







# S-LDA: Documents Classification Enrichment for Information Retrieval

Amani Drissi<sup>1</sup> , Anis Tissaoui<sup>2</sup> , Salma Sassi<sup>2</sup> , Richard Chbeir<sup>3</sup> ,  
and Abderrazak Jemai<sup>4</sup> 

<sup>1</sup> FST, University of Manar, SERCOM, Tunis, Tunisia  
drissiamani19@gmail.com

<sup>2</sup> VPNC Lab., FSJEGJ, University of Jendouba, 8189 Jendouba, Tunisia  
{anis.tissaoui,salma.sassi}@fsjegj.rnu.tn

<sup>3</sup> E2S UPPA, LIUPPA, University Pau & Pays Adour, EA3000 Anglet, France  
rchbeir@acm.org

<sup>4</sup> INSAT, University of Carthage, SERCOM, Tunis, Tunisia  
Abderrazak.Jemai@insat.rnu.tn

**Abstract.** In recent years, the research on topic modeling techniques has become a hot topic among researchers thanks to their ability to classify and understand a large text corpora which has a beneficial effect on information retrieval performance, but recently user queries are more complicated because they need to know not only which documents are most helpful to them, but also which parts of documents are more or less related to their request. Also, they need to search by topic or document, not merely by keywords.

In this context, we propose a new approach of automated text classification based on LDA topic modeling algorithm and the rich semantic document structure which helps to semantically enrich the generated classes by indexing them in the documents sections according to their probabilities distribution and visualize them through a hyper-graph.

Experiments have been conducted to measure the effectiveness of our solution compared to topic modeling classification approaches based on text content only. The results show the superiority of our approach.

**Keywords:** Document classification · Machine learning · LDA topic model · Document structure · Hyper-graph · Information retrieval

## 1 Introduction

The text classification task has recently attracted significant attention from researchers as an important paradigm for understanding massive text corpora.

So, to better manage the large amount of textual documents, it seems crucial to use new techniques or tools that deals with automatically organizing, searching and indexing the large collection of documents in order to facilitate the information access [15].

Topic modeling for information retrieval has attracted significant attention and demonstrated good performance in a wide variety of tasks [22] because it provides a convenient way to analyze large textual corpus and find abstract topics. It can discover the mixture of hidden or “latent” topics that varies from one document to another. It is successfully used in many applications, such information retrieval, analyzing historical documents, multilingual data and machine translation and understanding scientific publications [11].

LDA-based approaches have proven to provide the best result in document classification [1, 7, 12], thanks to their ability to map a query to its relevant documents at the semantic level. In addition, these models address the problem of language discrepancy between Web documents and search queries by grouping different terms that occur in a similar context into the same semantic cluster [16].

The process of extracting information has evolved in response to change user requirements. Today, the queries used by the user to interrogate the system have evolved from a simple keyword or a list of words to an entire topic. In addition, they can go through the document topics to see which documents are the most or the least similar based on the probability distribution of the topics in each used document. Also, it can go much further to search for their desired topic by document section to see which sections are more or less similar to their asked topic by extracting the document sections most relevant to the user request.

As user needs change, managing them will become more complicated even with the use of a powerful technique such as LDA, which is very efficient at the level of classifying documents in an probabilistic way according to their related topics, but do not able to map a query to its relevant sections in a large collection of documents or to search the most or the least similar documents according to their topics densities. Additionally, this technique do not able to index the extracted topics in their document sections, as a result, the information retrieval system will be not able to map a topic query to most relevant sections in the given corpus.

In order to overcome these challenges, we propose a new automatic approach named S-LDA to classify a large text corpora and semantically enrich the generated classes by integrating the document structure in the classification process. Our approach was represented through a hyper-graph which helps to improve the classification accuracy and make the information retrieval model more accurate, scalable and efficient.

To achieve our objectives, two main challenges have been addressed in our study:

1. How to automatically classify a textual corpus in a probabilistic way?
2. How to semantically enrich the extracted classes in order to ameliorate the information retrieval process?

