Journal of Computer Science

Review

Comparative Analysis of Topic Modeling on People Query-Based Data

Saranya M and Amutha B

Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

Article history
Received: 04-03-2024
Revised: 30-04-2024
Accepted: 07-05-2024

Corresponding Author:
Amutha B
Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chennai, India
Email: amuthab@srmist.edu.in

Abstract: As using the internet becomes more common in our daily lives, Perhaps greater numbers of individuals are buying things digitally. Specialized digital marketplaces for things like clothes and books have turned into megastores with many stores. This makes it harder to find what you're looking for and takes more time. The query's data content is estimated ahead of time from the query logs and usually includes one or more search terms. Classifying the changes that users make to the information in their query strings is one way to model how they search. The article explains how to use topic modeling to effectively pull out product behavior patterns from data. An effective and flexible topic modeling tool is used to create the final models. Lots of different models can be tested with this framework, including PLSA, LDA, PAM, NMF, LSA, and many more. The results show that the technique can gather data on the different ways that people use a certain product. In order to deal with this kind of problem, we were able to come up with a strong solution using topic modeling. Topic modeling clearly assisted with the categorization of the product review. PLSA does better than the topic models suggested by NMF and LDA, according to the results.

Keywords: Topic Model, Machine Learning, LDA, LSA, PLDA, PAM NMF, PLSA, Query, Text Analysis

Introduction

One of the most crucial aspects of data analytics is identifying the characteristics shared by many data sets. To do this, text analysis is typically used to identify the topics or occurrences being discussed in a document. While this information would make sense to a human reading a paper, a program is only provided the text as printed not the contents of every page. To do this, data scientists employ a software technique called topic modeling. One popular statistical method for removing latent variables from large datasets is topic modeling (Blei, 2012). It is particularly effective in text data analysis, but it has also been used in environmental, social, bioinformatics, and text data evaluations (Liu *et al*., 2016; Hong and Davison, 2010). A few examples of how this analysis may make large-scale datasets easier to access are grouping social media users based on post content, categorizing genetic data based on sequence structure, and classifying databases of journals and articles based on comparable topics. Topic modeling is widely used, but it has significant issues with noise sensitivity, stability, and optimization, all of which could lead to inaccurate results (Agrawal *et al*., 2018; Lafferty and Blei, 2005).

Another aspect of big data may have a broad range of consequences on social science, ranging from micro-level evaluations of daily interactions and interpersonal relationships to macro-level research on subjects like human behavior and social structure. Examples of assessing news (Chen *et al*., 2019), online reviews (Bi *et al*., 2019), and social media content (Yu and Egger, 2021), among other items, based on personal experiences and observable events, may be found in an expanding body of literature. Talks about social science, however, usually focus on the significant aspects of the discipline and infrequently address the useful uses of big data.

The big data discussion in social science usually revolves around the critical perspective of the field, despite the fact that the application itself is rarely investigated. Big data appears to hold great promise, yet it is always influenced by values and beliefs. Big data analysis is challenging since every step of the process depends on a number of variables, such as parameter choice, evaluation of incomplete results, and precise interpretations (Chen *et al*., 2014).

PLDA and NMF complement each other in data variability, dimensionality reduction, and interpretability, so this study used them. Classification, feature extraction, and data representation are likely research goals and both algorithms offer benefits. PLDA excels at classification tasks where class distinction is crucial. Because it models within-class and between-class variability, speaker verification and face recognition systems use it extensively. NMF's non-negativity makes it useful for feature extraction in image processing and text mining. Its parts-based representation is simple and useful in text analysis and bioinformatics. PLDA increases class separability by increasing the variance ratio between classes to within classes. PLDA can withstand class variations and is reliable in highly variable real-world applications because it uses a probabilistic data model. PLDA reduces data dimensionality, improving computational efficiency and high-dimensional data visualization. Use NMF for data types like pixel intensities that don't naturally contain negative values because it guarantees non-negative factors. Parts-based decomposition simplifies factor interpretation compared to PCA.NMF reduces noise like PLDA reduces data dimensionality, improving computational efficiency.

That being said, a network terminal like a computer or a smartphone is necessary to finish an online purchase. Product options are narrowed down and chosen from the list shown on the screen by entering search terms as a query. Online product searches can be more frustrating than conversing with a knowledgeable and accommodating salesperson because they often return too many or too few results. As a result, selecting appropriate search terms is a crucial component in influencing the behavior of virtual shoppers.

The information that goes with each word in a query is usually calculated ahead of time using the query log. Search terms with low information content are used a lot, while search terms with high information content are ones that were typed in by a person, like with a typo. Because there aren't many searches for things that aren't in malls, there are a lot of relevant results. People often type in a long query when they first start looking for something. As the search goes on, their questions become less specific. A query with low information content would be one where a lot of people enter the same words. No matter how advanced search has become, people who enter very specific queries are probably first-time visitors who don't know much about the mall or what it has to offer.

