10.31341/jios.46.2.3
Open Access
UDC 004.021:004.62
Preliminary Communication

Understanding Document Thematic Structure: A Systematic Review of Topic Modeling Algorithms

Seun Osuntoki *osuntokiseun@gmail.com*
Department of Computer Science
Faculty of Science, University of Lagos, Lagos, Nigeria

Victor Odumuyiwa *vodumuyiwa@unilag.edu.ng*
Department of Computer Science
Faculty of Science, University of Lagos, Lagos, Nigeria

Oladipupo Sennaike *osennaike@unilag.edu.ng*
Department of Computer Science
Faculty of Science, University of Lagos, Lagos, Nigeria

Abstract

The increasing usage of the Internet and other digital platforms has brought in the era of big data with the attending increase in the quantity of unstructured data that is available for processing and storage. However, the full benefits of analyzing this large quantity of unstructured data will not be realized without proper techniques and algorithms. Topic modeling algorithms have seen a major success in this area. Different topic modeling algorithms exist and each one either employs probabilistic or linear algebra approaches. Recent reviews on topic modeling algorithms dwell majorly on probabilistic methods without giving proper treatment to the linear-algebra-based algorithms. This review explores linear-algebra-based topic models as well as probability-based topic models. An overview of how models generated by each of these algorithms represent document thematic structure is also presented.

Keywords: Topic models, Information Retrieval, Text Mining, NMF, document structure

1. Introduction

The increase in the usage of the Internet as a social and educational tool has increased the amount of unstructured data available. Aside the common social media platforms like Facebook, Twitter and LinkedIn, the Internet has seen the emergence of several other platforms like Quora, Medium and other personal blogging websites. These platforms along with the other social media platforms have drastically increased the amount of textual data available.

The ubiquity of textual data is not limited to social networks and article publishing platforms. Several organizations now generate documents from communications both from internal and external sources. They also generate textual documents from work reports, legal documents, project documents and instant messaging platforms like Slack and Microsoft Teams. In essence, textual documents are all around us, and they serve various purposes [1], [2].

From a reader's perspective, textual document might contain a particular message, intention or purpose which commonly suffices for understanding. However, it holds other insights that cannot be directly observed. Unlike structured and numeric data, textual data cannot be easily analyzed; thus arises the need to use a model that can carry out such analysis. Topic modeling is important when trying to understand the underlying theme within large collections of textual documents. Therefore, topic modeling makes it possible to derive insights that are difficult to derive from traditional text mining methods [3] as it presents a probabilistic view into the hidden structure of an unstructured document and thus improving the quality and extent of analysis that can be performed on such data.

This review paper is in four sections. The first and the second section presents an overview of document thematic structure alongside discussion on the similarities and differences between document classification/clustering algorithms and topic modeling algorithms. The third section categorizes topic modeling algorithm based on algorithmic approaches and the fourth section reviews current research areas in topic modeling.

2. Document Thematic Structure

The need to understand and analyze document thematic structure has existed prior to the use of automated data analyzing software. Domains such as psychology and computational linguistic perform various analysis on textual data in order to derive necessary insights [4]. Content analysis and thematic analysis are prior methods for extracting relevant insights from text/unstructured data and they are performed using manual counting and coding techniques [5].

Likewise, in text mining, various automated methods exist for extracting insights from large unstructured data. Text mining methods generally include supervised and unsupervised algorithms. The supervised algorithms are majorly text classification algorithms while the unsupervised algorithms are generally clustering algorithms [6], [7]. Both methods have been widely applied in text analysis [8]–[12].

From a linguistic perspective, a textual document is a collection of paragraphs and paragraphs are collection of words grouped together based on the syntactic and semantic rule of the language in which the document was written. The paragraphs are supposed to contain a single message and the coherence between the messages in the paragraphs explains the themes of the document. The knowledge of document themes could be explored for document classification, document clustering and large document browsing [4], [13], [14].

Understanding and deriving document themes is fairly easy but could be time-consuming for humans especially when dealing with a large collection of documents.

Therefore, there is a need to automatically and efficiently extract document thematic content. This could be achieved by observing content within a document and document collections. The observed data in a textual document are the words and sometimes, the order of these words. As will be seen later in this review, some algorithms also observe document corpus (a collection of documents).

Earlier attempts at automatically determining the thematic structure of document employ document classification and document clustering algorithms [8], [15]–[20]. Document clustering and document classification usually involve the use of unsupervised and supervised machine learning algorithms respectively, to group documents into different categories based on the content. Document classification assigns documents to a set of pre-defined classes such that each document belongs to one and only one class, and it employs supervised machine learning algorithm. In the same way, document clustering algorithms assigns documents to a class based on the document content. However, some document clustering algorithms can dynamically specify the number of classes (clusters) and are based on unsupervised machine learning algorithm [16].

When the classes in document classification algorithm as well as clusters in document clustering algorithm are considered as themes of the document, then both text classification and document clustering algorithms assume that each document contains only one theme such that classification algorithms have a fixed set of themes that could be contained in a document within a corpus [3]. Text classification algorithm assumes a fixed set of predefined themes while text clustering assumes a variable number of themes.

These assumptions and representations have a number of limitations. From a human perspective a document can contain more than one theme. The one-theme-per-document model assumed by document clustering and classification algorithm presents an incomplete representation of document thematic structure [3]. Therefore, there is a need for a model with better representation of document themes.

