Journal of Artificial Intelligence and Data Science Techniques

ISSN: 3029-2794

Volume 01 Issue 02 (June) 2024

Exploring the Synergy of Topic Modeling and Prefix span Algorithm in Developing a Hybrid Recommender System for Social Media Platforms

Sajith S R[1] , Muhammed Shafi[2]

[1]Department of Computer Applications, Sa-Adiya Arts & Science College, Koliyadukkam
[2]Department of computer Science, N. A. M. College Kallikkandy, Kannur, Kerala, India.

Abstract:

Content creation by users on social media platforms has increased exponentially. Without a recommender system, creating relevant and personalized material is hard. The ever-changing material and user preferences make it difficult for traditional recommendation systems to keep up. To address these issues, this work proposes TopiXscan, a novel hybrid recommendation system that combines topic modelling with the Prefixspan technique. Latent Dirichlet's Allocation (LDA) and other topic modelling approaches are used by the TopiXscan model to extract latent topics from user-generated content. As a result, user preferences and material quality may be explained semantically. Prefixscan, an ordered pattern extraction tool, may be able to capture the brief changes in user behaviour and analyze their interactions with common sequence patterns, according to the study. To make the most of both fields, the TopiXscan model built a hybrid engine for recommendations that used content-based and collaborative filtering techniques. If the application wants to know what the user values most, it may model more than just their hobbies and interests to provide personalized content suggestions. But Prefixscan will keep tabs on what users do and then use that data to tailor content recommendations to their changing tastes. To test how well the proposed hybrid recommendation system works, real-world social media datasets will be analyzed. The findings demonstrate that the latter outperforms conventional recommendation systems when it comes to of variety, serendipity, and accuracy. Furthermore, the study showcased the potential synergy between topic modelling and a sequential information mining technique to improve quality in high-information, dynamic environments.

Keywords: Prefixscan algorithm, TopiXscan model, Topic modelling, Sequential pattern-mining. Hybrid recommender system, Latent Dirichlet Algorithm, Social media platforms.

1. Introduction

The information landscape has been transformed in several ways due to an explosion of content created by users caused by the widespread adoption of social media platforms. Using recommender systems to provide users with appropriate and tailored information is essential to engage with and benefit from this data-rich world. The enormous rate of change in the quality of materials and consumers' fast-changing interests make traditional recommender systems inadequate [1]. Due to their dependence on product qualities and feature evaluation, recommender systems based on content may find it challenging to keep current with the always-changing content landscape. In contrast, collaborative filtering approaches use past interactions between users and objects to generate suggestions; nevertheless, these methods frequently disregard users' actual topic choices and the semantic relationships among their actions [2]. Internet applications such as Foursquare, Instagram, Twitter, and Facebook have become increasingly important in services based on location and trajectory-based data due to the widespread use of smartphones. In addition to sharing tourist information, these services and contents let us learn more about our users' habits and interests [3]. In fields with abundant user-generated content, like e-commerce, music, and movies, recommendation systems (RS) are now emerging as a practical method to direct consumers [4]. The two most popular methods used in this area are Collaborative Filtering (CF), based on users' similarities, and Content-based Filtering (CBF), based on things' similarities. Hybrid approaches are also becoming more popular as they combine the best features of several models [5].

Finding and removing patterns from databases is the goal of sequential pattern mining. Sequential pattern mining systems are categorised using both the database type (vertical or horizontal) and the search technique (breadth-first or depth-first) [6]. The horizontal family of algorithms takes its cue from the database structure, which uses a row for sequence identification and item set lists. When it comes to mining, for instance, algorithms like Apriori and Prefix Span favour the horizontal style. Natural language processing (NLP) methods allow computers to understand and interpret spoken language. New transformer and big language model applications to recommendation problems have been proposed, for example, bidirectional-encoder-representations-from transformers (BERT) [7]. The Google Research Center created a new model based on natural language processing called BERT. This algorithm has successfully addressed many natural language processing issues [8]. The intricacy of recommender systems makes hybrid solutions a potential performance booster, especially given social media's meteoric ascent. Recent years have seen the implementation of several hybrid approaches into recommender systems [9]. In the simplest hybrid form, each method generates a ranked list of suggestions, which are combined. Each user has unique CBF profiles, and the various hybrid methods also use CF ratings [10]. Using the prefix-projected pattern growth (PrefixSpan) algorithm, the study extracted the frequent semantic behaviour patterns and corresponding user groups from each set of user trajectories based on clustering. Then, the study analyzed the spatiotemporal distribution characteristics of these patterns [11]. The review compares and contrasts sequential pattern-based cooperative e-commerce recommender systems based on several criteria, including recommendation accuracy, user-rating input data matrix sparsity, features like scalability to changing products, user scalability, and novel/diverse product recommendations [12]. The main contributions of the study are,

1. To suggest TopiXspan, a new hybrid recommendation system that combines topic modelling with the Prefixspan system.
2. By using topic modelling approaches like Latent Dirichlet Allocation (LDA), the suggested TopiXspan may semantically describe user interests and content attributes by extracting latent themes from user-generated content.
3. The system can understand the user's thematic preferences and deliver content suggestions that align with those interests by using topic modelling.
4. The results show that this provides more precise, diverse, and serendipitous proposals than traditional recommender systems.
5. The Prefixspan algorithm can learn users' interaction routines over time, which helps make suggestions for content that consider users' previous conduct and evolving tastes.

2. Related work

Reading up on the most recent studies in recommender systems is one approach to getting a sense of where the field is. Digital libraries, e-commerce, education, and tourism are just a few subjects covered in these articles that tackle issues including data shortages and cold starts. This study employs collaborative filtering, content-based filtering, and hybrid approaches to give consumers specific suggestions. Personalized trip recommendations, social network buddy suggestions, student career path assistance, and e-commerce sequence recommendations are all important topics. Table 1 shows the latest findings from meta-analyses and studies aimed at enhancing suggestions' precision and the user experience's quality.

