल से दनकाल जा रह है और उनक बेहुरती क जात है। नादरिशाह अपने सरदारों के साथ महल क सैर करता और अपन पसंद क चीजों क बट रता हुआ ि विाने -खास में आकर कारबि मसनि पर बैठ गया, सरदारों क वहां से लि जाने का हुक्म ेि दयिा, अपने सबहदथयाररख दयि और महल के रि गा क बुलाकर हुक्म दयिा–मै शाह बेगमों का ना ि ेखिना ाहिता हूं।
TextRank	कहीं घरों में आग लग हुई है, कहीं बाजार लुट रहा है; क ई दकस क फररया ि नहीं सुनता। रईसों क बेगमें महल से दनकाल जा रह है और उनक बेहुरती क जात है। नादरिशाह शाह महल में पहुंचा त वहां का सामान ेखिकर उसक आंखें खुल गयीं। नादरिशाह अपने सरदारों के साथ महल क सैर करता और अपन पसंद क चीजों क बट रता हुआ ि विाने -खास में आकर कारबि मसनि पर बैठ गया, सरदारों क वहां से लि जाने का हुक्म ेि दयिा, अपने सब हदथयाररख दयि और महल के रि गा क बुलाकर हुक्म दयिा–मै शाह बेगमों का ना ि ेखिना ाहिता हूं।
Latent Semantic Analysis (LSA)	गलियों मे खून क नदियां बह रह हैं। दिल्ली उन दिनों भ ग-दवलासक केंद्र बन हुई थ । समस्त प्रन्तो से धन खिंच-खिंच कर दिल्ली आता था और पान क भांति बहाया जाता था।
LexRank	नादरिशाह क सेना में दिल्ली के कत्लेआम कर रखा है। दिल्ली के ल ग घरों के द्वार बंद दकये जान क खैर मना रहे है। नादरिशाह शाह महल में पहुंचा त वहां का सामान ेखिकर उसक आंखें खुल गयीं। तुम इस वक्त उनक सुंदर वस्त्राभूषणों से सजाकर मेरे सामने लाओं खबरारि , जरा भ ेरि न ह ! मै क ई उज्र या इनकार नहीं सुन सकता।

above baseline models, we carried a statistical significance test. For this purpose, the significance level considered as 5%, or the confidence level is 95% while the sample size considered is 30. According to the assumed null hypothesis, the results produced by our model are not statistically significant against baseline algorithms while they are computed by chance.

To verify the contemplated hypothesis, "P-value that represents the percentage of observation of proposed approach against another baseline models (LDA-1, LDA, LexRank, TF-IDF and TextRank)" is computed as shown in Table 15, 16, 17 and 18 on both the datasets. The computed *P* values for topic diversity, retention ratio, ROUGE-1 score, and ROUGE-2 score show that our conjectured null hypothesis failed. It follows the alternative hypothesis i.e., the experimental results of our model and baselines are significantly different. Table 15 shows that *p*-values for GD of the proposed model are smaller than the significance level which in turn substantiates that the GD values are not computed by chance and are statistically significant. Similarly, much lower *p*-values for retention ratio (Table 16), ROUGE-1 (Table 17) score and ROUGE-2 score (Table 18) of our proposed model shows it is capable of generating quality summary than the baseline algorithms. During the test, we observed that our assumed null hypothesis failed and our model follows the alternative hypothesis. Hence, with the compression ratio between 0.1%–0.3%, our model significantly produces good content machine summaries.

Summary smoothing The main objective of summary smoothing is to remove any kind of noise from the produced summary and to generate a coherent, clear, and featured summary. To analyze the results, we compare the results obtained after and before employing summary smoothing. Figure 4a, b and c compare the ROUGE-1 scores of our four proposed models with baseline models after smoothing for a compression ratio of 10%, 20%, and 30% respectively on the Hindi dataset. Similarly, Fig. 4d, e and f illustrate the ROUGE-1 score of our four proposed models with baseline models after smoothing for a compression ratio of 10%, 20%, and 30% respectively on the English dataset. We also compare the ROUGE-2 score of our four proposed models with baseline models after smoothing for a compression ratio of 10%, 20%, and 30% respectively on the Hindi dataset, exemplified in Fig. 5a, b and c. Similarly, Fig. 5d, e and f demonstrate the ROUGE-2 score of our four proposed models with baseline models after smoothing for a compression ratio of between 10% - 30% on the English dataset. From the above results, we have noticed that our models have a high ROUGE score on comparing with other related systems.