Text miners often use topic models, which are a type of math, to find and pull out conceptual ideas from text data. Image retrieval, text mining, and data mining are all things that this tool can be used for. But its main goal is to organize huge collections of texts well (Griffiths and Steyvers, 2004; Niebles *et al*., 2008; Hariri *et al*., 2012). Latent Drichlet Allocation (LDA), Non-negative Matrix Factorization (NMF), Pachinko Allocation Topic Model (PAM), and Probabilistic Latent Drichlet Allocation (PLSA) are some of the topic modeling methods that are used right now (Zhao *et al*., 2016). The document-topic and word-topic distributions are used by Latent Drichlet Allocation (LDA) to make generalization better. The NMF model breaks high-dimensional vectors down into their low-dimensional parts to make them easier to work with. The aspect model that PLSA creates is specifically made to make it better at making predictions. It achieves effective modeling and improves adaptability by making a unique set of variables for each topic.

The PLSA topic modeling method is used in this study to show a new way to look at data about online shopping. In this dataset, you can find product reviews from online stores, user queries, product descriptions, and People IDs. Two matrices are made when these documents are run through the PLSA, NMF, LDA, LSA, PLDA, and PAM algorithms. We look at the matrices and use data mining methods like hierarchical clustering to find patterns in the Product behavior sequences. The suggested Probabilistic Latent Semantic Analysis (PLSA) method, the Non-Negative Matrix Factorization (NMF) method, and the Latent Drichlet Allocation (LDA) method are being compared to find the best model. The clustering method worked better than other topic modeling algorithms when used with k-means on PLSA. The findings indicate that the PLSA topic modeling algorithm can help us figure out how the system's Product Search works when we look at data from online stores.

Literature Review

Topic modeling has recently made great strides thanks to the incorporation of deep learning, especially in the form of neural architectures and transformer-based models. When it comes to topic discovery, neural topic models like ProdLDA and the Neural Variational Document Model (NVDM) are superior because of their increased scalability and flexibility. More recent transformer-based models, like BERTopic and Topic BERT, take advantage of the contextual knowledge of older models, like BERT, to generate better, more substantive topics. To deal with changing datasets, new online topic models have also arisen, allowing for real-time topic tracking. Topics can be effectively identified with little data using zero-shot and few-shot topic modeling, which are supported by big pre-trained language models. Interactive and human-in-the-loop models have also recently emerged in the field; these models use user feedback to improve topics and multimodal topic models analyze data across various formats, including text and images. These innovations increase the precision and usefulness of topic modeling and open up new possibilities for its use in fields as diverse as social media analysis and biomedical research.

Probabilistic data analysis has become more common in the data mining industry over the last few years, according to references (Hidayat *et al*., 2015; You *et al*., 2022; Wang and McCallum, 2006). Though there are other methods, the topic modeling method is the best at finding hidden information in electronic archives. Used scientific articles to test how well Latent Drichlet Allocation (LDA) works for finding scientific subjects (Jo and Oh, 2011). There is a topic model (Wang and McCallum, 2006) that goes beyond LDA and looks at how the data changes over time and how it is structured. Later, more topic models were suggested to help solve document analysis problems in certain areas, such as geographic analysis (Jiang *et al*., 2013) and sentiment analysis (Jiang *et al*., 2013). Also, (Moe, 2003) showed a topic-concept cube that uses query logs to make it easier to shop online. A new probabilistic method was introduced by Younus *et al*. (2024) to show both spatiotemporal theme patterns and subtopic themes at the same time. New research on query log analysis has also looked at how temporal factors affect the results. Our research is the first that we know of that uses probabilistic topic modeling to look into a wide range of ideas about how query terms and URLs are related in a thorough way. The results of the experiment show that topic modeling is a great way to figure out what query logs really mean. When it comes to useful applications and quantitative metrics, it does better than a number of robust baseline techniques.

Research on user search behavior has focused on navigational aid applications (Broder, 2002) and personalized search result organizers (Jiang *et al*., 2018). Research into how people use search engines has yielded a wealth of useful information. Broder (Schellong *et al*., 2016) classified web search queries as either "navigational" (used to move directly to a specific website), "informational" (used to seek out general information), or "transactional" (used to complete a specific action, such as making a purchase or downloading an item).

In order to personalize messages and make site navigation easier for shoppers, e-commerce platforms provide valuable data on user behaviors (Albalawi *et al*., 2020). Different user behaviors in online shopping have been identified by additional research by Albalawi *et al*. (2020). These actions can be classified into four distinct patterns: Direct buying, where customers buy an item right away; browsing, where customers peruse the store's inventory and make mental notes about what they want to buy; searching, where customers actively seek out products to buy; and knowledge-building, where customers research the store's offerings. By applying an unsupervised clustering technique to real-world data, they were able to verify that their classification was effective. Session features derived from different action types, such as query search frequency, page views, and changes in item categories, are used as clustering in these analyses.

Researchers in the field of natural language processing work with a corpus of documents made up of word sequences, or tokens, as they will be called in the future. Documents of a similar semantic significance should be grouped together when handling large document collections. Topic modeling is the name given to this clustering method. It establishes a latent dimension of topics that provides a brief synopsis for every document in the set. Managing a dataset of consumer inquiries that characterize product behavior is part of online shopping. The date, the product code, and the total amount paid for that code are all included in each query. Thus, by employing a topic model to analyze a person's past Product Behaviour, we can learn more about their behavior. Our efforts would result in a hidden embedding space that faithfully captures the different types of consumption as determined by the statistical examination of the query data. By grouping them into different consumption categories and providing clear descriptions, the topics serve as a representation of the clients.