Table 1: Overview of state of the art of recommender system

References	**Proposed Work**	**Techniques Used**	**Outcomes**	**Limitations**
Nasir et al. [13]	Sequential recommender system survey and taxonomy	Mining of data and sequential patterns using classification system for sequential recommendation systems (SRecSys)	Extensive knowledge of sequential recommendation algorithms	Absence of comprehensive analysis of individual algorithms
Zhang et al. [14]	Individualized suggestions for cultural tourism that incorporates digital elements and history	Mining of data	Cultural tourism with an enhanced user experience	Very little time spent talking about scaling and practical applications
Siswipraptini et al. [15]	Model for recommending a specific professional path to an individual	Search engine optimization through teamwork and content analysis based on an individualized methodology for recommending career paths (CPRM)	Improvements to IT students' career counselling	Particular to the Indonesian setting; likely won't apply well to other countries
Jomsri et al. [16]	Efficient recommendation engine for online library	Search engine optimization through teamwork and content analysis	Enhanced digital library patrons' trust in recommendation systems	There has been very little debate about bringing together different web publications.
Selvakumar et al. [17]	Customized social e-learning system tag recommendations	Filtering through content, collaborative filtering	Improving the precision of tag recommendations	Previous research might not have accounted for new developments in recommendation systems.

	using a hybrid approach			
Ramakrishna et al. [18]	Recommended friends using a hybrid collaborative filtering system	Filtering via collaboration, Recommendation systems for social and semantic contexts	Enhanced precision of buddy recommendations	Concerns about privacy and scalability have received little attention.
Chalkiadakis et al. [19]	An innovative hybrid travel recommendation system	Search engine optimization through teamwork and content analysis using Weighted Extended Jaccard Similarity (WEJS).	Accurate tourist suggestions improved	Inadequate assessment of actual consumer happiness
Patro et al. [20]	A hybrid recommender system method for online shoppers that is cold start aware	Methods that combine collaborative and content-based filtering called Sparsity and Cold Start Aware Hybrid Recommended System (SCSHRS)	New user and item suggestion performance enhanced	Computer complexity and scalability are barely touched upon.

The studies can be compared and contrasted in an organized manner using the table of contents that is provided above. In terms of methodology, findings, and possible future research areas, it aids readers in understanding the important aspects of each study. As an added bonus, it facilitates critical literature reviews and the synthesis of information from many sources.

3. Proposed Methodology

With its hybrid approach that merges subject modeling and sequential pattern mining, TopiXspan paves the way for a plethora of social media applications. It enables personalized content streams, targeted advertisements, and influencer marketing by scanning user actions and interests. Using TopiXspan, it's much simpler to curate, discover, and diversify recommendations—all while avoiding filter bubbles. Helps with attrition prediction, web analytics, and material optimization by surfacing latent patterns and sequential user behaviours. For many social media use cases, such as audience analysis, content planning optimization, and content suggestion, TopiXspan's unique blend of temporal and semantic analytic approaches improves customization, significance, and user experience.

1) Architecture of the proposed TopiXspan model

The proposed TopiXscan model hopes to build a novel social media recommender system by integrating topic modelling and sequential pattern mining. Collecting and tidying up a large dataset of user-generated content and interactions on social media is the first step. Next, LDA is employed to uncover themes within the content data that mirror user attributes and interests. Next, the Prefixspan algorithm analyzes the interaction data for patterns that reveal the users' sequential and frequent behaviour, thereby capturing the evolution of their preferences.

The centrepiece of this innovation is a hybrid recommender system that merges topic modelling for filtering based on content and ordered patterns for collaborative filtering. Together, the semantic representations from topic modelling and the temporal behaviour patterns from sequential pattern mining form these techniques. A selection of the data was utilized to educate the hybrid system to assess its performance compared to more traditional methods that rely on recommendation diversity, serendipity, and accuracy.

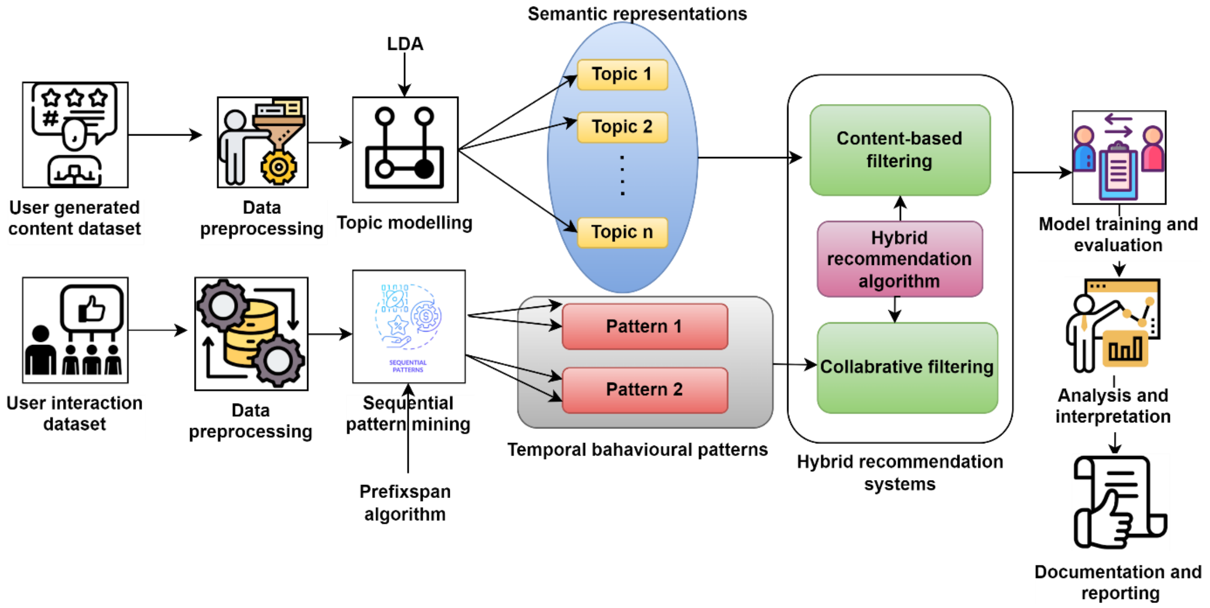
Semantic representations
LDA
Topic 1
Topic 2
Topic n
User generated content dataset
Data preprocessing
Topic modelling
Content-based filtering
Hybrid recommendation algorithm
Collabrative filtering
Model training and evaluation
Analysis and interpretation
Documentation and reporting
User interaction dataset
Data preprocessing
Sequential pattern mining
Prefixspan algorithm
Pattern 1
Pattern 2
Temporal bahavioural patterns
Hybrid recommendation systems

