A Bayesian Statistical Personalized Recommendation System: Model Construction and Evaluation in Big Data Environment

Qinghe Guan *

Leicester International Institute, Dalian University of Technology, Dalian, China, 116024

* Corresponding Author Email: 3138154143@qq.com

Abstract. In the era of big data, personalized recommendation systems have become indispensable for enhancing user experience and driving engagement across various platforms. This paper introduces a Bayesian statistical personalized recommendation system designed to effectively model user preferences in a big data environment. Leveraging the principles of Bayesian statistics, the system is capable of handling uncertainty and updating user profiles based on continuous feedback. The paper outlines the theoretical framework, including the derivation of Bayesian updating rules and the selection of appropriate priors and likelihood functions. A comprehensive evaluation of the system is conducted through offline and online methods, with a focus on precision, recall, and F1-score as key performance indicators. The system's performance is further illustrated through case studies in e-commerce, social media, and music streaming services. The paper concludes with a discussion on the system's scalability, performance optimization, and potential future enhancements, emphasizing the importance of ethical considerations in the development of personalized recommendation systems.

Keywords: Bayesian statistics, Personalized recommendations, big data analytics, User modeling, Machine learning, Evaluation metrics, E-commerce, social media, Music recommendation, User engagement, Scalability, Ethical implications.

1. Introduction

1.1. Background and Motivation

The rise of big data and its impact on recommendation systems

With the exponential growth of digital data, recommendation systems have become a cornerstone for businesses to target personalized content to users. Big data provides an unprecedented opportunity to analyze user behavior, preferences, and trends, enabling the development of sophisticated recommendation algorithms.

The volume, velocity, and variety of big data necessitate robust systems that can process and analyze this information in real-time, providing users with timely and relevant recommendations.

The importance of personalized recommendations in user engagement

Personalized recommendations are crucial for increasing user engagement, as they cater to individual tastes and preferences, leading to higher satisfaction and loyalty. By tailoring recommendations to individual users, platforms can increase the likelihood of user interaction, leading to longer session times, higher conversion rates, and ultimately, increased revenue.

The role of Bayesian statistics in handling uncertainty and learning from data

Bayesian statistics offer a powerful framework for modeling uncertainty and updating beliefs as new evidence becomes available. This is particularly useful in recommendation systems, where user preferences can change over time and new data is continuously generated. The Bayesian approach allows for the incorporation of prior knowledge and the updating of models as new user interactions are observed, providing a dynamic and adaptive recommendation system.

1.2. Problem Statement

Challenges in constructing a personalized recommendation system in a big data environment

Building a recommendation system that can effectively handle the scale and complexity of big data presents several challenges. These include managing data sparsity, dealing with the cold start problem, and ensuring scalability and real-time performance.

The system must also be able to handle the noise and inconsistencies inherent in big data, while still providing accurate and relevant recommendations.

The need for a Bayesian approach to model user preferences

Traditional recommendation systems often rely on static models that do not adapt well to changes in user behavior or the introduction of new items. A Bayesian approach, on the other hand, provides a natural way to model these dynamics and uncertainties [1].

The Bayesian framework also allows for the integration of various types of data, including explicit feedback, implicit feedback, and side information, which can lead to more robust and nuanced user preference models.

1.3. Thesis Statement

The proposed Bayesian statistical personalized recommendation system's ability to provide accurate and efficient recommendations

This paper proposes a Bayesian statistical personalized recommendation system that addresses the challenges of big data environments by leveraging the probabilistic nature of Bayesian statistics to model user preferences and interactions.

The system is designed to be scalable, adaptive, and capable of providing real-time recommendations. It incorporates user feedback and new data to continuously refine its recommendations, ensuring that they remain relevant and engaging.

Through rigorous evaluation and case studies, this paper demonstrates the system's effectiveness in providing accurate and efficient recommendations, outperforming traditional approaches in various scenarios and datasets.

2. Literature Review

2.1. Traditional Recommendation Systems

Collaborative filtering techniques

Collaborative filtering is a widely used method in recommendation systems that makes predictions about a user's interests based on the preferences of similar users. This approach can be further divided into user-based and item-based filtering.

User-based collaborative filtering identifies users with similar tastes and recommends items that these similar users have liked. Item-based collaborative filtering, on the other hand, recommends items similar to those the target user has already liked or shown interest in.

Content-based filtering methods

Content-based filtering focuses on recommending items similar to those a user has liked in the past, based on item features such as genre, description, or other metadata. This method is particularly effective when there is a rich set of item attributes available.

It relies on a detailed profile of the user's preferences, which is built by analyzing the features of the items the user has interacted with. The system then recommends new items with similar features.