Topic model provides another representation for understanding document thematic structure. Unlike models generated through document classification and document clustering algorithms, topic model assumes a document can contain more than one theme and in some cases these themes can be viewed in a hierarchical structure, where the theme above the structure is made up of the ones below it [21].

Unlike document clustering and document classification algorithm, topic modeling algorithms generate a topic model which assigns an association value of document to each topic. Therefore, topic models present a better model for document representation than document classification and clustering algorithms. The set of algorithms used for generating topic models are known as topic modeling algorithms.

3. Topic Modeling Algorithms

Topic modeling algorithms are a set of algorithms used to discover the thematic structure that exists within a large collection of documents such that the document can be further arranged or processed according to the themes discovered [14]. Unlike document clustering and classification algorithms, topic modeling algorithm can

discover more than one theme in a document. These themes are called topics. Topics discovered by these algorithms are group of words where each word has a degree of association with the topics. Meaning that a word can belong to more than one topic with different degree of association to each topic. In the statistical sense, topics can be seen as a distribution of words.

As mentioned in the previous section, the observable data for topic modeling algorithms are documents and the document contains a syntactically and semantically ordered collection of words. In most topic modeling algorithms, the order of the words is not considered and the topics are treated as a bag-of-words representation, thereby throwing away the order of such words. The order of words in a document is a syntactic representation of the language and the semantic representation is a function of word relationship [22]. Since the theme in the document is a semantic notion [4], it implies that the bag-of-word representation in topic modeling can represent the thematic content of the document. Therefore, topic modeling algorithms represent the thematic structure of a document as a probabilistic distribution of words.

Consequently, algorithms for generating topic models either use methods and techniques from linear algebra [23]–[26] or use methods and techniques from statistics and probability [27]–[29]. Topic models generated using probability and statistical theory usually have a probabilistic word or phrase embedded in the name of the algorithm [3], [30], [31].

Observation of linear-algebra-based topic modeling algorithms show that even though such algorithms employ linear algebra methods in their design, however, topic models generated using such algorithms have been seen to have a probabilistic interpretation [24]. Consequently, from the probabilistic perspective, topic models are mixture model of word-topic, topic-document relationship/distribution [27].

3.1 Linear Algebra Topic Modeling algorithms

3.1.1 Latent Semantic Analysis

Latent Semantic Analysis (LSA) was originally a document indexing method for information retrieval where it was popularly called Latent Semantic Analysis (LSI). It was proposed in [23] and it helps to solve the problem of synonymy in previous information retrieval indexing methods [32]. Prior to LSA, retrieval and document indexing methods use term-document matrices that performed keyword-based matching. Keyword-based matching ignores documents containing synonymous words with that of the query. LSA on the other hand, generates a model that has a similar representation for synonymous words, therefore, retrieving a more complete collection of documents.

Fundamentally, LSA uses term-document matrix such that the rows of the matrix correspond to the words and the columns correspond to the documents. Each cell in the matrix is a count of the number of times words at the row appear in the document that correspond to the column. Thus, the rows become a vector representation of the words relative to the document collection. This representation thus becomes a matrix

representation of the entire corpus. The term-document matrix generated is factorized using Singular Value Decomposition (SVD) as shown in equation 1:

$$X = T_o S_o D_o' \qquad \text{equation (1)}$$

X is the rectangular term-document matrix generated from the collection of documents. T_o, S_o and D_o are the factors of the original matrix such that T and S are orthonormal matrices and D is a diagonal matrix. A close approximation of the original matrix is derived from the equation by selecting the first k values from the three matrices. The final values derived based on the value of k creates a low-rank version of the original matrix.

The compressed matrix thus contains representation of the original corpus. This representation allows various queries and operations to be performed on the model. For instance, the similarity between two terms can be calculated by calculating the dot product of the vectors representing the two terms in the compressed matrix. The operation $\hat{X}\hat{X}'$produces the similarities between all the terms in the document, and $\hat{X}$is the matrix derived using SVD and $\hat{X}'$is it transpose.

The similarity between all documents is calculated as $\hat{X}'\hat{X}$and the corresponding cell of the result can be looked up for the similarity between two documents. Likewise, the relationship between a term and a document can be derived by performing $\hat{X}'\hat{X}'$then look up the corresponding cell from the result.

3.1.2 Non-negative Matrix Factorization (NMF)

In the previous section, the factorization of the original term-document matrix into three different matrices of low ranks is known as low rank matrix approximation [32]. Generally, low rank matrix approximation involves separating a matrix into two or more matrices whose multiplication approximate the value of the original matrix. One of the motivations for low rank matrix factorization is to allow fast computation of item similarities. For topic modeling, the motivation is to allow fast estimation of document-term relationships as seen in the previous section.

One other family of low rank matrix approximation methods is constrained based matrix approximation method [33]. SVD as used in the previous section can be seen as a constraint based matrix approximate method. The constraint in SVD is based on the least square error. The method tries to minimize the expression $|| M-UV||^2$ where M is the original matrix and U and V are low-rank factors of matrix M.

Constraint could also be set on the type of matrix to be generated, such is seen when a matrix is constrained to only have positive values. The application of such non-negative constraint in low-rank matrix factorization is known as non-negative low rank matrix approximation, or generally as Non-negative Matrix Factorization (NMF) [33]–[35].

The application of NMF in topic modeling was motivated by the work done in [28]. The authors applied NMF to a matrix of data where each column contains a human face. The author discovered that the result matrix U is a sparse matrix that represents the parts of human faces. They also applied NMF to a term-document matrix and discovered that matrix U represents a topical grouping of words in the document. This discovery spawns various interests in NMF by the research community in image analysis, text mining and spectral analysis [34], [35], [37].