The rest of the paper is organized as follows. We study in Sect. 2 the related works and we illustrate a comparative study between existing approaches. Next, we explain in Sect. 3 the methodology of our approach which consists of classifying a text corpora collecting from many web pages according to their dominant topics using LDA and document structure. Section 4 describes the experiments

conducted to validate our approach. Finally, Sect. 5 concludes the paper and discusses some future work.

## 2 Related Works

Text classification is an important task which could be used for information management applications by automatically allocating a specified document to one or more predefined classes. This technique aims to determine whether a given document belongs to the given category or not by looking at the words or terms of that category. Furthermore, text classification aids users' hold their fields of attention, specify them to be easily separated out texts that are not related to their attention by automatically grouping the texts according to their subjects [5, 18]. There are several works in the literature handling the text classification issue in order to facilitate the information retrieval task which help to make better decisions with more performance. [2, 5] classify text documents according to a predefined class using supervised machine learning algorithms and based on the text contents. [8] explores the rich semantic structure in order to automatically classify Elsevier articles and facilitate the analysis of these papers after the submission stage which helps to accelerate the papers treatments and guarantee a better performance because the article that does not respect the requested structure will be rejected. After that, the accepted papers in the first stage will be classified according to their text contents using a supervised machine learning algorithm. The combination between the document structure and the text content improve the classification accuracy.

On the other hand, several approaches are dedicated to automatically classify text document based on probabilistic techniques such as [9, 10, 14, 21], these approaches used unsupervised topic modeling algorithms, especially LDA [24] in order to model a given textual corpus in a probabilistic way which has a significant role and demonstrated good performance in the information retrieval task. These approaches treat the document as a probability distribution of topics which help to discover the mixture of hidden or "latent" topics that varies from document to document in a given corpus.

**Table 1.** Comparative study between existing studies

Criterion/Existing study	Challenge 1			Challenge 2	
	C1	C2	C3	C4	C5
Kadhim [5]	Unstructured	Non-probabilistic	Document	Predefined classes	No
Gong [9]	Unstructured	Probabilistic	Document	Topics	No
Bitew [8]	Unstructured	Non-Probabilistic	Document	Predefined classes	Yes
Luo [2]	Unstructured	Non-Probabilistic	Document	Predefined classes	Yes
Pavlinek [10]	Unstructured	Probabilistic	Document	Topics	No
Qiuxing [14]	Unstructured	Probabilistic	Document	Topics	No
Kim [21]	Unstructured	Probabilistic	Document	Topics	No
S-LDA	Unstructured	Probabilistic	Document + Section	Hyper-graph	Yes

In order to compare the existing approaches and to overcome the challenges described previously, we define here five criteria with respect to the defined challenges:

**Challenge 1.** How to automatically classify a heterogeneous corpus in a probabilistic way?

- **Criterion 1 (C1):** The Input data type which could be: (i) Structured, (ii) Semi-structured or (iii) Unstructured data.
- **Criterion 2 (C2):** The classification type, this criterion could be a Probabilistic classification or Non-probabilistic classification.
- **Criterion 3 (C3):** The level of classification, this criterion could be Document level or Section level.

**Challenge 2.** How to semantically enrich the extracted class in order to ameliorate the information retrieval process?

- **Criterion 4 (C4):** The output task which could be “Predefined classes”, “Topics” or “Hyper-graph”.
- **Criterion 5 (C5):** The document structure, it could be “Yes” if the approach explored the document structure or “No” if it did not.

Our comparison demonstrates that all existing works used unstructured data to classify their documents. Also, Table 1 illustrates that most of existing studies based on a probabilistic classification in order to model their textual corpus. In the other hand, these approaches neglected the rich semantic documents structure in their classification process, except [8] who combined a supervised machine learning technique with the documents structure in order to classify his textual corpus. However, none of the existing studies using probabilistic techniques integrate the document’s structure to semantically enrich the output classes or topics in order improve the information retrieval process. We noticed that all the existing approaches tried to manage the large used corpus by classifying the semantically similar documents in the same cluster, for our proposed approach, the documents and the document’s sections are classified based on their relevant topics. We observed also that the output task of all the existing studies was a predefined classes when the researchers used a supervised machine learning technique to classify their textual corpus or a set of topics when they used a topic modeling technique, for our approach, the output task was a hyper-graph. The main contribution of the paper consists of developing a new automatic method named S-LDA for automatically classifying a textual corpus based on document’s structure and LDA probabilistic topic model. One strong aspect of our contribution is the combination of the topic model and the semantic extracted tree structure to semantically enrich the generated topics and improve the information retrieval task.