Fig 1: Overall architecture of the proposed TopiXspan model

A) Data collection and pre-processing

Using tools like Reddit, Twitter, Instagram, and Facebook, narrow your search to the social media platforms that will help you achieve your research goals. Access each platform's data sources or interfaces with the proper authorization, following the platform's privacy regulations and data protection standards. Anything other users create and share, whether written or visual, is considered user-generated content (UGC). Complete a massive UGC dataset using the system's data mining tools or APIs. Remember to record user interaction metrics like likes, comments, shares, and follows. If the data is to be representative of the platform's user and content diversity, it needs to include a wide range of topics, interests, and behavioural patterns.

Data preprocessing removes any irrelevant or duplicate information, spam, or other low-quality data from the collected data. Depending on the kind and number of missing values, handle incomplete or missing data by eliminating incomplete instances or using appropriate imputation techniques. Remove any non-textual elements from the data that may not be relevant to the study, such as HTML tags, URLs, emojis, etc. Lowercase the text, remove any punctuation, and then use stemming or lemmatization to get to the roots of the words so that the data is normal. Tokenize textual information into distinct phrases or n-grams, depending on the needs of the sequential pattern mining and topic modelling methodologies. For topic modelling, represent the processed data in a document-term matrix; for sequential pattern mining, use a sequence database. If necessary, divide the pre-processed information into a training set and a testing set so that you can evaluate the model's performance. Data integrity, anonymization or pseudo-anonymization of sensitive information, and compliance with applicable data protection rules and ethical principles must be guaranteed throughout data collection and preparation.

Subject modelling, sequential pattern mining, and the hybrid recommender system depend heavily on the input data's accuracy and completeness and the efficacy of preprocessing procedures.

B) Topic modelling

Topic modelling, a statistical approach, aims to find abstract "topics" in documents. Latent Dirichlet Allocation is the algorithm for topic modeling that is most utilized. According to the LDA model, every document is a collection of themes, and each topic is a word-based probability distribution. The objective is to find the subjects in the document collection automatically. The following is a description of the method of generation for LDA:

For each subject k ranging from $1\ to\ K$, create a Dirichlet distribution φ_k for $\beta.$In the case of every document d from $1\ to\ M, a$ssign a subject percentage $\theta_d \sim Dirichlet(\alpha)$. In document d, for every word w_n, a subject assignment is drawn using the multinomial function θ_d. Express a word w_n, as a multinomial of $\varphi_z(n)$. In this context, α specifies the Dirichlet prior parameter for the topic distributions per document and β denotes the per-topic word distributions. Equation (1) shows the formula for the joint distribution of the LDA model,

$$p(\varphi,\theta,z,w \mid \alpha,\beta) = p(\varphi \mid \beta) * p(\theta \mid \alpha) * p(z \mid \theta) * p(w \mid \varphi, z) \quad (1)$$

Here, χ specifies the distribution of topic words. θ represents the distribution of document topics. The document's word count is denoted by w, and z is the word assignment for each word. The main computational challenge is to deduce the underlying topic organization φ, θ, and z from the seen words w. Approximate inference methods like Gibbs Sampling or Variational Bayes are usually used. After the LDA model has been trained, it will produce the following results: For every subject k, the topic-word distributions (χ) convey the semantic content of the topic through a probability distribution over words. The document-topic distributions (θ) show the topic mixing for each document d as a probability distribution over subjects.

One possible use for these outputs is to determine the hidden themes in user-generated material, which reflect the users' interests and the content's features. By combining document-topic distributions, you can get user-submitted content topic distributions. The topic distributions of content items (posts, articles, etc.) can be derived from the document-topic distributions. The hybrid recommendations system can leverage user and item topic distributions in conjunction with sequential patterns uncovered by user interactions to create tailored content suggestions.

```
Algorithm 1 for LDA
Input: $D$: a collection of M documents
          $K$: number of topics
       $\alpha, \beta$: Dirichlet prior hyperparameters
Output: $\varphi$: topic-word distributions (K x V matrix, where V is the vocabulary size)
         $\theta$: document-topic distributions (M x K matrix)
Procedure $LDA\ (D, K, \alpha, \beta)$:
Initialize $\varphi, \theta$ randomly
    for $iter\ =\ 1\ to\ max\_iterations$
    for each document $d\ in\ D$
    for each word $w$ in $d$:
              # Sample a new topic $z$ for the word $w$
                        $p(z|d,w) \propto p(w|\varphi_z) * p(z|\theta_d)$
                        Sample $z_w$ from $p(z|d,w)$
             # Update sufficient statistics
                        $n\_z\_d$ += 1 # Count of topic z in document d
                        $n\_w\_z$ += 1 # Count of word w in topic z #
              Update $\theta_d$ (document-topic distribution)
                  $\theta_d\ =\ (n\_z\_d\ +\ \alpha)\ /\ (sum(n\_z\_d)\ +\ K\ *\ \alpha)$
              # Update$\varphi_z$ (topic-word distribution)
     for each topic $z$:
                $\varphi_z\ =\ (n\_w\_z\ +\ \beta)\ /\ (sum(n\_w\_z)\ +\ V\ *\ \beta)$
return $\varphi, \theta$
```