One of the important aspects of NMF is that it produces output that represents identifiable parts of the original data. For example, in the experiment performed in [28], the output shows identifiable section of human face. This property makes NMF a suitable algorithm for representational learning [35]. Unlike SVD and Vector Quantization methods, models learned from NMF have intuitive representations, and it has been shown to extract more coherent topics than traditional algorithms like LDA [35].

Although NMF presents a lot of promises for topic modeling, it is however computationally intractable; directly solving NMF is NP-hard. However, different algorithms have been proposed to provide an approximate solution for solving NMF in polynomial time [26]. Various assumptions have been made to reduce the computation cost of computing NMF. For instance, the concept of document separability in topic modeling was assumed to provide a polynomial time algorithm for solving NMF [26].

Separability in a document means that each topic has an anchor word that only appears in one topic such that the presence of the word is enough to differentiate a topic. Topics generated using NMF with separability assumption are shown to exhibit topic correlation unlike traditional topic modeling algorithms like LDA [38].

Aside from the assumption of separability in [38], other algorithms exist that provide approximate solutions to NMF. For instance, [25] proposed defining a cost function such that the two factor matrices can be randomly initialized and then periodically updated until convergence using different update rules. A similar approach was also proposed in [37], such that the initialized matrix is updated with the non-zero constraint and any value less than zero is automatically updated to zero before performing the next update.

A general problem identified with NMF, aside from its computational complexity, is that the solution could be ill-posed [37], that is, it is possible to find different factors that combine to form the original matrix therefore, there can be more than one solution for the factorization problem and all except one will have a false representation of the document components.

Linear algebra models as described in this section maps a term-document matrix into a low dimensional space such that the resulting matrix is a compressed factor

representation of the original matrix. Therefore, linear algebra based topic modeling algorithms represent the thematic structure of a document as the position of this document in low rank dimensional planes. These planes represent topics and a document can intercept more than one plane.

3.2 Probabilistic Topic Modeling algorithm

The previous section described topic modeling algorithms with basis in linear algebra even though the algorithms have probabilistic interpretations. This section describes algorithms that are directly based on probabilistic models. This section will start with a probabilistic extension to LSA.

In [24], LSA was modeled using a generative approach by considering a corpus as a quadruple containing the universal terms U, the set of topics T, set of author styles S and a probability distribution D, that combines the other three variables. The quadruple is shown in equation 2:

$$C=(U,T,S,D) \qquad \text{equation (2)}$$

where C represents the corpus model.

The generative model given in [24] gives probabilistic insight into the inner workings of LSA. Also, [28] gave a probabilistic analysis that involves a form of mixture model to solve the limitation of LSA, thereby proposing Probabilistic Latent Semantic Analysis (PLSA) topic modeling algorithm.

3.2.1 Probabilistic Latent Semantic analysis

PLSA, also known as aspect model, presents a generative statistical modeling approach for understanding document components and provides a more explanatory model to topic modeling [28] by viewing the relationship between document and terms as a mixture model. The model was based on the joint probability between the document and the words that exist in the document since these two are easily observable from the documents. The joint probability is given in equation 3:

$$P(d,w)=P(d)P(w|d) \qquad \text{equation (3)}$$

where d represents the documents and w represents the term.

The joint probability model could be written to include the hidden variable as given below:

$$P(w_j|d_i) = \sum_{k=1}^{N} P\left(w_j|z_k\right)P(z_k|d_i) \qquad \text{equation (4)}$$

z is the hidden variable that represents the topic. Equation 4 is based on the assumption that the p and w are conditionally independent on the topic.

Consequently, equation 3 can be rewritten as equation 5:

$$P(d_i, w_j) = P(d_i) \sum_{k=1}^{N} P(w_j|z_k) P(z_k|d_i) \quad \text{equation(5)}$$

This representation in equation 5 enables PLSA to model polysemous words as PLSA has a similar geometric representation for words that have occurred in similar contexts. Inference in PLSA is achieved through Expectation-Maximization (EM) algorithm; the inference process overfits on the training data and therefore the model performs poorly when presented with unseen data [39].

Various methods have been proposed to overcome the overfitting problem of PLSA [39]–[42]. These methods include graph regularization and randomization techniques while others are based on modification of the generative process of PLSA. In [42], the overfitting problem of PLSA was assumed to be a result of the initialization methods and therefore proposed a conjugate prior method to better initialize the algorithm. The idea was that a well initialized PLSA could prevent the algorithm from getting stuck in a local minimum.

Likewise, a weighted incremental update was also proposed in [43]. The idea was to measure and adjust the impact of updates from previous data to the currently observed data. The intention is that new documents that have less relationship with the existing ones are treated specially since they might contain more information. There are also multi-modal variants of PLSA [44], and continuous PLSA. Continuous PLSA models topic as a continuous distribution on words through the use of Gaussian models [45]–[47].

3.2.2 Latent Dirichlet Allocation (LDA)

There are two major issues with PLSA, the first is the problem of overfitting which was mentioned in the previous section and the second problem is that PLSA has a problem representing unseen documents. This problem occurs because PLSA has no proper representation for documents as it does not properly observe the corpus from whence the document came. Therefore, documents in PLSA are just modeled as a list of numbers [29].