### 3 Methodology

The originality of S-LDA lies not only in the integration of document structure in the classification process, but also in the generation of a hyper-graph which helps to automatically classify a large textual corpus according to their topics as well as index the generated topics in document sections based on their probability distribution in order to explore more deeply the relationships between topics. The proposed technique extends LDA algorithm by taking into account the semantic structure of the input documents in the classification phase.

The S-LDA framework is summarized in Fig. 1 which consists of two modules: (1) Document structure analysis, (2) Document classification. We detail them below.

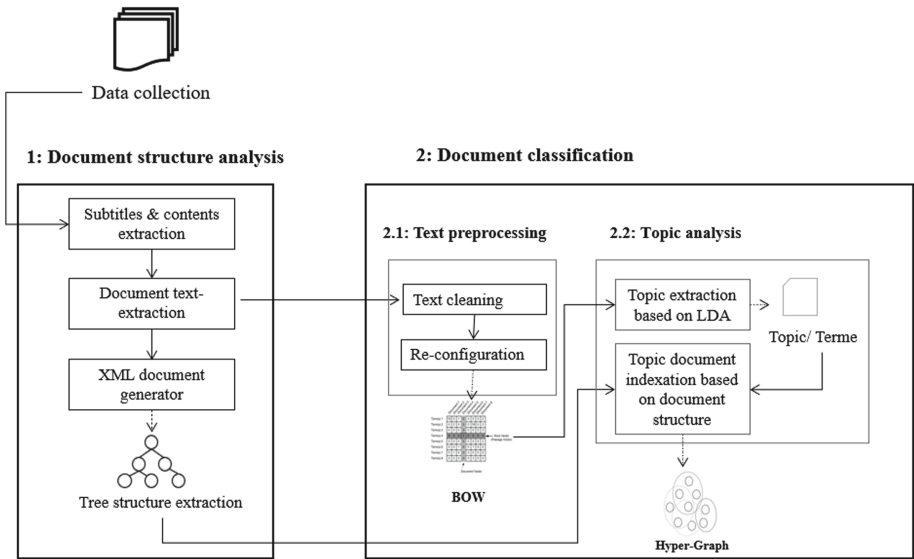


Fig. 1. S-LDA framework.

#### 3.1 Document Structure Analysis

The Documents structure analysis aims to extract the semantic documents structure which helps to facilitate the classification of each document sections according to their relevant topics, this phase consists of three steps:

- **Subtitles and contents extraction:** this module consists of exploiting the document structure in the classification task. In this step, we extract the subtitles and the document contents in order to convert the used textual documents to a semantic tree structure and conserve the documents/word relationships which helps to better explore the relationships between the extracted topics based on their context root path.

- **Document text extraction:** this module consists of extracting the text from the collected data which will be restructured and used as an input in the Document classification phase in order to model the used corpus data according to their dominant topics.
- **XML document generation:** this module aims at representing the extracted text into an XML representation which provides semantic knowledge about the document where the HTML mark-up only indicates the structure and layout of documents, but not the document semantics [17]. As a result, we generate a tree structured document which facilitate the semantic process of the data by conserving the semantic context of each word. The idea is to produce a semantic tree structure to be consequently exploited in topics indexing and topics relationships extraction which makes the data access more easier.

### 3.2 Document Classification

This phase consists of cleaning data and extracting the corpus topics in order to classify each document as well as each document section according to their relevant topics which helps to improve the information retrieval systems.