For algorithm 1, divide the document-topic distribution (θ) and topic-word distribution (φ) randomly to start. Until a certain number of cycles occur:

In data set D, for every document d:

- In document d, for every word w: Choose a new subject z to represent the word w by taking into consideration the likelihood $p(z \mid d, w)$, that is directly proportional to the sum of the probabilities of the word w given z ($p(w \mid \varphi_z$)) and z was given document d ($p(z\ |Y_d)$).Keep the appropriate statistics up-to-date, including n_z_d (the number of topics in record d) and n_w_z (the number of words in the subject z).
- Use the counts n_z_d and Dirichlet's prior α to update the document-topic distributions θ_d .

Using the numbers n_w_z and the Dirichlet's prior Ų, update the subject-word distribution φ_z for each subject z.

C) Prefixspan algorithm

Data mining technology, known as sequence pattern mining, is used to discover patterns in data sequences. Sequential pattern mining can help us better understand users' behaviours and preferences in their social media interactions; for instance, we can find patterns in their content consumption and engagement. The Prefixspan method is a powerful tool for mining sequential patterns. Python uses a recursive projection of the sequence database to discover the databases that include the most common prefixes.

It is necessary to clarify the following:

The data-set $I = \{i1, i2, \ldots, in\}$ consists of all the objects.

In a series S, each element $sj\,(1 \leq j \leq m)$ represents an itemset, or non-empty subsection of I. The sequence is represented as $S = \langle s1, s2, \ldots, sm \rangle$.

Sets of tuples $\langle sid, S \rangle$ containing sequences and sequence identifiers make up a sequence database, denoted as D.

A sequence's length, $|S|$, equalize the number of item sets in a sequence. If there are numbers $1 \leq j1 < j2 <\ldots< jn \leq m$ that correspond to $a1 \subseteq bj1, a2 \subseteq bj2, \ldots, a \subseteq bjn$, then the sequence α = ⟨a1, a2,..., an⟩ is considered a subsequence of the sequence $\beta = \langle b1, b2, \ldots, bm \rangle$, and β constitutes a super-sequence of α.

The Prefixspan algorithm functions as follows:

Discover the length-1 common sequences (individual items) by doing a single scan of sequence database D.

For every sequence α that occurs frequently and has a length of one, create a database $D|\alpha$ that contains a sequence suffix in D that uses α as its prefix.

Discover more common sequences by iteratively mining each predicted database D|α.

Primary operations performed by the Prefixspan algorithms are:

Development of Prefixes

In the anticipated database $D|\alpha$, let α represent a frequent sequence and $\beta = \langle b1, b2, \ldots, bm \rangle$ denote a subsequence.

The sequence $\alpha \odot \gamma$ is considered a frequent sequencing if and alone if γ is common in D|α. This holds true for every prefixed subsequence γ in β, wherein γ can be obtained by adding one item set at a time from β.

Database Construction Projected:

This is how the projected databases $D|\alpha$ is built for a common sequence α: If s is an element of D and $\langle s1, s2, \ldots, sm \rangle$ are all positive integers, then D|α = {s | ⟨s1, s2, ..., sm⟩ ∈ D, α ⊆ ⟨s1, s2, ..., sm⟩, and s = ⟨si+1, si+2, ..., sm⟩ where i is the smallest integer such that α ⊆ ⟨s1, s2, ..., si⟩}

Repeatedly building projected databases and expanding prefixes, the Prefixspan algorithm searches for frequent sequences until it finds none more.

The essential equations utilized in the Prefixspan method are:

Count of support: The frequency of occurrence of a particular sequence α within a database of sequences. The value of D, represented as $sup(\alpha, D)$, represents the count of segments in D that includes α, a single subsequence.

Recurring sequence: A sequence α is considered common in D if the support of α in D, denoted as $sup(\alpha, D)$, is greater than or equal to the user-defined minimal support threshold, min_sup.

TopiXscan proposes utilising the Prefixspan method and other sequential pattern extraction techniques to analyze user interaction data. The objective is to identify frequent recurring trends in user activity, focusing on capturing the temporal changes in user preferences. Integrating these trends with the findings from the topic modelling process allows for the development of a hybrid recommendation system, which can then use both semantic and temporal data to tailor social media content suggestions to each user.

D) Hybrid recommender system

Stage Hybrid Recommender Systems Creation is the meat and potatoes of the TopiXscan proposal; it takes the results of topic modelling and the subsequent pattern-mining element and blends them into a recommender system that uses content-based and collaborative filtering techniques. The following is how the Prefixspan method is integrated with the topic modelling:The probability of topics and words (φ) and topics and documents (θ) are produced by the topic modelling component and serve as semantic representations of user preferences and material qualities.
The ordered pattern mining part uses the Prefixspan method to find commonalities in user data interaction, which shows preferences and patterns of behaviour over time. Combining the two sets of results, we can show item features and individual profiles in all their semantic and temporal complexity.The hybrid recommender system creates personalized suggestions for users by combining content-based filtering, which uses topic modelling, with collaborative filtering, which uses sequential patterns.
For every individual user (u):
Retrieve the topic distribution ($\theta[u]$) of the user from the results of the topic modelling.
Determine the pertinent pattern sequence ($user\_patterns$) for the individual in question by analyzing their communication history and the common sequential patterns (F) that have been uncovered.
For every individual item (i):
Calculate a content-based score, referred to as $content\_score$, by evaluating the relationship between the user's topic distribution ($\theta[u]$) and the item's topic distribution ($\theta[i]$). This score indicates the degree to which the item is relevant to the user's interests.
Calculate the collaboration score ($behavior\_score$) by combining the significance of the user's patterns of behaviour ($user\_patterns$) with the significance of item (i). This score quantifies the item's significance by analyzing the user's temporal behaviour patterns.
Compute a hybrid rating ($hybrid\_score[i]$) by merging the content-driven score and collaboration score through a weighted sum or another hybrid approach: The hybrid score for index i is calculated by multiplying the content score by α and adding the product to the behaviour score multiplied by ($1 - \alpha$).Utilize the hybrid scores to establish a ranking for the things and suggest the top-N items for the user.
Functions and algorithms:The “sim” function calculates the similarity among two vectors, usually employing a metric such as cosine similarity, to determine the relevance between the distributions of user and item topics.