Topic Modeling Techniques

Every corpus document has roughly the same amount of words and every document is associated with one of the queries. The NMF, PLSA, and LDA topic models were implemented using the Mallet software (Zhao *et al*., 2016). We were able to model the corpus and derive query-specific topics and topic mixture distributions with the aid of these models (Anupriya and Karpagavalli, 2015). The LDA topic modeling was implemented by following the procedure outlined in reference (Blei *et al*., 2003).

Latent Drichlet Allocation

It is not uncommon to see generative probabilistic models such as Latent Drichlet Allocation (LDA). It is the simplest way to model topics. When it comes to capturing the interchangeability of words and documents, LDA is designed to be an improvement over its predecessors, PLSA and LSA. There are many different kinds of documents that contain data nowadays. Some examples are articles, webpages, blogs, social networks, and news. As a result, there is a growing need for an automated system that can sort, understand, and compress these document collections. Modern techniques for latent topic modeling extract themes from large datasets using an unsupervised approach (Porteous *et al*., 2008). According to LDA, each document covers a wide range of subjects (Deerwester *et al*., 1990). A topic is the collection of words that comprise it and the likelihood of a term appearing in that topic is defined as its vocabulary. Using nothing more than word count and subject statistics, it treats each document as a random assortment of words using a "bag of words" strategy. The fundamental idea behind LDA is that it should work in a manner analogous to writing. Put simply, it accepts a subject as input and

DOI: 10.3844/jcssp.2024.1734.1743

outputs a document on that same subject. It reveals the central theme within a dataset. The LDA model is illustrated graphically in Fig. (1). As a mixture of T latent topics, where each topic describes a multinomial distribution of D words, LDA represents each C document.

The generative process of the basic LDA looks like this: For every word in document *j* that is *Dj*. Pick a subject: Aij ~ Mult(δij). Choose a word Bij such that it is equal to Mult(βij) where the multinomial parameters for subjects in drichlet priors are applied to words in a topic αT and documents δj (Niebles *et al*., 2008).

Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a method in Natural Language Processing (NLP) that looks at how a document is related to the terms used in it. Analysis of group dynamics and document terminology is accomplished through the generation of a set of concepts pertinent to the documents and their contents. The original name of LSA was LSI, which stands for Latent Semantic Indexing. Compared to LSI, LSA improves the information retrieval task by making it more efficient. A vector-based representation of text is the main objective of LSA in order to produce semantic content. This is how it finds relevant texts and chooses the best heist words according to their similarity. This means that only the most pertinent documents are considered from a huge pool of results. By applying Singular Value Decomposition (SVD) (Lafferty and Blei, 2005), the dimensionality of the term-document matrix is decreased. In response to the word frequencies found in individual documents, LSA makes adjustments to its many features, such as weighted keyword matching and vector representation. In Latent Semantic Analysis, SVD is used to reorganize data. The singular value decomposition technique (SVD) factors out the real term-document matrix M to reduce its dimensionality (Chen *et al*., 2019). M is equal to T multiplied by S multiplied by DT and in SVD, M is divided into three matrices.

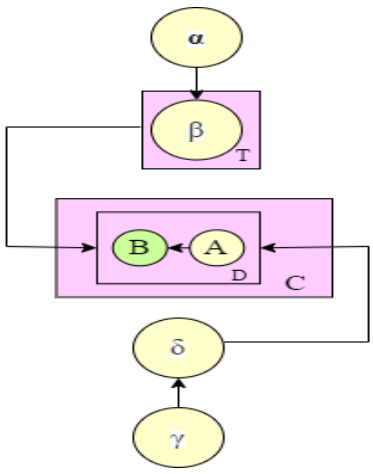

Fig. 1: Graphical representation of latent drichlet allocation

Parallel Latent Drichlet Allocation

A variation on the classic LDA model called Parallel Latent Drichlet Allocation (PLDA) uses parallel computer systems to speed up the topic modeling process. PLDA was created in order to get around LDA's computational constraints while working with big datasets. Topic inference is enhanced by the application of parallelization. PLDA uses several processors or cores sharing calculations to speed up the training process, allowing for faster topic modeling on large text document collections. Due to the significant time savings that this parallelization technique provides, PLDA has become an essential tool for researchers and practitioners working with large volumes of textual data in a variety of fields, such as information retrieval and natural language processing. K-means clustering is a well-liked unsupervised machine learning technique in natural language processing (NLP) that may be applied to a range of tasks involving the arrangement and examination of textual data. In NLP applications, K-means clustering comprises the following: Document Grouping: Using the K-means text mining method, documents can be sorted according to how similar their contents are.