1. **Text preprocessing:** preprocessing is an important task and critical step in Natural Language Processing (NLP), it acts a significant role [13] for transferring text from human language to machine-readable format and it affects substantially the results of the experiments. The preprocessing stage is important to structure the unstructured text and keep the keywords which are useful to represent the category of text topics [23]. Natural language text can contain many words with no specific meaning, such as prepositions, pronouns, etc. So, after obtaining the text, the preprocessing process consists of two steps: **(a)** Text cleaning step and **(b)** Re-configuration step:
  - (a) The text cleaning step** includes three sub-modules:
    - Normalization: this sub-module aims to transform the text into a single basic format or a more uniform sequence by converting the characters to lowercase, deleting all numbers, symbols, removing punctuation. This step is important in order to shrink the size of the vocabulary.
    - Tokenization: this sub-module aims to divide the given text into sentences and each sentence into smaller pieces called tokens (words).
    - Lemmatization: this sub-module aims to provide the Part-Of-Speech (POS).
    - Bi-grams extraction: this sub-module aims to extract the bi-grams for each tokenized document. This task consists of combining multi-word terms into single token, such as data-mining, web-page and machine-Learning.
  - (b) The Re-configuration step** aims to convert text data to an appropriate format, this task is necessary for an automated process. The used method

to establish this step is a Bag Of Words matrix representation where each document in our corpus represent a vector of tokens and the tokens represents the document terms. The output of this task is a BOW matrix where we have the corpus documents as well as the number of times of each term in the document.

2. **Topic analysis** The topic analysis module consists of two steps which are respectively: (a) Topics extraction based on LDA, (b) Topic document indexation based on document structure.
  - (a) **Topics extraction based on LDA:** this sub-module consists of four steps which are respectively: The model parameters initialization, the model evaluation, the model execution and the documents cluster.
    - **The model parameters initialization:** this sub-module aims at stabilizing the LDA model. The stabilization process is based on **alpha** parameter representing a document topic density (Document concentration). (A high alpha value point to that every document is tend to contain a mixture of the most of the topics, and not any single topic especially. The lower value of alpha, means that the documents contain fewer topics [4].) and **beta** parameter representing a topic word density (Topic concentration). (It assumes that the topic is made of up most of the words and result in a more specific word distribution per topic. A high beta value means each topic is more likely to contain a specific word mix and in practice, that leads to topics being more alike in terms of what words they include and the lower value of beta, means they are composed of few word [4]) values as well as **the topics number**. So, to have a stable and efficient model, it is necessary to select the optimal combination of alpha and beta values by taking into consideration the optimal number of topics according to the used corpus, this step was explained in more details in our previous approach named Learn2Construct [3].
    - **The model evaluation:** this sub-module uses the coherence metrics to quantitatively evaluate the quality of the generated topics. It quantifies how much the words on one topic are, in fact, related to each other and are thus an attempt to capture human interpretability of topics [20].
    - **The model execution:** this sub-module executes the LDA algorithm using the optimal alpha, beta and topic numbers values according to the used corpus based on the coherence metric.
    - **The documents cluster:** this sub-module consists in generating and viewing the document/topic clusters as well as the topic/term clusters.
  - (b) **Topic document indexation based on documents structure:** to index the document topics with high accuracy, we tacked the most semantically expressive terms for each topic. The input of this step was the semantic tree structure extracted in the first S-LDA framework phase. It is crucial to exploit not only the text content of a document, but also the rich semantic structure that organizes the document contents and

their latent semantics, which involves reasoning under a probabilistic way. This sub-module, indexes the extracted topics using the extracted semantic tree structure which helps to explore more deeply the semantic relationships between the topics as well as their terms. However, each document is represented with the highest probability topics and each extracted topic is represented with the highest probability terms. In addition, the topic/document indexation based on document structure improves the classification accuracy and have beneficial effects on information retrieval performance. To achieve our goal and facilitate the topics sections indexation, we generated in this step three clusters: the first one indicates the probability distribution of each topic in each used document. The second one give us an idea about the probability distribution of each term in each generated topic. And we propose, in this paper, a third probability distribution which indicates the probability distribution of each topic in the document sections. To calculate this probability, we propose the above formula:

$$P(T_i/S) = \sum_{(t_i \in T), (S_j \in d)} \frac{P(t_i/S_j)}{P(t_i/d)} \quad (1)$$

where T is a set of the generated topics, t is a set of words describing the topic, S represent the document sections and d represent the document. The output of this step was a hyper-graph which guarantee a more expressive data structure that capture both the relations and the intersections of nodes because of its expressiveness, also, it can provide further insights regarding intersections and subsumption between nodes [6]. The integration of this hyper-graph in the information retrieval system makes it more efficient with the ability to answer to any complex user queries.