To solve the challenges posed by PLSA, [19] proposed LDA based on the exchangeability of words and documents. Unlike PLSA, LDA provides a model for the document and have the following generative process:

1. Choose the vocabulary size N from a Poisson distribution, *Poisson* (ξ)
2. Select Topic parameter θ from a Dirichlet distribution, *Dir*(α)
3. For each word w_n in N:
   i. Choose a topic z_n from a multinomial distribution, *Multinomial* (θ)
   ii. Choose a word w_n based on topic z_n

This generative model form a probabilistic graphical representation of word document and topic as shown in Figure 1: where M represents the total number of documents in the corpus and N is the vocabulary size, w represents words from the vocabulary, z is the hidden variable that represents the topic, β is a k-by-N dimensional matrix that represents the probability distribution of topic and words, θ| is a k-dimensional vector that parameterizes the topic and α is the Dirichlet distribution parameter such that $\alpha_i > 0$.

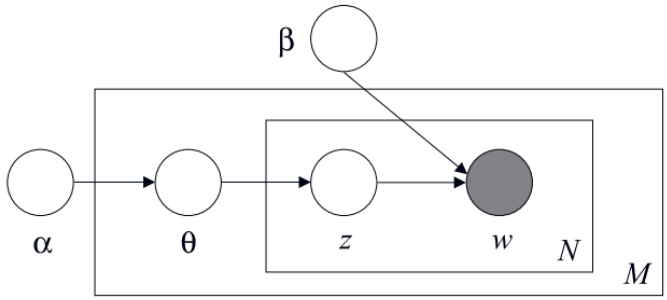

Figure 1. LDA graphical model representation [19]

The dimensionality of the Dirichlet parameter θ and the size of the topic k are decided before building the model. The key inferential problem is given in equation 6. This equation along with the graphical model given in Figure 1, imply that the exact inference and parameter estimation in LDA is intractable. The original paper thus proposed variational inference as the approximation algorithm for parameter inference.

$$P(\theta, z \mid w, \alpha, \beta) = \frac{P(\theta, z, w, \mid \alpha, \beta)}{P(w \mid \alpha, \beta)} \qquad \text{equation (6)}$$

The success of LDA has informed its application in text mining, image analysis and collaborative filtering [29], [48], [49]. Apart from variational inference algorithm, other inference algorithms have also been proposed for an improved parameter estimation in LDA [50], [51].

Probabilistic topic modeling algorithms give a thematic structure of documents as a probability distribution of words with different degrees of observations. PLSA observes words and the documents while LDA observes the words, document and corpus and therefore models the probability distribution of topics to document and not just the distribution of words to topic like in PLSA. This makes LDA a better representation of document thematic structure than PLSA.

Topic modeling algorithms like LSA, PLSA and LDA are standard traditional topic modeling algorithms that have seen applications in various areas [52]–[55]. However, these algorithms have been observed to be limited in some cases that will be discussed in the next section.

4. Beyond Traditional Topic Modeling Algorithms

Despite its successes, traditional topic modeling algorithms like LDA cannot efficiently model short text like posts on Twitter [12], [56]–[58]. LDA is limited when modeling documents with sparse matrix representation and therefore it is inefficient when modeling short text [57], [59]. Likewise, traditional LDA does not model topic correlation [38], [60], [61] and also, traditional topic modeling algorithms do not consider the evolution of topics over time [58], [62], [63].

LDA performs poorly with short text because the term-document matrix generated is generally sparse for short text and therefore fail to successfully model word correlation as a result of the sparsity [64]. To solve the problem of text sparsity, NMF was applied to the term-term correlation matrix in [64].

The term-term correlation is a measure of how related each term is to another. Since each term is represented in the matrix, the generated matrix is not sparse and the resulting non-sparse matrix is factorized using NMF. The generated topic model using this method was applied to different text-related tasks such as document clustering, document classification, etc, and was seen to have a better performance compared to traditional topic models like LDA [64].

Probabilistic NMF was also proposed in [65] and [66]. These models were derived through a semi-supervised approach by first converting the term document matrix into a probabilistic distribution through normalization and then applying NMF for topic factorization.

Topics are expected to be correlated. For example, a topic on sport should have some level of correlation with a topic on football, traditional topic modeling algorithms are unable to model this type of relationship between topics. LDA does not model the relationship or similarities among extracted topics. However, an approach known as *correlated topic model* (CTM) [61] employs LDA generation procedure while making use of logistic normal distribution in place of the Dirichlet distribution used in LDA. The logistic normal distribution uses a covariance matrix that measures the pairwise correlation between topics.

A more recent approach to solving the correlation problem of LDA aside from NMF that was discussed earlier, is the application of word embeddings to topic correlation. This is done by representing words in the document with their equivalent vector representations forming a correlated Gaussian Topic Model (CGTM) [67]. This approach and CTM only provide pair-wise correlation between topics and not a hierarchical relationship between topics. Another algorithm that models topic correlation alongside topic hierarchy is the *Pachinko Allocation Model* (PAM) [21], [68].

PAM is a topic modeling algorithm that employs Directed Acyclic Graph (DAG) to model topic correlation. In the graph, the leaf node represents the vocabulary and other layers represent the topics at different levels. Each level, aside the leaf node, treats the topics as a Dirichlet distribution such that the traditional LDA can be modelled as PAM with three levels where the leaf nodes represent the vocabulary; the middle layer represents topics and the root represents the overall topical distribution.

The superiority of PAM to other hierarchical topic models like Hierarchical LDA in [69] is that PAM presents a more granular and sparse representation of topic correlation [21] as shown in Figure 2(c).