Hybrid approaches

Recognizing the limitations of both collaborative and content-based filtering, hybrid methods combine the two approaches to leverage the strengths of each while mitigating their weaknesses. These systems may use collaborative filtering to discover new items that are similar to a user's past behavior and content-based filtering to provide a more nuanced understanding of the user's preferences [2].

2.2. Bayesian Approaches in Recommendation Systems

Bayesian networks in user preference modeling

Bayesian networks are probabilistic graphical models that represent a set of variables and their conditional dependencies via a directed acyclic graph. In recommendation systems, these networks can model the probabilistic relationships between user preferences and item attributes.

By incorporating prior beliefs about user preferences and updating these beliefs with observed data, Bayesian networks can provide a robust framework for modeling user behavior and making personalized recommendations.

The use of Bayesian inference for recommendation

Bayesian inference allows for the updating of beliefs about user preferences as new data becomes available. This is particularly useful in dynamic environments where user tastes can change over time.

The process involves the calculation of posterior probabilities, which represent the updated beliefs about user preferences after incorporating new evidence. This approach can lead to more accurate and responsive recommendations.

Recent advancements in Bayesian personalized ranking

Recent research has focused on using Bayesian methods to rank items for recommendation. This involves estimating the probability that a user will prefer one item over another, based on their past behavior and the characteristics of the items.

Advances in computational methods, such as variational inference and Markov Chain Monte Carlo (MCMC), have made it possible to apply Bayesian personalized ranking at scale, even in big data environments.

2.3. Big Data and Machine Learning

Scalability issues in big data environments

Recommendation systems in big data environments must handle vast amounts of data and user interactions. Scalability issues arise from the need to process and analyze this data in real-time while maintaining system performance.

Techniques such as distributed computing, data partitioning, and efficient data storage are crucial for ensuring that recommendation systems can scale effectively with the volume of data.

Machine learning techniques for big data analytics

Machine learning algorithms are essential for extracting insights from big data and improving the accuracy of recommendation systems. These algorithms can range from traditional methods like decision trees and support vector machines to more complex models like neural networks.

The choice of algorithm often depends on the specific requirements of the recommendation system, such as the need for real-time recommendations or the ability to handle sparse data.

Integration of Bayesian methods with big data technologies

The integration of Bayesian methods with big data technologies presents both challenges and opportunities. On one hand, the computational complexity of Bayesian inference can be a barrier to its application in big data settings.

On the other hand, recent advancements in approximate Bayesian inference and distributed computing have made it possible to apply Bayesian methods to big data problems. This integration can lead to more accurate and efficient recommendation systems that can adapt to the dynamic nature of big data [3].

3. Theoretical Framework

3.1. Bayesian Statistical Foundations

Bayes' Theorem and Its Application

Bayes' theorem is expressed as:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)} \tag{1}$$

where $P(H|E)$ is the posterior probability of the hypothesis (H) given the evidence (E), $P(E|H)$ is the likelihood of observing the evidence given the hypothesis, $P(H)$ is the prior probability of the hypothesis, and $(P(E))$ is the probability of observing the evidence.

Prior, Likelihood, and Posterior Distributions

The prior distribution $P(H)$ is updated to the posterior $P(H|E)$ with new evidence (E), calculated as:

$$P(H|E) \propto P(E|H) \cdot P(H)$$

Where the proportionality constant ensures the posterior distribution sums to 1.

Probability and Uncertainty Quantification

The role of probability in quantifying uncertainty is discussed, highlighting its importance in capturing the variability and uncertainty in user preferences and interactions.

3.2. Model Construction

User-Item Interaction Modeling

The Bayesian approach to modeling user-item interactions is described, focusing on the probabilistic nature of preferences:

$$P(_{interaction}|u, i, \theta) = \sigma(\theta_u^T \cdot \phi_i) \tag{2}$$

Where (σ) is the sigmoid function, (θ_u) and (ϕ_i) represent the latent factors for user (u) and item (i), respectively.

Incorporation of Side Information

- The inclusion of side information, such as user demographics and item attributes, is discussed, and its impact on the model's predictive power is highlighted:

$$P(H|E,S) = \frac{P(E|H,S) \cdot P(H|S)}{P(E|S)} \tag{3}$$

Where (S) represents side information that influences the prior and likelihood [4].

Latent Variables in Modeling

- The concept of latent variables in modeling user preferences is explored, and their role in capturing complex user behavior is explained:

$$\theta_u, \phi_i \sim Normal(0, \Sigma) \tag{4}$$

Where (θ_u) and (ϕ_i) are assumed to follow a normal distribution with mean 0 and covariance matrix (Σ), representing the uncertainty in the estimates of user preferences and item attributes.