Pachinko Allocation Topic Model (PAM)

An unsupervised hierarchical topic modeling algorithm for identifying topic correlations is the Pachinko Allocation Topic Model (PAM). Directed acyclic graphs (DAGs) are used to depict the hierarchy in this model. The subject is represented by the root and interior nodes of a DAG, while the individual words in the vocabulary are represented at the leaf level. The distribution of words and other topics are both represented by this structure. The tree structure of hLDA requires all nodes to be connected, but DAG allows for sparse connections, making it more flexible. The purpose of this algorithm is to find out how the document topics are related to each other. Finding the ideal number of topics '*i*' is a limitation of both the PAM algorithm and LDA. Label distribution over vocabulary is not a valid representation of the PAM algorithm.

Non-Negative Matrix Factorization

Mathematical models for decomposing high-dimensional vectors into low-dimensional spaces are Non-Negative Matrix Factorization (NMF or NMF). NMF reduces vectors into lower dimension (Dillon, 1983) non-negative components (*Topic Modeling with LSA, PSLA, LDA & lda2Vec | NanoNets*. (n.d.)). Think of Picture A as a matrix that, multiplied by matrix Y, solves equation *A* = *XY*. The NMF's clusters process results in matrix *X* and matrix *Y* corresponding to the data in matrix *A*:

- A (Document-word matrix) is a representation of the input words found in particular documents. Topics

(or clusters) extracted from the texts make up X (Basis vectors)
- Each report's Y-coefficient matrix shows the weights of the participant's contributions to the various topics covered.

Through iterative updates, the *X* and *Y* values can be obtained by using the objective function's equation (1), such as an expectation-maximization (EM) algorithm, until convergence is reached:

$$\frac{1}{2}||A - XY|| = \sum_{y=1}^{n}(Ai - (XY)) \quad (1)$$

In this scenario, the Euclidean distance is employed to calculate the reconstruction error between *X* and the result of multiplying *X* and *Y*. Equation (2) demonstrates the guidelines for updating *T* and *M*, utilizing the objective function stated in Eq. (1):

$$Xi = \frac{Xi(AY)}{XYY} \quad (2)$$

As soon as the new values are found through parallel tasks, we use the new X and Y to recalculate the reconstruction error. This process is carried out again and again until convergence is reached. Non-negative Matrix Factorization (NMF) can be done with a Python program. A free piece of software from a library for machine learning called "Scikit-learn" has a graph that shows the PLSA topic model. In addition to making a Query-topic proportion matrix, it can perform nonnegative matrix factorization. The position and base matrices of each Product for each People Query are in an input matrix that this matrix is based on.

Probabilistic Latent Semantic Analysis

To address the issue of dimensionality reduction, probabilistic latent semantic analysis (PLSA) employs the probabilistic approach. After latent semantic analysis (LSA) (Alemayehu and Fang, 2024), the PLSA adds a probabilistic treatment of words and topics. For every pair of documents *i* and *w*, the document-term matrix entry is denoted by *P*(*i*,*c*) in this PLSA model. Furthermore, each document is composed of multiple topics and each topic is itself composed of a set of words. Figure (2) shows the PLSA topic model in its general form.

The PLSA model gives the following assumptions a probabilistic twist:

- The presence of topic *j* in document *i* is associated with the probability *X*(*j*|*i*)
- Word *z* has the probability *X* (*z*|*j*) to come from topic *j* in a given topic *j*

In a more formal sense, Eq. (2) represents the combined likelihood of a given document and word.

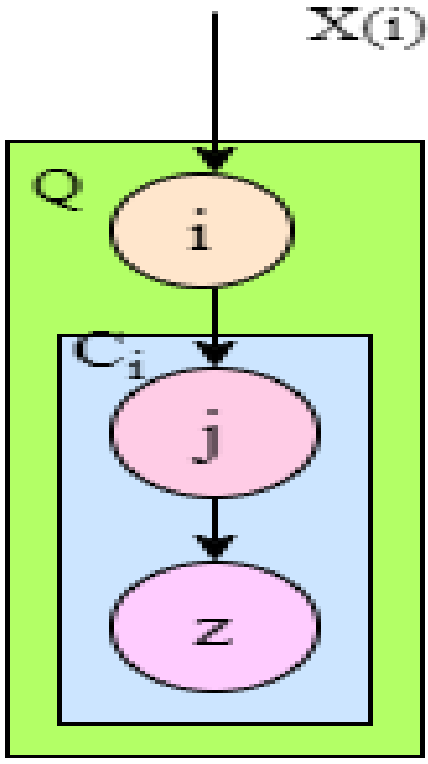

Fig. 2: Graphical representation of probabilistic latent semantic analysis

It is very helpful to use the Expectation-Maximization (EM) algorithm to train multinomial distributions like *X*(*j*|*i*) and *X*(*z*|*j*). It is possible to get a good idea of a model's parameters with the Expectation-Maximization (EM) algorithm. It turns out that the parameter count is equal to *ji* plus *zc*. How many parameters there are is directly related to how many documents there are. On top of that, PLSA is a computer model that can make documents. The PLSA algorithm, which can be found at https://github.com/laserwave/PLSA, was used to write the Python code. To do the EM calculation, 100 queries were run with the log-likelihood convergence threshold set to 1. The input, which was a grid with the position and base values of each Product for each Query, was used to make the query-topic proportion matrix.

Materials and Methods

Query Overview

For the sake of this analysis, we will assume that a mountain of query logs is provided. The query, which is a string of one or more words separated by spaces, along with the submission time and user ID makes up each query record. Because of this, we break down each query into its component words. When we get the query "best Quality," for instance, we pull out the words "best" and "Quality."