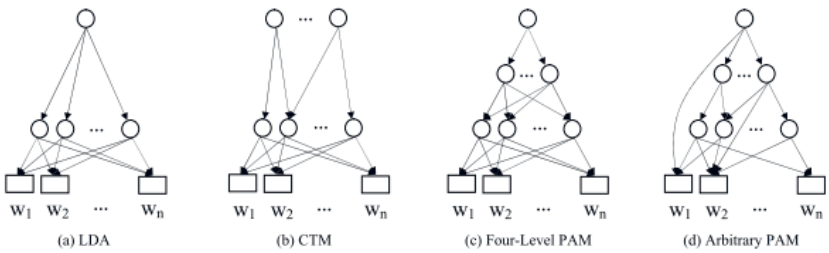

Figure 2. Comparing Pachinko with CTM and LDA [63]

PAM graphical structure can have arbitrary levels as shown in Figure 2(d), However, the original paper proposed a four-level PAM model as shown in Figure 2(c). Documents are generated from PAM by first sampling the Dirichlet parameter from the root, then sampling the path to the leaf node that contains the word. The middle levels in PAM have representations according to LDA, thus the graphical structure and the levels in the graph increases the number of parameters and complexity of variational inference algorithm on PAM therefore, Gibbs sampling was employed as the inference method in the original paper. Also, methods from hierarchical LDA were proposed for a more hierarchy-oriented PAM [21]. A non-parametric prior for improving topic correlation was also proposed in [70] for PAM. Also, a variational-inference based parameter estimation was proposed in [71] to reduce the computational complexity in PAM inference.

PAM and other correlated topic modeling algorithms, like LDA models topics as probability distribution of words and documents. Unlike LDA, the relationship between each topic could be measured using some form of distance measures. In hierarchical topic modeling algorithms like PAM, the topics are explained by other topics at a lower hierarchical level. This representation increases the possibility of deriving more useful insights from a collection of documents.

In CTM, the observed variables are documents and corpus vocabulary, a variable that was not observed in this model is the timestamp on the document. Since topics are distribution of words and the meaning of a word can change over time [72], [73], modeling topics without observing document timestamp will generate an incomplete representation of the topics and document distribution.

Historical documents such as news articles intrinsically include document timestamp, and a proper analysis of this type of document while observing document timestamp could provide insights on historical events and various scientific trends and evolution [74]. Therefore, finding answers to the question of how to include document timestamp as an observed variable to derive time sensitive topic models is the

problem being solved by a class of topic modeling algorithms called *Dynamic Topic Modeling* Algorithm (DTM) [63], [72].

Various DTM algorithms exists [63], [72], [75]–[77]. An early DTM algorithm proposed in [72] uses LDA-like generative process while exploring state space model, Kalman filter, to model the evolutionary changes of topics. This approach is parameterized and requires that the time be discretized. The challenge with discretized time is knowing the perfect bin size for the time. The approach raises questions like, should the time bin be yearly, monthly or quarterly. A continuous time DTM was proposed in [63] for solving the discretization problem.

Parameterized DTM algorithms require that the size of the topic be fixed. This model is limited as it cannot model the death and birth of topic. Non-parametric DTM algorithms exist to solve this problem [74], [78], [79]. The non-parametric DTM algorithms can model a variable number of topics at different times and can also model the birth and death of topics overtime.

5. Conclusion

Topic modeling algorithms generate topic models that help understand the thematic structures of large unstructured data. This paper has reviewed and classified topic modeling algorithms into two main categories: linear-algebra-based topic modeling algorithm and probabilistic topic modeling algorithm. Linear Algebra model are LSA and NMF while the probabilistic algorithms are PLSA and LDA. These traditional topic modeling algorithms generate models that represent topics as a probability distribution of words while only observing the document collections and the corpus vocabulary.

Other algorithms such as Short-text topic models, CTM, PAM and DTM extends the model presented by the traditional topic modeling algorithm. Short-text topic modeling algorithms extends traditional models capability to model short text like Twitter feeds. Models from CTM and PAM extend traditional topic models to explain topic correlation and topic hierarchy while models from DTM algorithms extends traditional topic model by observing document timestamp to model topic evolution.

According to the reviewed algorithms it can be said that the thematic structure of documents is a collection of word and document distribution derived while observing document collections with timestamp such that the derived distributions maintain some form of relationships.

References

[1] Y. Zuo, J. Zhao, and K. Xu, “Word network topic model: a simple but general solution for short and imbalanced texts,” Knowl. Inf. Syst., vol. 48, no. 2, 2016, doi: 10.1007/s10115-015-0882-z.

[2] B. Jeong, J. Yoon, and J. M. Lee, “Social media mining for product planning: A product opportunity mining approach based on topic modeling

and sentiment analysis," Int. J. Inf. Manage., vol. 48, no. October, pp. 280–290, 2019, doi: 10.1016/j.ijinfomgt.2017.09.009.

[3] D. M. Blei, A. Y. Ng, and M. T. Jordan, "Latent dirichlet allocation," in Advances in Neural Information Processing Systems, 2002, no. January.

[4] U. Kulatunga, D. Amaratunga, and R. Haigh, "Structuring the unstructured data: the use of content analysis," 2007.

[5] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: LIWC and computerized text analysis methods," Journal of Language and Social Psychology, vol. 29, no. 1. pp. 24–54, 2010, doi: 10.1177/0261927X09351676.