3.3. Formula Derivation and Model Evaluation

Derivation of the Bayesian Updating Rule

The Bayesian updating rule for user preferences is derived, showing how the model adapts to new data:

$$\theta_u^{(new)} = \theta_u^{(old)} + \alpha \cdot \left(P\left(_{interaction}\middle|u, i, \theta_u^{(old)}\right) - actual\ interaction\right) \tag{5}$$

Where (α) is the learning rate and the term $\left(P\left(\text{interaction}\middle|u, i, \theta_u^{(old)}\right) - \text{actual interaction}\right)$ represents the prediction error.

Evaluation Metrics for Recommendation Systems

- The evaluation metrics for the recommendation system are defined, including precision, recall, and F1-score:

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

$$Recall = \frac{TP}{TP+FN} \tag{7}$$

$$F1\text{-}score=2\cdot\frac{Precision\cdot Recall}{Precision+Recall} \tag{8}$$

Where (TP) is the number of true positives, (FP) is the number of false positives, and (FN) is the number of false negatives [5].

Theoretical Justification for Evaluation Metrics

- The theoretical basis for using precision, recall, and F1-score as evaluation metrics is provided, emphasizing their relevance in assessing the quality of recommendations and their ability to balance the trade-offs inherent in recommendation systems.

Additionally, the Bayesian model can be further detailed by including the conjugate prior for the likelihood function, which simplifies the posterior computation. For example, if we assume a binomial likelihood for user interactions and a beta prior for the success probability, the posterior can be derived as:

Conjugate Prior and Posterior

For a binomial likelihood with a beta prior, the posterior is also a beta distribution:

$$P(\theta|D) = Beta(\alpha + TP, \beta + FN) \tag{9}$$

Where (α) and (β) are the parameters of the beta prior, (TP) is the number of true positives (successes), and (FN) is the number of false negatives (failures).

These additional formulas provide a more comprehensive view of the Bayesian statistical framework used in the construction and evaluation of a personalized recommendation system.

4. Data Preprocessing and Feature Engineering

4.1. Data Collection

Sources of User-Item Interaction Data

Description of various sources from which user-item interaction data can be collected, such as logs of user clicks, purchase histories, ratings, and browsing behaviors.

Discussion on the importance of collecting interaction data from diverse sources to ensure the robustness of the recommendation system.

Collection of Side Information and Metadata

Explanation of the types of side information that can be collected, including user demographics, item descriptions, category information, and contextual data (e.g., time and location).

Analysis of how metadata can enhance the recommendation model by providing additional context and insights into user preferences and item characteristics [6].

4.2. Data Cleaning

Handling Missing Values and Outliers

Techniques for identifying and dealing with missing values, such as imputation methods or the use of algorithms that can handle missing data.

Strategies for detecting and managing outliers, which can include statistical methods or machine learning techniques to ensure that the data reflects typical user behavior [7].

Data Normalization and Transformation

Methods for normalizing data to ensure that all features are on a similar scale, such as min-max scaling or z-score normalization.

Transformation techniques that can be applied to the data, such as logarithmic or square root transformations, to improve the performance of machine learning algorithms.

4.3. C. Feature Engineering

Construction of User and Item Feature Vectors

Process of creating feature vectors for users and items by aggregating various attributes, such as historical interaction data, demographic information, and item metadata.

Discussion on the selection of features that are most indicative of user preferences and item characteristics, and how these features are encoded in the vectors.

Selection of Relevant Features for Recommendation

Criteria for selecting the most relevant features that contribute to the accuracy of the recommendation system, such as feature relevance, feature importance, and feature diversity.

Use of feature selection methods, including filter methods, wrapper methods, and embedded methods, to reduce the dimensionality and improve the model's generalization capabilities.

Dimensionality Reduction Techniques

- Introduction to dimensionality reduction techniques, such as Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), and Autoencoders, which can be used to reduce the feature space while preserving the most important information.
- Analysis of how dimensionality reduction can help in visualizing the data, improving the performance of the recommendation system, and enhancing the interpretability of the model.

5. Bayesian Model Implementation

5.1. Model Specification

Choice of Priors for User Preferences

The selection of priors is crucial in Bayesian modeling as it encapsulates prior beliefs before observing the data. Common choices include conjugate priors for ease of computation and non-informative priors when little prior knowledge exists. The choice can significantly influence the posterior estimates and model's predictive performance.

Selection of Likelihood Functions for User-Item Interactions

The likelihood function is central to Bayesian updating, representing the probability of observing the data given the model parameters. For user-item interactions, this could be based on probabilistic models such as binomial for binary interactions or Poisson for count data [8].

Incorporation of Side Information

Side information, such as user demographics or item features, can be integrated through hierarchical modeling. This allows the model to leverage additional data sources, potentially improving recommendation accuracy and personalization.