Fig. 6 compares the gist diversity values obtained from four proposed models on Hindi and English data sets. Similarly, Fig. 7 compares the retention ratio values obtained from four proposed models on Hindi and English data set. We also assess the ROUGE-1 and ROUGE-2 score on the Hindi and English dataset obtained before and after applying the gist smoothing technique, shown in Fig. 8a and b respectively. Tables 7, 8, 9, 10, 11, 12, 13 and 14 summarize the results after applying the smoothing technique.

From all outcomes, we perceive that our proposed models perform preferable than the baseline algorithms for 10% - 30% compression ratios and given evaluation metrics. We also analyze the intelligibility, precision, and coherence of the generated gist increase in applying the smoothing technique. In terms of the compression ratio, we scrutinize that the least condensed gist is more coherent. Thirdly, we also discovered that the increase in the number of topics, accuracy, and quality of the summary can be increased. Finally, comparing results from before and after applying the gist smoothing approach shows a significant difference in terms of applied metrics.

Table 19 shows the summary of the sample input document from the Hindi literary novel for different models proposed along with baselines on a 10% compression ratio.

6 Conclusion

In current work, we have proposed a lexical features rich topic modeling method for automatic extractive text summarization for Hindi novel and story documents. The proposed model is implemented by infusing linguistic features into LDA based topic modeling to discover a set of quiescent pertinent topic-words amalgam from the input document. Further, we deploy the identified topic-words set to designate appropriate sentences from the input document to devise a constrict, diverse, salient, and coherent machine summary. Besides, four distinct sentence weighting scheme based variants are derived by manipulating the proposed system. Additionally, the summary smoothing technique involved to transfigure initially produced summary into a more definite, faultless, and redundancy free summary. Later on, we have performed experiments on two datasets, Hindi and English individually to ensure the effectiveness of the automated summaries of proposed models. We also compared these techniques with baselines models. The experimental study manifested that the results of our model were more encouraging and stimulating than the baseline approaches. In the future, we will try to address the following issues: i) include semantic features to enhance the summary essence; ii) introduce named entities in the model to extract more informative topics; and iii) evaluate the model summary against more summary evaluation metrics like Pyramid score, BLEU score, etc.

Acknowledgments The authors are grateful to anonymous reviewers for their valuable comments.

References

1. Aggarwal CC (2018) Text Summarization, In Machine Learning for Text, Springer, pp. 361–380
2. Al-Radaideh QA, Bataineh DQ (2018) A hybrid approach for arabic text summarization using domain knowledge and genetic algorithms, Cognit Comput, 1–19
3. Bairi R, Iyer R, Ramakrishnan G, Bilmes J (2015) Summarization of multi-document topic hierarchies using submodular mixtures, 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), vol. 1, pp. 553–563
4. Bamman D, Smith NA (2013) New alignment methods for discriminative book summarization, arXiv Prepr. arXiv1305.1319
5. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3(Jan):993–1022
6. Blogwriter, Munshi Premchand's Stories. [Online]. Available: http://premchand.kahaani.org/. (Accessed: 29-Mar-2019)
7. Brainy, "Brainy questions." [Online]. Available: https://brainly.in/subject/hindi. (Accessed: 27-Mar-2019)
8. Ceylan H (2011) Investigating the extractive summarization of literary novels. University of North Texas, Denton
9. Chi L, Li B, Zhu X (2014) Context-preserving hashing for fast text classification, In Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 100–108
10. Di Fabbrizio G, Aker A, Gaizauskas R (2013) Summarizing online reviews using aspect rating distributions and language modeling. IEEE Intell Syst 28(3):28–37