DOI: 10.3844/jcssp.2024.1734.1743

Our proposed method consists of the following steps, as illustrated in Fig. (3): Once the information content of the query has been established, you can proceed to extract the product's behavior, compute the information content of the query, and subsequently categorize the behavior of the interaction. The quantity of valuable information it encompasses serves as a gauge of its popularity. We observe elevated values when events have a low frequency and decreased values when they have a high frequency. Utilizing information content is a characteristic of user input. Attempt utilizing commonly used search terms that are frequently entered by individuals when searching for items in shopping malls.

On the other hand, a misspelled name or an attempt to find an item that isn't available at that specific shopping center could be the cause of unusual search terms. Another compelling reason to use multiple words in a query is the ability to add information content. A product behavior ID is used to make it easier to search for a specific item. The search logs are where these sessions are taken from. The steps to take in order to extract sessions are as follows: You have the option to search the organized logs, sort the classified results by date, and sort the search results by user ID from these two locations.

We consider two products to be identical if there is a time difference of less than sixty minutes between their IDs. They are considered different Product IDs if there is a time difference of more than sixty minutes. Table (1) displays a sample of a search product.

The amount of information in word C (zj) in query cx is what each query's information content is. The information in the *C* (*Qx*) query is shown by Eq. (3). Table (2) shows some of the data that can be found with a query:

$$C(Qx) = \sum_j (C(zi) \quad (3)$$

Proposed PLSA Model

The PLSA-obtained themes group words with similar meanings into categories. The graphical interpretation of the proposed model is shown in Fig. (4).

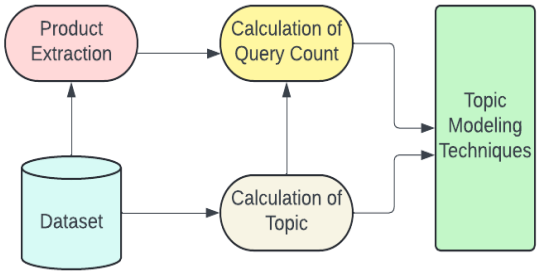

Fig. 3: Overview of product behavior based on query

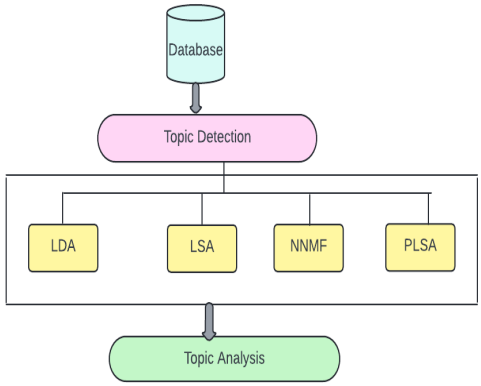

Fig. 4: Graphical interpretation of the proposed model

Table 1: Example of product behavior

People ID	Time (PM)	Query	Product ID
1001	08.07.2022 1.30	Good product	PI1101
1002	08.07.2022 2.30	Recommended	PI2345
1003	08.07.2022 3.35	Long-time use suitable	PI1245
1004	08.07.2022 4.29	Bad product	PI1145
1005	08.07.2022 5.30	Not recommended	PI1243
1006	08.07.2022 6.37	Good	PI1122
1007	08.07.2022 7.38	Money is not worth	PI1150

Table 2: Query information

People ID	Product ID	Query	C(Qx)
1001	PI1101	Good product	3.489
1002	PI2345	Recommended	4.678
1003	PI1245	Long-time use suitable	5.980
1004	PI1145	Bad product	4.908
1005	PI1243	Not recommended	6.768
1006	PI1122	Good	2.98
1007	PI1150	Money is not worth	7.568
People ID	Product ID	Query	C(Qx)
1001	PI1101	Good product	3.489
1002	PI2345	Recommended	4.678
1003	PI1245	Long-time use suitable	5.980
1004	PI1145	Bad Product	4.908
1005	PI1243	Not recommended	6.768
1006	PI1122	Good	2.98
1007	PI1150	Money is not worth	7.568

The ten most probable words from each of the five topics (T0, T1, T2, T3, and T4) that make up the PLSA topic model are shown in Table (3). The words were displayed in each category in order of their probability value, from most likely to least likely. The top ten words from each of the five subjects were absolutely astounding, and each subject has its own distinct word arrangement. There is a subject list and a probability comparison in the record for every query. Queries for the identical Product had comparable topic mixture coefficients. Utilized the gathered queries to ascertain the reasonable expectations for each Product. With this diverse range of subjects, we are able to tag products.