[6] R. Feldman and J. Sanger, The Text Mining Handbook. 2006.

[7] M. Allahyari, E. D. Trippe, and J. B. Gutierrez, "A Brief Survey of Text Mining : Classification , Clustering and Extraction Techniques," 2017.

[8] A. Kumar, V. Dabas, and P. Hooda, "Text classification algorithms for mining unstructured data: a SWOT analysis," Int. J. Inf. Technol., vol. 12, no. 4, pp. 1159–1169, 2020, doi: 10.1007/s41870-017-0072-1.

[9] J. D. Lee, K. Kolodge, and J. D. Power, "Exploring Trust in Self-Driving Vehicles Through Text Analysis," 2019, doi: 10.1177/0018720819872672.

[10] L. Celardo and M. G. Everett, "Network text analysis: A two-way classification approach," Int. J. Inf. Manage., vol. 51, no. September, 2020, doi: 10.1016/j.ijinfomgt.2019.09.005.

[11] A. Humphreys and R. J. Wang, "Automated Text Analysis for Consumer Research," vol. 44, no. June, pp. 1274–1306, 2018, doi: 10.1093/jcr/ucx104.

[12] X. Cheng, X. Yan, Y. Lan, and J. Guo, "BTM: Topic modeling over short texts," IEEE Trans. Knowl. Data Eng., vol. 26, no. 12, pp. 2928–2941, 2014, doi: 10.1109/TKDE.2014.2313872.

[13] A. J. B. Chaney and D. M. Blei, "Visualizing topic models," ICWSM 2012 - Proc. 6th Int. AAAI Conf. Weblogs Soc. Media, pp. 419–422, 2012.

[14] D. M. Blei, "Probabilistic Topic Models," Commun. ACM, vol. 55, no. 4, pp. 77–84, 2012.

[15] P. Bafna, D. Pramod, and A. Vaidya, "Document clustering: TF-IDF approach," Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016, no. March 2016, pp. 61–66, 2016, doi: 10.1109/ICEEOT.2016.7754750.

[16] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey, "Scatter/Gather," ACM SIGIR Forum, vol. 51, no. 2, pp. 148–159, 2017, doi: 10.1145/3130348.3130362.

[17] R. Raina, Y. Shen, A. Y. Ng, and A. McCallum, "Classification with hybrid generative/discriminative models," Adv. Neural Inf. Process. Syst., 2004.

[18] K. Tanaka-Ishii, S. Tezuka, and H. Terada, "Sorting Texts by Readability," Comput. Linguist., vol. 36, no. 2, pp. 203–227, 2010, doi: 10.1162/coli.2010.09-036-R2-08-050.

[19] T. Luong, T. Tran, and Q. Truong, "Learning to Filter User Explicit Intents," vol. 2, no. 2006, pp. 13–24, 2016, doi: 10.1007/978-3-662-49390-8.

[20] P. C. Science, A. Sboev, T. Litvinova, D. Gudovskikh, R. Rybka, and I. Moloshnikov, "Machine Learning Models of Text Categorization by Author Gender Using Topic-Independent Features," Procedia - Procedia Comput. Sci., vol. 101, pp. 135–142, 2016, doi: 10.1016/j.procs.2016.11.017.

[21] D. Mimno, W. Li, and A. McCallum, "Mixtures of hierarchical topics with Pachinko allocation," ACM Int. Conf. Proceeding Ser., vol. 227, pp. 633–640, 2007, doi: 10.1145/1273496.1273576.

[22] T. K. Landauer, "Latent Semantic Analysis," in Encyclopedia of Cognitive Science, 2006, pp. 1–14.

[23] H. Deerwester, Scott Susan, T. Dumais George , W. Furnas Thomas, K. Landauer Richard, "Indexing by latent semantic analysis," J. Am. Soc. Inf. Sci., vol. 41, no. 6, 1990.

[24] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala, "Latent semantic indexing: A probabilistic analysis," J. Comput. Syst. Sci., vol. 61, no. 2, pp. 217–235, 2000, doi: 10.1006/jcss.2000.1711.

[25] D. D. Lee and H. Sebastian Seung, "Algorithms for Non-negative Matrix Factorization," New Math. Nat. Comput., vol. 11, no. 2, pp. 121–133, 2015, doi: 10.1142/S1793005715400013.

[26] R. Albalawi, T. H. Yeap, and M. Benyoucef, "Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis," Front. Artif. Intell., vol. 3, no. July, pp. 1–14, 2020, doi: 10.3389/frai.2020.00042.

[27] M. Steyvers and T. Griffiths, "Probabilistic Topic Models," Handb. latent Semant. Anal., vol. 427, no. 7, pp. 424–440, 2007, doi: 10.1016/S0364-0213(01)00040-4.

[28] T. Hofmann, "Probabilistic Latent Semantic Analysis," arXiv Prepr. arxiv, 2013, doi: 10.1.1.33.1187.

[29] D. M. Blei, A. Ng, and M. Jordan, "Latent Dirichlet Allocation," J. Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.

[30] R. Alghamdi and K. Alfalqi, "A Survey of Topic Modeling in Text Mining," Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 1, pp. 147–153, 2015, doi: 10.14569/ijacsa.2015.060121.

[31] T. Hofmann, "Unsupervised learning by probabilistic Latent Semantic Analysis," Mach. Learn., vol. 42, no. 1–2, pp. 177–196, 2001, doi: 10.1023/A:1007617005950.