## 4 Experiments

Our experiments aim to evaluate the performance of our approach based on annotated and no-annotated documents structure using several metrics such as: the precision, recall and F-score.

### 4.1 Environment

As a programming language, we used Python. For the natural language processing we used NLTK<sup>1</sup> (Natural Language Toolkit), this library is used for tokenization, lemmatization and stop words removal. Regarding topic modeling, we used Gensim<sup>2</sup>, a Python library for topic modelling, document indexing and similarity retrieval with large corpora. To train our models, we used laptop on Intel core (TM) i7-6500U 2.59 GHz of CPU with 8 GB of RAM and 64 GB of disk.

<sup>1</sup> <https://www.nltk.org/>.

<sup>2</sup> <https://pypi.org/project/gensim/>.



## 4.2 Evaluation Metrics

Topic Coherence measures scores a single topic by measuring the degree of semantic similarity between high scoring words in the topic [19]. The coherence of a topic calculated as the sum of pairwise distributional similarity scores over the set of topic words,  $V$ . We generalize this as

$$Coherence(V) = \sum_{(V_i, V_j) \in V} Score_{(V_i, V_j, e)} \quad (2)$$

where  $V$  is a set of words describing the topic,  $V_i$  and  $V_j$  are topic words and  $e$  indicates a smoothing factor which guarantees that score returns real numbers. (We will be exploring the effect of the choice of  $e$ ; the original authors used  $e = 1$ .)

Also, we evaluate our approach using conventional measures in information retrieval such as recall, precision, and F-score, denoted as  $R$ ,  $P$ , and F-score respectively. The Recall  $R$  is defined as:

$$R = \frac{C_{Wp}}{K_{Wp}} \quad (3)$$

where  $C_{Wp}$  defines the number of correct learned topic sections and  $K_{Wp}$  defines the number of correct topics document indexation.

The Precision  $P$  is defined as:

$$P = \frac{Cd_{Wp}}{Id_{Wp}} \quad (4)$$

where  $Co_{sc}$  is the number of correct learned topic sections,  $Id_{sc}$  is the total number of learned topics documents indexation.

To assess the performance of our approach, we note that precision measure alone is not sufficient. The F-score measure (or F1) is defined as the harmonic mean of recall and precision:

$$F - score = \frac{2 \times P \times R}{P + R} \quad (5)$$

## 4.3 Experimental Protocol

We have generated 350 Scientific papers from the web using Springer API<sup>3</sup> in PDF format for three domains which are: Ontology learning, Biological and Artificial Intelligence.

The objective of our study is to evaluate the performance of our approach using two documents types: “Annotated documents structures” where the document has an annotated sections and subsections which helps to better index the generated topics. And “No-annotated documents structures” where the document has no annotated section. In our experiments we used two textual corpus: the first one contains a set of annotated documents structures (represented

<sup>3</sup> <https://dev.springernature.com/>.

by the 350 scientific papers from springer) and the second one represented by 350 heterogeneous documents (50 annotated documents structures and 300 no-annotated ones) collected from the web arbitrarily and they are in different formats such as PDF, HTML and Words.

In the first step, we build our classification models by considering a sequence of topics values that starts with 2 up to 20, to guarantee a better classification we have chosen the smallest number of topics that has the highest coherence value. In our study, the optimal number of topics was 3.