DOI: 10.3844/jcssp.2024.1734.1743

Table 3: Top 5 frequent words in product

Topic ID	Type	Top 5 most frequent words
T0	Product Description	Nestcam, nest learning, nest protect, security, thermostat, camera
T1	Product category	Nest USA, apparel, lifestyle, drinkware, notebooks and journals
T2	Product code	GGOENEBJ079499, GGOENEBQ079099, GGOENEBQ078999, GGOENEBQ079199, GGOENEBQ079099
T3	Coupon status	Used, not used, clicked, available, not available
T4	Product behavior or quality	Good, bad, recommended, not recommended, money is not worth

There is a distinct word order for each of the five subjects and the top ten words from each were absolutely astounding. Each query's record contains a set of topics and a probability comparison. When looking for the same Product, the topic mixture coefficients were quite close. Based on the questions asked, we were able to ascertain the realistic expectations for each Product. We use this diverse range of subjects to classify Products.

Experimental Setup

Data Preparation

Our industrial partner supplied the initial data, which is not publicly available. People ID, purchase date and time, product total, and product code are the fields that make up the client data. The gender and date of birth of individuals were also stored in separate tables. By establishing a hierarchy of Product codes, we grouped together Product codes that were similar and labeled them with descriptive terms like "Product Category," "Delivery Charge," and others to improve the data. There are two ways that we used to encode the Product behavior in each Product during preprocessing. To begin, we categorized each Product according to its behavior as a term frequency. When there is more money going into the product code, the product behavior in the user profile is more common. Because of this, there is a risk of distortion when an unusually high-priced purchase might be considered equivalent to numerous frequent low-priced product purchases. We added new tokens to our data "dictionary" to make up for this and improve it by dividing each Product code into quantiles. Each token represents a product's code for behavior and quantile of that code, which can be below average, average, or above average in terms of quality. As a last step, we expand our model's modalities to include the Product code hierarchy. Included in the initial hierarchy were the embedding used in this article solely containing product codes and small group modality. You can use other methods to make sure the topic model you have is sane. For example, you shouldn't put product codes from completely unrelated groups into the same topic. As an illustration, during training, a model that contains topics with codes from the "Average Price" and "Delivery Charge" groups would be rejected.

Experimental Results

We validate the capability of our approach to produce precise vector representations of online shoppers' data. Furthermore, we analyze the impact of data preprocessing on our work, in addition to our main objective. Crucial hyper parameters for training the standard topic model comprise the number of topics, the number of steps in the EM algorithm, and the regularization coefficients, which can be determined by assessing the model's coherence score, a metric known to be associated with interpretability. In order to assess the performance of different topic models, we establish the main hyper parameters, namely the number of topics and the number of steps in the EM algorithm. Subsequently, we allow users to adjust the regularization coefficients according to the metrics they prefer. Upon conducting a thorough analysis of the dataset, we have determined that the most optimal models are centered around 40 distinct topics. In order to maintain uniformity across this document, we made necessary modifications to this hyper parameter for all of the topic models. Table (4) illustrates the concept of interpretative topics by substituting Product tokens with clusters of topics that are associated with the Product Category. The probability of expenditures is not equal to one, as an inquisitive reader may observe. The absence of the main theme of the topic in this representation is due to a long tail in the distribution. By applying the given standard evaluation criteria for information retrieval, we redefine the problem as follows: Precision is calculated by dividing the number of relevant documents retrieved (PP) by the total number of documents retrieved (PNr). Recall is determined by dividing the number of relevant documents retrieved (PP) by the total number of relevant documents in the collection (NP). F-measure is calculated using the formula [2(precision)(recall)]/[(precision + recall)(PP)].

The next step is to use PLSA models that have had their topic model embedding adjusted on the training dataset to forecast how well products will behave. Table (5) displays the results of the performance tests conducted on different embedding.

Table 4: Product topic

Product category	Probability
Nest USA	0.567
Life Style	0.345
Waze	0.678
Headgear	0.234
Notebooks and Journal	0.098
Gender	Probability
Male	0.567
Female	0.789
Price	Probability
100–500	0.765
501–1000	0.678
1001–2000	0.546
2001–3000	0.986

Table 5: Model comparison with topic modeling techniques

Model type	Product accuracy	Product behavior accuracy	F1-score
LDA	0.987	0.789	0.679
LSA	0.890	0.678	0.589
NMF	0.765	0.543	0.512
PLDA	0.467	0.576	0.476
PAM	0.356	0.478	0.587
PLSA	0.632	0.347	0.314

Results and Discussion

PLSA topic modeling has been compared to other topic modeling methods like LDA and NMF to see how well it works and how accurate it is. The k-means algorithm is used on the Online Shopping dataset to do this. The *R* package was used to make the k-means function work. There were a total of ten runs of this algorithm.

The clustering performance evaluation results using the Topic modeling validation metrics are presented in Table (6). A higher validation value signifies superior clustering quality, with values ranging from 0 to 1. The PLSA and NMF topic models achieved their maximum values in measurements when the number of topics was set at 20. PLSA demonstrates superior performance compared to both LDA and NMF topic models in terms of similarity detection when analyzing online shopping data. Diagram 7 illustrates the different comparisons of topic modeling using accuracy criteria.

In this case, the research demonstrates that the LDA algorithm exhibits a decline in cluster quality measure as I increase from 5-10 and then an increase as I approach 15. However, the PLSA algorithm's clustering quality is getting better all the time.

When we compared the results of PLSA topic modeling to those of an existing LDA algorithm on data from online stores, we discovered that PLSA performed better for shown in Fig. (5). Therefore, when it comes to quality data analysis or product behavior, PLSA topic modeling is the way to go.