[32] C. D. Manning, R. Prabhakar, and S. Hinrich, Introduction to Information Retrieval (2nd edition), vol. 53, no. 9. 2009.

[33] N. Gillis, "Introduction to Nonnegative Matrix Factorization," pp. 1–18, 2017, [Online]. Available: http://arxiv.org/abs/1703.00663.

[34] S. Arora, R. Ge, R. Kannan, and A. Moitra, "Computing a nonnegative matrix factorization-provably," SIAM J. Comput., vol. 45, no. 4, pp. 1582–1611, 2016, doi: 10.1137/130913869.

[35] X. Fu, K. Huang, N. D. Sidiropoulos, and W. K. Ma, "Nonnegative Matrix Factorization for Signal and Data Analytics: Identifiability, Algorithms, and Applications," IEEE Signal Process. Mag., vol. 36, no. 2, pp. 59–80, 2019, doi: 10.1109/MSP.2018.2877582.

[36] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," Nature, vol. 401, no. 6755, pp. 788–791, 1999, doi: 10.1038/44565.

[37] N. Gillis, "The Why and How of Nonnegative Matrix Factorization," in Regularization, Optimization, Kernels, and Support Vector Machines, 2020, pp. 275–310.

[38] S. Arora, R. Ge, and A. Moitra, "Learning topic models - Going beyond SVD," Proc. - Annu. IEEE Symp. Found. Comput. Sci. FOCS, pp. 1–10, 2012, doi: 10.1109/FOCS.2012.49.

[39] E. Rodner and J. Denzler, "Randomized probabilistic latent semantic analysis for scene recognition," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2009, vol. 5856 LNCS, pp. 945–953, doi: 10.1007/978-3-642-10268-4_110.

[40] X. Wang, M. C. Chang, L. Wang, and S. Lyu, "Efficient algorithms for graph regularized PLSA for probabilistic topic modeling," Pattern Recognit., vol. 86, pp. 236–247, 2019, doi: 10.1016/j.patcog.2018.09.004.

[41] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," J. Mach. Learn. Res., vol. 3, no. 4–5, pp. 993–1022, 2003, doi: 10.1162/jmlr.2003.3.4-5.993.

[42] M. Klasson, S. I. Adalbjörnsson, J. Swärd, and S. V. Andersen, "Conjugate-prior-regularized multinomial pLSA for collaborative

filtering," 25th Eur. Signal Process. Conf. EUSIPCO 2017, vol. 2017-Janua, pp. 2501–2505, 2017, doi: 10.23919/EUSIPCO.2017.8081661.

[43] N. Li, W. Luo, K. Yang, F. Zhuang, Q. He, and Z. Shi, "Self-organizing weighted incremental probabilistic latent semantic analysis," Int. J. Mach. Learn. Cybern., vol. 9, no. 12, pp. 1987–1998, 2018, doi: 10.1007/s13042-017-0681-9.

[44] R. Lienhart, S. Romberg, and E. Hörster, "Multilayer pLSA for multimodal image retrieval," CIVR 2009 - Proc. ACM Int. Conf. Image Video Retr., pp. 60–67, 2009, doi: 10.1145/1646396.1646408.

[45] Z. Li, Z. Shi, X. Liu, and Z. Shi, "Automatic image annotation with continuous PLSA," ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., pp. 806–809, 2010, doi: 10.1109/ICASSP.2010.5494943.

[46] H. Zhang, R. Edwards, and L. Parker, "Regularized probabilistic latent semantic analysis with continuous observations," Proc. - 2012 11th Int. Conf. Mach. Learn. Appl. ICMLA 2012, vol. 1, pp. 560–563, 2012, doi: 10.1109/ICMLA.2012.102.

[47] S. O. Ba, "Discovering Topics With Neural Topic Models Built From Plsa Assumptions," arXiv, 2019.

[48] D. M. Blei and M. I. Jordan, "Modeling Annotated Data," SIGIR Forum (ACM Spec. Interes. Gr. Inf. Retrieval), no. SPEC. ISS., pp. 127–134, 2003, doi: 10.1145/860458.860460.

[49] C. Chen, A. Zare, and J. T. Cobb, "Partial Membership Latent Dirichlet Allocation for Image Segmentation," pp. 2368–2373, 2016.

[50] A. Anandkumar, D. P. Foster, D. Hsu, S. M. Kakade, and Y. K. Liu, "A Spectral Algorithm for Latent Dirichlet Allocation," Algorithmica, vol. 72, no. 1, pp. 193–214, 2015, doi: 10.1007/s00453-014-9909-1.

[51] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," EMNLP 2011 - Conf. Empir. Methods Nat. Lang. Process. Proc. Conf., no. 2, pp. 262–272, 2011.

[52] R. Alghamdi and K. Alfalqi, "A Survey of Topic Modeling in Text Mining: LSA, LDA topic modelling and topic evolution model," IJACSA) Int. J. Adv. Comput. Sci. Appl., 2015, doi: 10.14569/IJACSA.2015.060121.

[53] H. Misra, F. Yvon, J. M. Jose, and O. Cappé, "Text segmentation via topic modeling: An analytical study," Int. Conf. Inf. Knowl. Manag. Proc., pp. 1553–1556, 2009, doi: 10.1145/1645953.1646170.

[54] L. Liu, L. Tang, W. Dong, S. Yao, and W. Zhou, "An overview of topic modeling and its current applications in bioinformatics," SpringerPlus, vol. 5, no. 1. 2016, doi: 10.1186/s40064-016-3252-8.