Dominant_Topic	Topic_Perc_Contrib	Keywords	Document_Name
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	Paper 1
Topic3	0.783599973	ontology, data, approach, process, system, technique, learning	Paper 2
Topic2	0.98180002	chemical, biology, concentration, paper, gene, system, result	Paper 3
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	Paper 4
Topic3	1	ontology, data, approach, process, system, technique, learning	Paper 5
Topic2	1	chemical, biology, concentration, paper, gene, system, result	Paper 6
Topic2	1	chemical, biology, concentration, paper, gene, system, result	Paper 7
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	Paper 8
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	paper 9
Topic3	1	ontology, data, approach, process, system, technique, learning	paper 10
Topic1	0.999100029	paper, data, intelligence, Machine-learning, system, deep-learnin, information	paper 11
Topic2	0.917900026	chemical, biology, concentration, paper, gene, system, result	paper 12
Topic2	1	chemical, biology, concentration, paper, gene, system, result	paper 13
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	paper 14
Topic2	1	chemical, biology, concentration, paper, gene, system, result	paper 15
Topic1	0.846000016	paper, data, intelligence, Machine-learning, system, deep-learnin, information	paper 16
Topic1	1	paper, data, intelligence, Machine-learning, system, deep-learnin, information	paper 17
Topic2	1	chemical, biology, concentration, paper, gene, system, result	paper 18
Topic3	1	ontology, data, approach, process, system, techniqe, learning	paper 19

**Fig. 2.** LDA classification output

The Fig. 2 shows the result of LDA based classification, which helps to discover the most relevant topic of each scientific paper as well as the most expressive terms of each topic. It is important to mention that if the probability distribution of the topic in the document less than 10% this probability will be ignored, that is why the most used articles are 100% related to one topic which is the most dominant.

Document	Topics	Outline	Topic-Section-Distribution
paper 2	['ontology', 'data', 'approach', 'process', 'system', 'technique', 'learning']	abstract	0.451325487
		1 introduction	0.582364781
		2 literature review	0.654782139
		2.1 background	0.758786123
		2.2 related work	0.854796358
		3 methodology	0.785946823
		3.1 preprocessing	0.387124569
		3.2 term extraction	0.252136942
		3.3 topic modeling	0.101200004
		3.4 concepts & relation extraction	0.821546987
		3.5 ontology visualization	0.654125879
		4 experiments	0.853694125
		4.1 experimental protocols	0.75692001
		4.2 evaluation	0.659800321
		5 conclusion	0.65842395

**Fig. 3.** LDA combined with document structure classification output

We integrated the document structure in the classification process in order to enrich the generated topics and give the user more details about their documents corpus which helps to interpret the data more deeply. The Fig. 3 shows the probability distribution of the third topic in each section in the second paper. In this reason, if two topics are related to the same section, or if one of these topics is more distributed in a section and the second one is more frequent in a subsection of this section, certainly, there is a semantic relationship between these two topics which designed through a hyper-graph model. So, the integration of topics relationships in the information retrieval system helps to improve the classification accuracy and have beneficial effects on this task.

**Table 2.** Model performance evaluation

Model	Precision (P)	Recall (R)	F-score (F-S)
Annotated documents	0.88	0.85	0.86
Non-annotated documents	0.21	0.15	0.175

The Table 2 resumes the results of the automatic evaluation of our approach using annotated and non-annotated documents. The obtained values are automatically calculated with reference to the classification and annotated sections given by Springer for each document. The discussed results in Table 2 shows that the existing of documents structures positively influences the performance of our approach. However, the annotated documents structures helps also to facilitate the topic sections indexation, but the use of non-annotated documents structure helps only to generate the most relevant topics of each used document.

## 5 Conclusion

In this paper, we have combined a machine learning approach which consists in the use of LDA -in order to generate the document/topic clusters, topic/section clusters, as well as the topics/term clusters- with the documents structure in order to semantically enrich the generated topics and index them in the documents sections based on their probabilities distribution which helps to improve the information retrieval task. We also evaluated the performance of our approach based on the precision, recall and F-score, which are the most recommended measures especially in information retrieval domain. In future work, we aim at extracting the semantic relationships between the generated topics based on the generated hyper-graph, and evaluating the performance of our approach with different user queries.