Table 6: Evaluating several topical models for varying topic values

Topic Modeling	i = 5	i = 10	i = 15	i = 20
K Means	0.4671	0.5874	0.6542	0.6789
LDA	0.4789	0.5987	0.5432	0.6543
LSA	0.5345	0.6598	0.6234	0.6458
NMF	0.6578	0.7432	0.6789	0.7345
PLSA	0.7896	0.7987	0.8234	0.8674

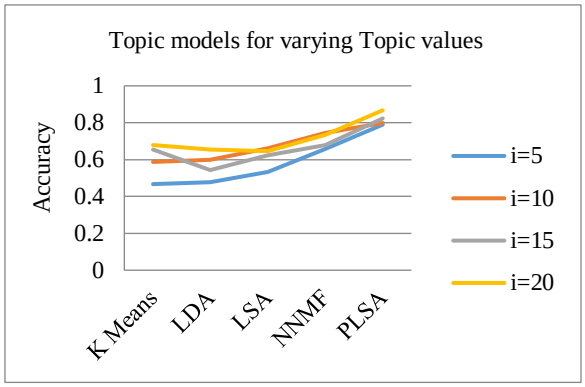

Fig. 5: Comparison of various topic modeling

Conclusion

In this study, we showcase an innovative approach to product behavior data analysis using PLSA topic models. Out of the four-step process, which starts with data collection, the last step is to evaluate the model. This method was also put to the test in conjunction with other cluster-based topic modeling algorithms like NMF and LDA. With an increase in the ask from 5-10, our comparison revealed that LDA reduced the quality of the Product cluster. Further, we discovered that PLSA outperformed the other topic modeling algorithms when it came to extracting information about product quality and behavior from e-commerce data. Even after getting it tuned up, the topic model couldn't beat the baseline for Product Behaviour-based categorized product codes—it just performed better than uncategorized data, though. Based on our analysis, it is possible that one of the two variables is to blame. We may find that the number of model topics we self-imposed becomes critical as the token dictionary grows as a result of our pre-processing. Additionally, when working with categorized data, the PLSA Model picked up on a few details that the model architecture might have missed. Our subsequent actions will focus on enhancing our model to handle the severe data imbalance that we discovered in our dataset and to account for the co-occurrence of online shopping. We will delve deeper into the reasons why users who initially use common queries to find products end up using rare queries to find what they want in future work. Similarly, we will examine why users who initially use rare queries to find products initially find their targets using common queries. Data contained in the query terms is the basis of our investigation. Additional important factors to consider are cluster queries and user query modification.

Acknowledgment

The authors would like to thank anonymous reviewers for their constructive comments and suggestions to update the manuscript.

Funding Information

This research received no external funding

Author's Contributions

Saranya M: Conceptualized the study, conducted the experiments and analyzed the results, prepared the initial manuscript, and contributed to the manuscript's editing and review.

Amutha B: Conceptualized the study, conducted the experiments, and contributed to the manuscript's editing and review.

Ethics

It has been testified by the authors that this article has not been submitted to be published in any other journal and contains no conflicts of interest or ethical issues.

References

Agrawal, A., Fu, W., & Menzies, T. (2018). What is wrong with topic modeling? And how to fix it using search-based software engineering. *Information and Software Technology*, *98*, 74–88. https://doi.org/10.1016/j.infsof.2018.02.005

Albalawi, R., Yeap, T. H., & Benyoucef, M. (2020). Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis. *Frontiers in Artificial Intelligence*, *3*, 42. https://doi.org/10.3389/frai.2020.00042

Alemayehu, E., & Fang, Y. (2024). Supervised probabilistic latent semantic analysis with applications to controversy analysis of legislative bills. *Intelligent Data Analysis*, *28*(1), 161–183. https://doi.org/10.3233/ida-227202

Anupriya, P., & Karpagavalli, S. (2015). LDA-based topic modeling of journal abstracts. *2015 International Conference on Advanced Computing and Communication Systems*. 2015 International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India. https://doi.org/10.1109/icaccs.2015.7324058

Bi, J.-W., Liu, Y., Fan, Z.-P., & Cambria, E. (2019). Modelling customer satisfaction from online reviews using ensemble neural network and effect-based Kano model. *International Journal of Production Research*, *57*(22), 7068–7088. https://doi.org/10.1080/00207543.2019.1574989

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77–84. https://doi.org/10.1145/2133806.2133826

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Broder, A. (2002). A taxonomy of web search. *SIGIR Forum*, *36*(2), 3–10. https://doi.org/10.1145/792550.792552

Chen, M., Mao, S., & Liu, Y. (2014). Big Data: A Survey. *Mobile Networks and Applications*, *19*(2), 171–209. https://doi.org/10.1007/s11036-013-0489-0

Chen, Y., Zhang, H., Liu, R., Ye, Z., & Lin, J. (2019). Experimental explorations on short text topic mining between LDA and NMF based Schemes. *Knowledge-Based Systems*, *163*, 1–13. https://doi.org/10.1016/j.knosys.2018.08.011

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*(6), 391–407. https://doi.org/10.1002/(sici)1097-4571(199009)41:6<391::aid-asi1>3.0.co;2-9