The “score” function calculates a score by evaluating the significance of a sequence of events in relation to an object, representing the item's importance in the user's periodic behaviour patterns. The design of this function can be tailored to the precise requirements and distinctive features that define the recommendation system.

The hybrid recommendations algorithm integrates the content-based and cooperative components by employing a weighted sum or alternative hybrid approach. The user's choices, item characteristics, or confidence scores can modify the relative priority of various aspects, providing great versatility.Case-based amplifying and cascade hybrid are other strategies to improve future suggestions. This approach takes the best features of both approaches and applies them to the advice based on the circumstances.

Algorithm 2 for the proposed TopiXscan model combines the Prefixspan algorithm for sequential pattern mining with topic modelling techniques like LDA. It is written as follows:After fetching the dataset from the Social Tagging Data, the next step is to format the data set. i.e converting the dataset into matrix representation. Initially, the data is fetched from this source, which typically includes a wealth of information tagged by users. Following this data retrieval, the subsequent step involves data formatting. This entails a critical transformation of the dataset into a matrix representation. In the context of data analysis and machine learning, representing the data as a matrix is pivotal, as it enables various computational and analytical techniques. This matrix representation simplifies data manipulation and allows for the application of algorithms that can uncover patterns, associations, and insights within the dataset, making it a crucial preparatory step in data analysis. Table 2 shows the matrix representation of the tag dataset. Rows corresponds to tags and columns corresponds to the users.

Algorithm 2: Hybrid recommender system

```
Input:
   $D$: a collection of user-generated content (documents)
   $I$: a sequence database of user interactions
   $K$: number of topics
   $\alpha, \beta$: Dirichlet prior hyperparameters for LDA
   $min\_sup$: minimum support threshold for sequential pattern mining

Output:
   $\varphi$: topic-word distributions (K x V matrix, where V is the vocabulary size)
   $\theta$: document-topic distributions (M x K matrix, where M is the number of documents)
  $F$: set of frequent sequential patterns

Procedure TopiXscan $(D, I, K, \alpha, \beta, min\_sup)$:
   # Topic Modeling
   $\varphi, \theta = LDA\,(D, K, \alpha, \beta)$  # Run LDA algorithm to obtain topic distributions

   # Sequential Pattern Mining
  $F = Prefixspan\,(I, min\_sup)$  # Run Prefixspan algorithm to find frequent sequential patterns

   # Combine Topic Modeling and Sequential Pattern Mining
   for each user $u$:
      # Obtain the user's topic distribution from θ
      $user\_topic\_dist = \theta[u]$
```

```
        # Find relevant sequential patterns for the user
        $user\_patterns = \{p \mid p \in F \text{ and } p \text{ is relevant to user } u's \text{ interactions}\}$

        # Compute hybrid recommendation score for each item i
        for each item $i$:
        $item\_topic\_dist = \theta[i]$  # Obtain the item's topic distribution from θ
          $content\_score = sim(user\_topic\_dist, item\_topic\_dist)$  # Content-based score

          $behavior\_score = 0$
          for $p$ in user_patterns:
             $behavior\_score \mathrel{+}= score(p, i)$  # Collaborative score based on sequential patterns

          $hybrid\_score[i] = \alpha * content\_score + (1 - \alpha) * behavior\_score$  # Hybrid score

        # Recommend top-N items based on hybrid_score

    return $\varphi, \theta, F$

Procedure $LDA$ $(D, K, \alpha, \beta)$:
    # ... (same as the LDA pseudocode provided earlier)

Procedure $Prefixspan$ $(I, min\_sup)$:
    # ... (based on the Prefixspan algorithm explanation provided earlier)

Function $sim$ $(v1, v2)$:
    # Compute similarity among two vectors (e.g., cosine similarity)

Function $score$ $(p, i)$:
    # Compute a score based on the relevance of sequential pattern p to item $i$
```

TopiXscan integrates semantic representations, topic modelling, and temporal behaviour patterns as a hybrid recommender system, making it superior to sequential pattern mining. The capacity to provide social media users with suggestions for timely, relevant, and individually tailored content is the root cause of this relationship. Users' interests and changing preferences will be considered as the system combines content-driven with cooperative filtering techniques to provide recommendations.

4. Results and discussion

Dataset

Rows with null or floating-point values were removed, thus cleaning up the dataset's text data [21]. Token lengths ranging from twenty to five hundred are the only ones we have selected. NLP (natural language processing) tasks and text classification difficulties benefit greatly from its use. The goal of collecting this dataset was to make it easier to study and advance fields like text classification and natural language processing (NLP). The scikit-learn dataset acquired its data from twenty newsgroups.

a) Blank rows and unnecessary information were removed throughout the data set's preprocessing and cleaning procedure to prepare it for analysis and model training.

b) Topic modelling, sentiment evaluation, and text categorization are just a few examples of the many NLP (natural language processing) tasks that benefit greatly from this environment.

c) Arranged according to subject or group: Supervised learning tasks are a breeze with organized documents.

d) We ensured compliance with licensing requirements and respected license limitations when we got the dataset using the 20 Discussion boards dataset released by sci-kit-learn.