11. Elhadad M, Miranda-Jiménez S, Steinberger J, Giannakopoulos G (2013) Multi-document multilingual summarization corpus preparation, part 2: Czech, hebrew and spanish, In Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization, pp. 13–19
12. Erkan G, Radev DR (2004) Lexpagerank: prestige in multi-document text summarization, In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing
13. Erkan G, Radev DR (2004) Lexrank: graph-based lexical centrality as salience in text summarization. J Artif Intell Res 22:457–479
14. Gambhir M, Gupta V (2017) Recent automatic text summarization techniques: a survey. Artif Intell Rev 47(1):1–66
15. Gong Y, Liu X (2001) Generic text summarization using relevance measure and latent semantic analysis," in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 19–25
16. Gupta V (2013) Hybrid algorithm for multilingual summarization of Hindi and Punjabi documents, In Mining Intelligence and Knowledge Exploration, Springer, pp. 717–727
17. Gupta V, Kaur N (2016) A novel hybrid text summarization system for Punjabi text. Cogn Comput 8(2): 261–277
18. Hafeez R, Khan S, Abbas MA, Maqbool F (2018) Topic based Summarization of Multiple Documents using Semantic Analysis and Clustering, In 2018 15th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT), pp. 70–74
19. Hu Y-H, Chen Y-L, Chou H-L (2017) Opinion mining from online hotel reviews–a text summarization approach. Inf Process Manag 53(2):436–449
20. Huang T, Li L, Zhang Y (2016) Multilingual multi-document summarization with enhanced hLDA features, In Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, Springer, pp. 299–312
21. John A, Premjith PS, Wilscy M (2017) Extractive multi-document summarization using population-based multicriteria optimization. Expert Syst Appl 86:385–397
22. Kabadjov M, Atkinson M, Steinberger J, Steinberger R, Van Der Goot E (2010) NewsGist: a multilingual statistical news summarizer, In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 591–594
23. Kaljahi R, Foster J, Roturier J (2014) Semantic role labelling with minimal resources: Experiments with french, In Proceedings of the Third Joint Conference on Lexical and Computational Semantics (* SEM 2014), pp. 87–92
24. Kazantseva A, Szpakowicz S (2010) Summarizing short stories. Comput Linguist 36(1):71–109
25. Lee S, Belkasim S, Zhang Y (2013) Multi-document text summarization using topic model and fuzzy logic, In International Workshop on Machine Learning and Data Mining in Pattern Recognition, pp. 159–168
26. Li L, Forascu C, El-Haj M, Giannakopoulos G (2013) Multi-document multilingual summarization corpus preparation, part 1: Arabic, English, Greek, Chinese, Romanian
27. Lin C-Y (2004) Rouge: a package for automatic evaluation of summaries, Text Summ. Branches Out
28. Litvak M, Vanetik N, Liu C, Xiao L, Savas O (2015) Improving summarization quality with topic modeling, In Proceedings of the 2015 Workshop on Topic Models: Post-Processing and Applications, pp. 39–47
29. Litvak M, Vanetik N, Li L (2017) Summarizing Weibo with Topics Compression, In International Conference on Computational Linguistics and Intelligent Text Processing, pp. 522–534
30. Liu N, Tang X-J, Lu Y, Li M-X, Wang H-W, Xiao P (2014) Topic-Sensitive Multi-document Summarization Algorithm, In 2014 Sixth International Symposium on Parallel Architectures, Algorithms and Programming, pp. 69–74
31. Luhn HP (1999) The automatic creation of literature abstracts. Adv Autom Text Summ:15–22
32. Mendoza M, Bonilla S, Noguera C, Cobos C, León E (2014) Extractive single-document summarization based on genetic operators and guided local search. Expert Syst Appl 41(9):4158–4169
33. Mihalcea R, Tarau P (2004) Textrank: bringing order into text. In: Proceedings of the 2004 conference on empirical methods in natural language processing
34. Mishra R, Bian J, Fiszman M, Weir CR, Jonnalagadda S, Mostafa J, del Fiol G (2014) Text summarization in the biomedical domain: a systematic review of recent research. J Biomed Inform 52:457–467
35. Na L, Ming-xia L, Ying L, Xiao-jun T, Hai-wen W, Peng X (2014) Mixture of topic model for multi-document summarization, In The 26th Chinese Control and Decision Conference (2014 CCDC), pp. 5168–5172
36. Na L, Ying L, Xiao-jun T, Hai-wen W, Peng X, Ming-xia L (2016) Multi-document summarization algorithm based on significance sentences, In 2016 Chinese Control and Decision Conference (CCDC), pp. 3847–3852