Dillon, M. (1983). Introduction to modern information retrieval. *Information Processing & Management*, *19*(6), 402–403. https://doi.org/10.1016/0306-4573(83)90062-6

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, *101*(suppl_1), 5228–5235. https://doi.org/10.1073/pnas.0307752101

Hariri, N., Mobasher, B., & Burke, R. (2012). Context-aware music recommendation based on latent topic sequential patterns. *Proceedings of the Sixth ACM Conference on Recommender Systems*, 131–138. https://doi.org/10.1145/2365952.2365979

Hidayat, E. Y., Firdausillah, F., Hastuti, K., Dewi, I. N., & Azhari, A. (2015). Automatic Text Summarization Using Latent Drichlet Allocation (LDA) for Document Clustering. *International Journal of Advances in Intelligent Informatics*, *1*(3), 132–139. https://doi.org/10.26555/ijain.v1i3.43

Hong, L., & Davison, B. D. (2010). Empirical study of topic modeling in Twitter. *Proceedings of the First Workshop on Social Media Analytics*, 80–88. https://doi.org/10.1145/1964858.1964870

Greene, D., Cunningham, P., & Mayer, R. (2008). Unsupervised Learning and Clustering. In *Springer eBooks* (pp. 51–90). https://doi.org/10.1007/978-3-540-75171-7_3

Jiang, D., Leung, K. W.-T., Ng, W., & Li, H. (2013). Beyond Click Graph: Topic Modeling for Search Engine Query Log Analysis. *Database Systems for Advanced Applications*, 209–223. https://doi.org/10.1007/978-3-642-37487-6_18

Jiang, D., Vosecky, J., Leung, K. W.-T., & Ng, W. (2013). Panorama: a semantic-aware application search framework. *Proceedings of the 16th International Conference on Extending Database Technology*, 371–382.

Jiang, T., Yang, J., Yu, C., & Sang, Y. (2018). A Clickstream Data Analysis of the Differences between Visiting Behaviors of Desktop and Mobile Users. *Data and Information Management*, *2*(3), 130–140. https://doi.org/10.2478/dim-2018-0012

Jo, Y., & Oh, A. H. (2011). Aspect and sentiment unification model for online review analysis. *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, 815–824. https://doi.org/10.1145/1935826.1935932

Lafferty, J., & Blei, D. (2005). Correlated Topic Models. *Advances in Neural Information Processing Systems*. Advances in Neural Information Processing Systems.

Liu, L., Tang, L., Dong, W., Yao, S., & Zhou, W. (2016). An overview of topic modeling and its current applications in bioinformatics. *Springer Plus*, *5*(1), 1608. https://doi.org/10.1186/s40064-016-3252-8

Moe, W. W. (2003). Buying, Searching, or Browsing: Differentiating Between Online Shoppers Using In-Store Navigational Clickstream. *Journal of Consumer Psychology*, *13*(1–2), 29–39. https://doi.org/10.1207/S15327663JCP13-1&2_03

Niebles, J. C., Wang, H., & Fei-Fei, L. (2008). Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words. *International Journal of Computer Vision*, *79*(3), 299–318. https://doi.org/10.1007/s11263-007-0122-4

Porteous, I., Newman, D., Ihler, A., Asuncion, A., Smyth, P., & Welling, M. (2008). Fast collapsed gibbs sampling for latent dirichlet allocation. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 569–577. https://doi.org/10.1145/1401890.1401960

Schellong, D., Kemper, J., & Brettel, M. (2016). Clickstream Data as a Source to Uncover Con-Sumer Shopping Types in a Large-Scale Online Setting. In *European Conference on Information Systems*.

Topic Modeling with LSA, PSLA, LDA & lda2Vec | Nano Nets. (n.d.). Medium. https://medium.com/nanonets/topic-modeling-with-lsa-psla-lda-and-lda2vec-555ff65b0b05

Wang, X., & McCallum, A. (2006). Topics over time: a non-Markov continuous-time model of topical trends. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 424–433. https://doi.org/10.1145/1150402.1150450

You, T., Sun, Y., Zhang, Y., Chen, J., Zhang, P., & Yang, M. (2022). Accelerated Frequent Closed Sequential Pattern Mining for uncertain data. *Expert Systems with Applications*, *204*, 117254. https://doi.org/10.1016/j.eswa.2022.117254

Younus, A., O'Riordan, C., & Pasi, G. (2014). A Language Modeling Approach to Personalized Search Based on Users' Microblog Behavior. *Advances in Information Retrieval*, 727–732. https://doi.org/10.1007/978-3-319-06028-6_83

Yu, J., & Egger, R. (2021). Color and engagement in touristic Instagram pictures: A machine learning approach. *Annals of Tourism Research*, *89*, 103204. https://doi.org/10.1016/j.annals.2021.103204

Zhao, W., Chen, J. J., Perkins, R., Wang, Y., Liu, Z., Hong, H., Tong, W., & Zou, W. (2016). A novel procedure on next generation sequencing data analysis using text mining algorithm. *BMC Bioinformatics*, *17*(1), 213. https://doi.org/10.1186/s12859-016-1075-9