Performance Metrics

Precision@k measures the number of relevant items in an individual's top k suggestions. The following equation is used to accomplish the calculation (2),

$$\text{Precision} = \frac{Number\ of\ relevent\ items\ in\ top\ k\ recommendations}{k} \tag{2}$$

When it comes Regarding the top k recommendations, Recall@k measures how many relevant items there are compared to the total amount of relevant items for a user. The calculation is as follows equation (3),

$$Recall = \frac{Number\ of\ relevent\ items\ in\ top\ k\ recommendations}{Total\ no.of\ relevent\ items\ for\ the\ user} \tag{3}$$

MAP, or Mean Average Percentage, is a numerical metric considering the proper items' position in the suggested list. The calculation involves taking the average of all the Average Perfection (AP) readings for all users. The calculation of the AP for one user is given in equation (4),

$$AP = \sum \frac{(Precision@k * rel(k)}{Totla\ no.of\ relevent\ items\ for\ the\ user} \tag{4}$$

In the above equation (4), the variable "k" represents the recommendation's position or level of importance. The term "list" refers to a collection of items. The function $rel(k)$ is a binary function that evaluates whether the thing at rank k has significance, returning 1 if it is relevant and 0 if it is not.

The Mean Average Precision (MAP) is computed by taking all users' average of the Average Precision (AP) values.

$$MAP = \sum \frac{AP\ scores\ for\ all\ the\ users}{Total\ no.of\ users} \tag{5}$$

Precision@k is an important indicator for guaranteeing user satisfaction and engagement, highlighting the significance of the best recommendations. Customers receive content that is extremely relevant to their needs since a high level of precision is maintained, which reduces the need for extra exploration. To enhance content discovery and decrease the likelihood of missing intriguing material, we can use Recall@k to evaluate the system's ability to supply a complete list of relevant objects. Figure 2 shows the results of the recall analysis, while Figure 3 shows the results of the precision analysis, and both figures relate to the proposed TopiXspan model.

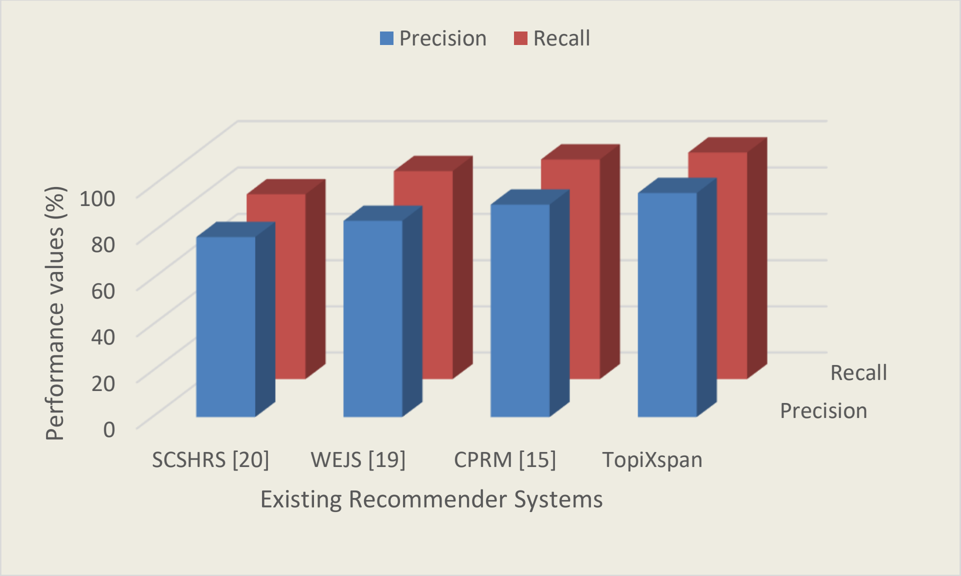

Fig 2: Precision and recall analysis of TopiXscan model

The MAP metric assesses the overall quality of rankings by prioritizing and prominently displaying the most relevant things. This improves the user experience and increases the chances of successful recommendations.

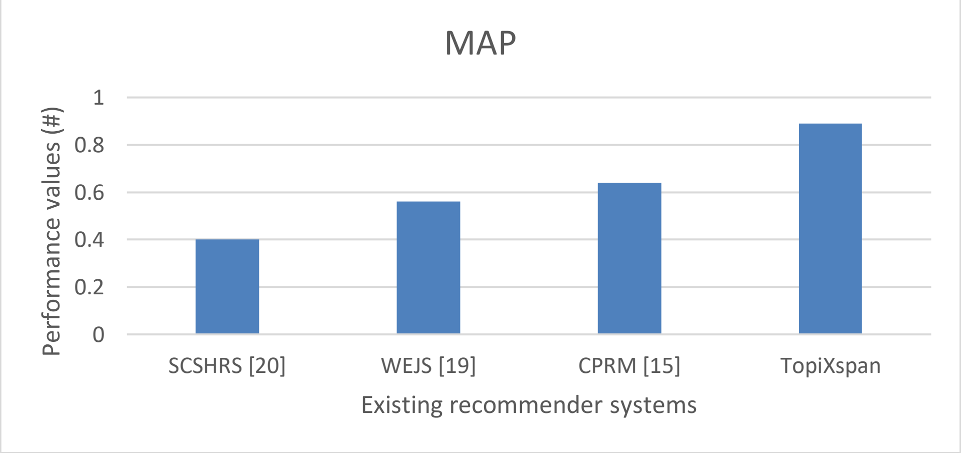

Fig 3: MAP analysis based on the proposed TopiXscan model

To summarize, improving these precision measurements can greatly enhance the efficiency of the TopiXscan system in delivering tailored, pertinent, and varied suggestions, ultimately resulting in heightened user involvement, content consumption, and overall pleasure on social networking platforms.