## References

1. Slimane, B., Mounsif, M., Ghada, I.D.: Topic modeling: comparison of LSA and LDA on scientific publications. In: DSDE 2021, Barcelona, Spain, 18–20 February 2021 (2021)
2. Luo, X.: Efficient English text classification using selected machine learning techniques. *Alexandria Eng. J.* **60**, 3401–3409 (2021). <https://doi.org/10.1016/j.aej.2021.02.009>
3. Khemiri, A., Drissi, A., Tissaoui, A., Sassi, S., Chbier, R.: Learn2Construct: an automatic ontology construction based on LDA from textual data. In: MEDES 2021, Proceedings of the 13th International Conference on Management of Digital Ecosystems, November 2021, pp. 49–56 (2021)
4. Shaymaa, H.M., Al-augby, S.: LSA and LDA topic modeling classification: comparison study on E-books. *Indones. J. Electr. Eng. Comput. Sci.* **19**(1), 353–362 (2020)
5. Kadhim, A.I.: Survey on supervised machine learning techniques for automatic text classification. *Artif. Intell. Rev.* **52**(1), 273–292 (2019). <https://doi.org/10.1007/s10462-018-09677-1>
6. Devezas, J., Nunes, S.: Hypergraph-of-entity. *Open Comput. Sci.* **9**, 103–127 (2019)
7. Kherwa, P., Bansal, P.: Topic modeling: a comprehensive review. *Researchgate* (2018). <https://www.researchgate.net/publication/334667298-Topic-Modeling-A-Comprehensive-Review>
8. Bitew, S.K.: Logical structure extraction of electronic documents using contextual information. University of Twente (2018)
9. Gong, H., You, F., Guan, X., Cao, Y., Lai, S.: Application of LDA topic model in E-mail subject classification. In: International Conference on Transportation & Logistics, Information & Communication, Smart City (TLICSC 2018) (2018)
10. Pavlinek, M., Podgorelec, V.: Text classification method based on Self-Training and LDA topic models. *Expert Syst. Appl. J.* **80**, 83–93 (2017)
11. Boyd-Graber, J., Yuening, H., Mimno, D.: Applications of topic models. *Found. Trends Inf. Retr.* **11**(2–3), 143–296 (2017). <https://doi.org/10.1561/15000000030>
12. Rani, M., Dhar, A.K., Vyas, OP.: Semi-automatic terminology ontology learning based on topic modeling. *Semantic scholar* (2017). <https://www.semanticscholar.org/paper/Semi-automatic-terminology-ontology-learning-based-Rani-Dhar/4948d5f16cd1f6733f2d989577119fdd18c83d02>
13. Rajasundari, T., Subathra, P., Kumar, P.: Performance analysis of topic modeling algorithms for news articles. *J. Adv. Res. Dyn. Control Syst.* **11**, 175–183 (2017)
14. Chen, Q., Yao, L., Yang, J.: Short text classification based on LDA topic model. *IEEE, ICALIP* (2016)
15. Rubayyi, A., Khalid, A.: A survey of topic modeling in text mining. *Int. J. Adv. Comput. Sci. Appl.* **6**(1), 147–194 (2015)
16. Shen, Y., He, X., Gao, J., Deng, L., Mesnil, G.: A latent semantic model with convolutional-pooling structure for information retrieval. *ACM* (2014). <https://doi.org/10.1145/2661829.2661935>
17. Tyagi, N., Rishi, R., Agarwal, R.P.: Semantic structure representation of HTML document suitable for semantic document retrieval. *Int. J. Comput. Appl.* **46**(13), 0975–8887 (2012)
18. Bindra A.: SocialLDA: scalable topic modeling in social networks. Dissertation University of Washington (2012)

19. Keith, S., Philip, K., David, A., David, B.: Exploring topic coherence over many models and many topics. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 952–961 (2012)
20. David, M., Hanna, M. W., Edmund, T., Miriam, L., Andrew, M.: Optimizing semantic coherence in topic models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, United Kingdom, pp. 262–272. Association for Computational Linguistics, USA (2011)
21. Kim, B.G., Park, S.I., Kim, H.J., Lee, S.H.: Automatic extraction of apparent semantic structure from text contents of a structural calculation document. *J. Comput. Civ. Eng.* **24**(3), 312–324 (2010)
22. Wu, D., Wang, H.L.: Role of ontology in information retrieval. *J. Electron. Sci. Technol. China* **4**(2), 148–154 (2006). <https://www.researchgate.net/publication/301227711>
23. Gonçalves, T., Quaresma, P.: Evaluating preprocessing techniques in a text classification problem. São Leopoldo, RS, Bras. SBC-Sociedade Brasileira De Computacao, pp. 841–850 (2005)
24. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)