[55] N. Prollochs and S. Feuerriegel, “Business analytics for strategic management: Identifying and assessing corporate challenges via topic modeling,” Inf. Manag. 57(, vol. 15, 2020, doi: 10.3929/ethz-a-010782581.

[56] X. Quan, C. Kit, Y. Ge, and S. J. Pan, “Short and sparse text topic modeling via self-aggregation,” IJCAI Int. Jt. Conf. Artif. Intell., vol. 2015-Janua, no. Ijcai, pp. 2270–2276, 2015.

[57] Y. Zuo et al., “Topic modeling of short texts: A pseudo-document view,” Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 13-17-Augu, pp. 2105–2114, 2016, doi: 10.1145/2939672.2939880.

[58] K. Ghoorchian and M. Sahlgren, “GDTM: Graph-based Dynamic Topic Models,” Prog. Artif. Intell., 2020, doi: 10.1007/s13748-020-00206-2.

[59] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma, “Topic modeling for short texts with auxiliary word embeddings,” SIGIR 2016 - Proc. 39th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr., pp. 165–174, 2016, doi: 10.1145/2911451.2911499.

[60] W. Li and A. McCallum, “Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations,” in International Conference on Machine Learning, Pittsburgh, 2006, vol. 224, no. 23, pp. 231–238.

[61] D. M. Blei and J. D. Lafferty, “Correlated topic models,” Adv. Neural Inf. Process. Syst., pp. 147–154, 2005.

[62] A. B. Dieng, F. J. R. Ruiz, and D. M. Blei, “The Dynamic Embedded Topic Model,” pp. 1–17, 2019, [Online]. Available: http://arxiv.org/abs/1907.05545.

[63] C. Wang, D. Blei, and D. Heckerman, “Continuous time dynamic topic models,” Proc. 24th Conf. Uncertain. Artif. Intell. UAI 2008, pp. 579–586, 2008.

[64] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang, “Learning topics in short texts by non-negative matrix factorization on term correlation matrix,” in Proceedings of the 2013 SIAM International Conference on Data Mining, SDM 2013, 2013, pp. 749–757, doi: 10.1137/1.9781611972832.83.

[65] K. MacMillan and J. D. Wilson, “Topic supervised non-negative matrix factorization,” pp. 1–20, 2017, [Online]. Available: http://arxiv.org/abs/1706.05084.

[66] M. Luo, F. Nie, X. Chang, Y. Yang, A. Hauptmann, and Q. Zheng, “Probabilistic non-negative matrix factorization and its robust extensions for topic modeling,” in 31st AAAI Conference on Artificial Intelligence, AAAI 2017, 2017, pp. 2308–2314.

[67] G. Xun, Y. Li, W. X. Zhao, J. Gao, and A. Zhang, “A Correlated Topic Model Using Word Embeddings,” Twenty-Sixth Int. Jt. Conf. Artif. Intell.,

vol. 22, no. 5, pp. 382–387, 2017, doi: 10.1097/00004424-198705000-00005.

[68] W. Li and A. McCallum, "Pachinko allocation: Scalable mixture models of topic correlations," J. Mach. Learn. Res., 2008, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.219.3333&rep=rep1&type=pdf.

[69] D. M. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum, "Hierarchical topic models and the nested Chinese restaurant process," Adv. Neural Inf. Process. Syst., vol. 16, no. 16, pp. 17–24, 2004.

[70] W. Li, D. Blei, and A. McCallum, "Nonparametric bayes pachinko allocation," Proc. 23rd Conf. Uncertain. Artif. Intell. UAI 2007, pp. 243–250, 2007.

[71] A. Srivastava and C. Sutton, "Variational Inference In Pachinko Allocation Machines," 2018, [Online]. Available: http://arxiv.org/abs/1804.07944.

[72] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in ACM International Conference Proceeding Series, 2006, vol. 148, pp. 113–120, doi: 10.1145/1143844.1143859.

[73] M. Rudolph and D. Blei, "Dynamic Embeddings for Language Evolution," vol. 2, pp. 1003–1011, 2018, doi: 10.1145/3178876.3185999.

[74] H. Liu, Z. Chen, J. Tang, Y. Zhou, and S. Liu, Mapping the technology evolution path: a novel model for dynamic topic detection and tracking, vol. 125, no. 3. Springer International Publishing, 2020.

[75] D. J. Arnold et al., "Dynamic Topic Modeling : Spatiotemporal Analysis of Los Angeles Twitter Data," p. 2.

[76] J. Sleeman, M. Halem, T. Finin, and M. Cane, "Discovering scientific influence using cross-domain dynamic topic modeling," Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017, vol. 2018, pp. 1325–1332, 2017, doi: 10.1109/BigData.2017.8258063.

[77] R. Hida, N. Takeishi, T. Yairi, and K. Hori, "Dynamic and static topic model for analyzing time-series document collections," ACL 2018 - 56th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap., vol. 2, pp. 516–520, 2018, doi: 10.18653/v1/p18-2082.

[78] A. Ahmed and E. P. Xing, "Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream," in Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence, UAI 2010, 2010, pp. 20–29.

[79] A. Dubey, A. Hefny, S. Williamson, and E. P. Xing, "A nonparametric mixture model for topic modeling over time," Proc. 2013 SIAM Int. Conf. Data Mining, SDM 2013, pp. 530–538, 2013, doi: 10.1137/1.9781611972832.59.