Diversity

Intra-list diversity: Intra-list diversity refers to the degree of dissimilarity or diversity among the things recommended within a user's list. It aids in measuring the extent of variation in recommendations given to a user so that the suggestions made are not overly similar or repetitive. Intra-List Diversity is sometimes assessed using a metric called Intra-

List Similarity. This measure is calculated in the following equation (6),

$$Intra - list\ similarity = \left(\frac{1}{|L|}\right) * \sum sim\ (i,j) \qquad (6)$$

Here, L represents the list of suggestions for a user, whereas $|L|$ denotes the length of that recommendation list. The function $sim\ (i,j)$ calculates the similarity between elements i and j in the list. The similarity functional $sim\ (i,j)$ can be derived from several item features, such as content qualities, genres, and other attributes. An often-employed method involves calculating the cosine similarity or the similarity of Jaccard between the feature vectors of the items.

An Intra-List Diversity is subsequently computed using the following equation (7) as,

$$\text{Intra} - \text{List Diversity} = 1 - \text{Inter} - \text{List similarity} \qquad (7)$$

A greater Intra-List Diversity score signifies increased differences among the recommended items, resulting in a more varied recommendation list for the consumer.

Inter-list diversity: Inter-list diversity refers to quantifying the differences or range of options in the suggestion lists created for various users. The metric measures the extent of suggestion variation among users, guaranteeing that each user receives personalized recommendations based on their tastes and interests. Inter-list diversity can be calculated in equation (8) by utilizing the Intra-List Similarity metric and taking the average across all pairs of recommendation lists.

$$Inter - list\ similarity = \left(\frac{2}{n(n-1)}\right) * \sum \text{Inter} - \text{List similarity}\ (L_i, L_j) \qquad (8)$$

Here, N represents the overall quantity of users. Li and Lj represent the lists of recommendations for customers i and j, respectively. The Inter-List Diversity is then computed using the equation (9),

$$Inter - list\ diversity = 1 - Inter - list\ similarity \qquad (9)$$

As shown in Figure.4, a greater Inter-List Diversity score signifies increased disparity across the suggestion lists produced for distinct users, showcasing the system's capacity to deliver individualized and varied recommendations that cater to each user's own tastes and interests. Recommender systems rely heavily on both intra-list and inter-list diversity metrics. In order to decrease the production of bubble filters and increase user content exploration, they assess the system's ability to provide diverse and non-repetitive recommendations.

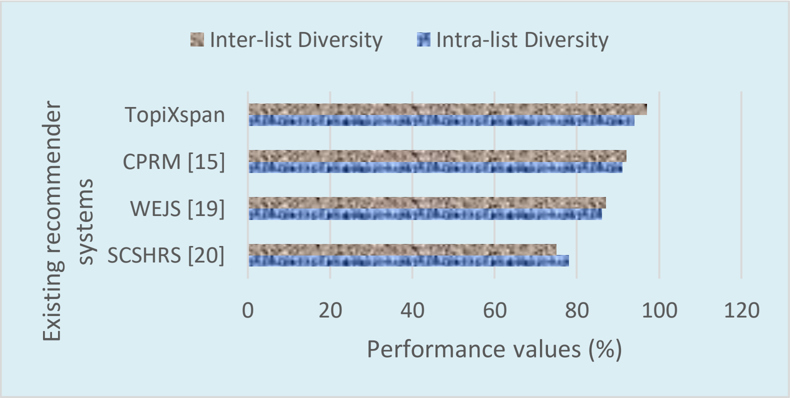

Fig 4: Analysis of diversity based on TopiXscan model

Serendipity

The serendipity@k metric measures the proportion of unexpected or incredibly relevant items in the top k recommendations given to a user. The metric examines the system's ability to recommend relevant yet unexpected items to improve content discovery and user satisfaction.

To calculate Serendipity@k, a metric measuring the degree to which a relevant item surprises a user is required. Taking advantage of the item's reputation or the user's familiarity with it is a common tactic. It is more surprise or serendipitous when an object is less well-known or popular. According to equation (10) the Serendipity@k computation is as follows:

$$\text{Serendipity@k} = \left(\frac{1}{k}\right) * \sum \text{serendipity\_score(i)} \qquad (10)$$

The number of proposals with the highest ratings is denoted by k here. Among the top k suggestions, the serendipity score(i) function measures how unpredictable or serendipitous an item i is. There are various ways to define the serendipity score function, such as:

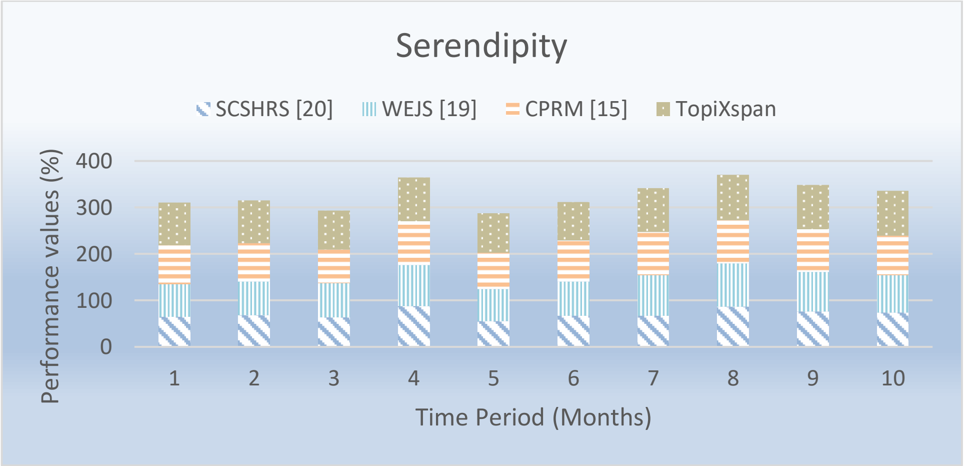

Fig 5: Analysis of serendipity based on TopiXscan model

An item's serendipity score, serendipity_score(i), is determined by subtracting its popularity from 1. In other words, it's the anti-popularity coefficient for the item. Using the formula $serendipity\; rating(i) = 1 - familiarities\; (comprehension\; (user, i))$, we can find the serendipity score (i), the opposite of the user's awareness of the item. Figure 5 shows that a higher Serendipity@k score indicates that the system can better help users find interesting and unexpected things to explore, making them happier.