37. Nomoto T, Matsumoto Y (2003) The diversity-based approach to open-domain text summarization. Inf Process Manag 39(3):363–389
38. Oufaida H, Blache P, Nouali O (2015) Using Distributed Word Representations and mRMR Discriminant Analysis for Multilingual Text Summarization, In International Conference on Applications of Natural Language to Information Systems, pp. 51–63
39. Ozsoy MG, Alpaslan FN, Cicekli I (2011) Text summarization using latent semantic analysis. J Inf Sci 37(4):405–417
40. Parveen D, Strube M (2015) Integrating importance, non-redundancy and coherence in graph-based extractive summarization, In Twenty-Fourth International Joint Conference on Artificial Intelligence
41. Patel A, Siddiqui T, Tiwary US (2007) A language independent approach to multilingual text summarization, In Large scale semantic access to content (text, image, video, and sound), pp. 123–132
42. Radev DR et al. (2004) MEAD-a platform for multidocument multilingual text summarization
43. Rani R, Lobiyal DK (2018) Social Choice Theory Based Domain Specific Hindi Stop Words List Construction and Its Application in Text Mining, In International Conference on Intelligent Human Computer Interaction, pp. 123–135
44. Rani R, Lobiyal DK (2018) Automatic Construction of Generic Stop Words List for Hindi Text, In Procedia Computer Science Elsevier Journal, pp. 1–7
45. Roul RK, Mehrotra S, Pungaliya Y, Sahoo JK (2019) A New Automatic Multi-document Text Summarization using Topic Modeling, In International Conference on Distributed Computing and Internet Technology, pp. 212–221
46. Sanchez-Gomez JM, Vega-Rodríguez MA, Pérez CJ (2018) Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach. Knowledge-Based Syst 159:1–8
47. Singh J, Gupta V (2017) An efficient corpus-based stemmer. Cogn Comput 9(5):671–688
48. Torres-Moreno J-M, St-Onge P-L, Gagnon M, El-Beze M, Bellot P (2009) Automatic summarization system coupled with a question-answering system (qaas), arXiv Prepr. arXiv0905.2990
49. Wang D, Zhu S, Li T, Gong Y (2009) Multi-document summarization using sentence-based topic models, In Proceedings of the ACL-IJCNLP 2009 conference short papers, pp. 297–300
50. Wikipedia, Premchand (2019) [Online]. Available: https://en.wikipedia.org/wiki/Premchand. (Accessed: 29-Mar-2019)
51. Wu Z, Lei L, Li G, Huang H, Zheng C, Chen E, Xu G (2017) A topic modeling based approach to novel document automatic summarization. Expert Syst Appl 84:12–23
52. Yang G, Wen D, Chen N-S, Sutinen E (2015) A novel contextual topic model for multi-document summarization. Expert Syst Appl 42(3):1340–1352
53. Yao J, Wan X, Xiao J (2015) Phrase-based compressive cross-language summarization, In Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 118–127
54. Yao J, Wan X, Xiao J (2015) Compressive document summarization via sparse optimization, In Twenty-Fourth International Joint Conference on Artificial Intelligence
55. Zhuang L, Jing F, Zhu X-Y (2006) Movie review mining and summarization, In Proceedings of the 15th ACM international conference on Information and knowledge management, pp. 43–50