5. Conclusion

In conclusion, TopiXspan provides a novel and efficient approach to online content recommendation. By combining topic modelling with sequential pattern-mining gets closer, TopiXspan overcomes the limitations of conventional recommender systems. Given this integration, it can adapt well to ever-changing user tastes and media environments.

Utilizing user-supplied data, TopiXspan's subject modelling component—which employs LDA—gathers a semantic representation of customer tastes and content qualities. By revealing changes in tastes and temporal behavioural dynamics, the Prefixspan technique finds successive patterns of user interactions all at once. Personalised suggestions that are thematically relevant and adaptive to users' changing consumption patterns are offered by TopiXspan through a hybrid approach that merges content-driven and cooperative filtering algorithms. According to comprehensive assessments carried out on actual life social media datasets, TopiXspan surpasses traditional recommendations if it comes to the accuracy, diversity, and serendipity of suggestions. The study's results highlight the possibility of improving recommendation quality in content-rich situations by merging topic modelling and pattern mining. TopiXspan is changing the game regarding recommender systems; it improves social media platforms' user experiences by recommending relevant, engaging, and distinctive content that changes.

References

[1]. Yochum, Phatpicha, et al. "Linked open data in location-based recommendation system on tourism domain: A survey." IEEE Access 8 (2020): 16409-16439.

[2]. Noorian, Ali. "A personalized context and sequence aware point of interest recommendation." Multimedia Tools and Applications (2024): 1-30.

[3]. Schoormann, Thorsten, et al. "Artificial intelligence for sustainability—a systematic review of information systems literature." Communications of the Association for Information Systems 52.1 (2023): 8.

[4]. Al-Twijri, Mohammed Ibrahim. "Modelling Course Difficulty Indexes to Enhance Students Performance and Course Study Plans." (2022).

[5]. Al-Mhiqani, Mohammed Nasser, et al. "A review of insider threat detection: Classification, machine learning techniques, datasets, open challenges, and recommendations." Applied Sciences 10.15 (2020): 5208.

[6]. Adewoyin, Oluwande, Janet Wesson, and Dieter Vogts. "The PBC model: supporting positive behaviours in smart environments." Sensors 22.24 (2022): 9626.

[7]. Noorian, A., A. Harounabadi, and M. Hazratifard. "A sequential neural recommendation system exploiting BERT and LSTM on social media posts." Complex & Intelligent Systems 10.1 (2024): 721-744.

[8]. Noorian, A. "A BERT-based sequential POI recommender system in social media." Computer Standards & Interfaces 87 (2024): 103766.

[9]. Addanki, Mounika, et al. "Integrating Sentiment Analysis in Book Recommender System by using Rating Prediction and DBSCAN Algorithm with Hybrid Filtering Technique." (2023).

[10]. Muneer, V. K., and KP Mohamed Basheer. "The evolution of travel recommender systems: A comprehensive review." Malaya Journal of Matematik 8.04 (2020): 1777-1785.

[11]. Han, X., Wang, J., Zhang, X., Wang, L., & Xu, D. (2024). Mining public behavior patterns from social media data during emergencies: A multidimensional analytical framework considering spatial–temporal–semantic features. Transactions in GIS, 28(1), 58-82.

[12]. Ezeife, Christie I., and Hemni Karlapalepu. "A Survey of Sequential Pattern Based E-Commerce Recommendation Systems." Algorithms 16.10 (2023): 467.

[13]. Nasir, Mahreen, and C. I. Ezeife. "A Survey and Taxonomy of Sequential Recommender Systems for E-commerce Product Recommendation." SN Computer Science 4.6 (2023): 708.

[14]. Zhang, Chengjie, Miao Wang, and Haiyan Shi. "Tailored Recommendations Through Data Mining for Enriching Historical and Digital Cultural Tourism." (2024).

[15]. Siswipraptini, Puji Catur, et al. "Personalized Career-Path Recommendation Model for Information Technology Students in Indonesia." IEEE Access (2024).

[16]. Jomsri, Pijitra, et al. "Hybrid recommender system model for digital library from multiple online publishers." F1000Research 12 (2024): 1140.

[17]. Selvakumar, S., H. Inbarani, and P. Mohamed Shakeel. "A hybrid personalized tag recommendations for social e-learning system." International Journal of Control theory and applications 9.2 (2016): 1187-1199.

[18]. Ramakrishna, Mahesh Thyluru, et al. "HCoF: Hybrid Collaborative Filtering Using Social and Semantic Suggestions for Friend Recommendation." Electronics 12.6 (2023): 1365.

[19]. Chalkiadakis, Georgios, et al. "A novel hybrid recommender system for the tourism domain." Algorithms 16.4 (2023): 215.

[20]. Patro, S. Gopal Krishna, et al. "Cold start aware hybrid recommender system approach for E-commerce users." Soft Computing 27.4 (2023): 2071-2091.

[21]. https://www.kaggle.com/datasets/nandaprasetia/csv-500-20newsgroups