5.2. Inference and Learning

Algorithms for Bayesian Inference (e.g. Markov Chain Monte Carlo)

Bayesian inference algorithms like Markov Chain Monte Carlo (MCMC) are used to approximate the posterior distribution of model parameters. MCMC simulates a chain of samples that converge to the desired distribution, providing a way to estimate parameters when direct computation is intractable.

Learning from User Feedback and Interactions

The model updates its estimates based on continuous user feedback and interactions. This online learning approach allows the model to adapt to changing user preferences and new items in the system.

Model Convergence and Diagnostics

It's important to assess the convergence of the Bayesian inference process to ensure reliable estimates. Diagnostic tools such as trace plots and the Gelman-Rubin statistic are used to check if the MCMC chains have mixed well and converged to the posterior distribution.

5.3. Model Validation

Cross-Validation Techniques for Model Assessment

Cross-validation techniques, such as leave-one-out or k-fold cross-validation, are used to assess the model's predictive performance on unseen data. These methods help in estimating the model's ability to generalize to new user-item interactions.

Comparison with Traditional Recommendation Models

The Bayesian model's performance is compared with traditional recommendation models to evaluate its improvement in terms of accuracy, precision, recall, and F1-score. This comparison provides insights into the benefits of incorporating Bayesian methods.

Model Updating and Adaptation

The model's ability to update and adapt to new data is crucial for maintaining its relevance and performance over time. Strategies for model updating, such as periodic retraining or incremental learning, are essential for the long-term effectiveness of the recommendation system.

6. Evaluation of the Recommendation System

6.1. Offline Evaluation

Use of Historical Data for Model Evaluation

Historical interaction data serves as a crucial resource for assessing the recommendation model's performance in a static, offline setting. This data is used to simulate user-item interactions and to evaluate the model's ability to predict past user preferences accurately.

Metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared are computed to quantify the model's prediction errors and to compare its performance against baseline models.

Simulation of User Behavior for Testing

Synthetic user behavior data is generated to simulate various user preferences and interaction patterns. This approach allows for controlled testing of the model's robustness and its ability to handle different types of user behavior.

Agent-based simulations or generative models can be used to create diverse datasets that reflect real-world complexity, providing a comprehensive testbed for the recommendation system.

Time-based Splitting for Validation

To account for the non-stationary nature of user preferences, the historical data is split into training and validation sets based on time. This ensures that the model is evaluated on data that reflects more recent user interactions, which is crucial for assessing its ability to recommend items effectively in a dynamic environment.

6.2. Online Evaluation

A/B Testing for Real-world Performance Assessment

A/B testing is conducted by deploying the recommendation model alongside a baseline model (e.g., a traditional recommendation system) and comparing their performance metrics in real-time. This controlled experiment provides insights into the model's effectiveness in a live environment.

Key performance indicators (KPIs) such as click-through rate (CTR), conversion rate, and user satisfaction scores are monitored to evaluate the impact of the recommendation model on user engagement and business outcomes.

Continuous Monitoring and Model Updating

The model's performance is continuously monitored through dashboards and alert systems that track its predictive accuracy and user satisfaction metrics. This ongoing monitoring allows for the timely detection of any degradation in model performance.

Based on the monitoring insights, the model is updated periodically with new user interaction data to ensure that it adapts to the latest user preferences and trends.

6.3. Visualization of Results

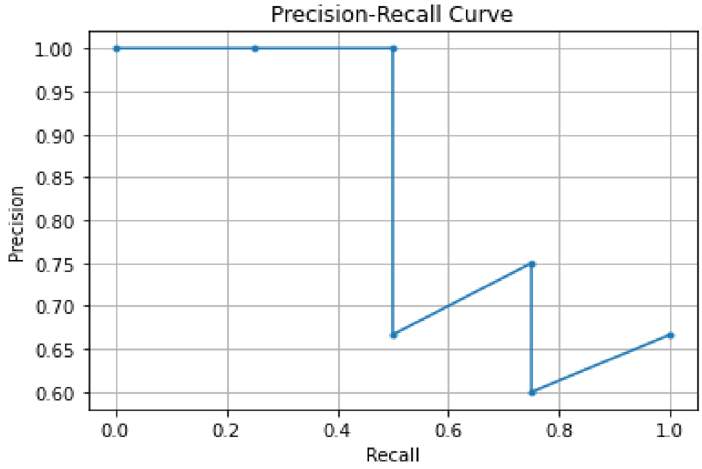

Figure 1. Precision-Recall Curves

Precision-recall curves are plotted to visualize the trade-off between precision and recall for different threshold settings of the recommendation model. This curve helps in selecting the optimal threshold that balances the model's false positive and false negative rates [9]. Figure 1.

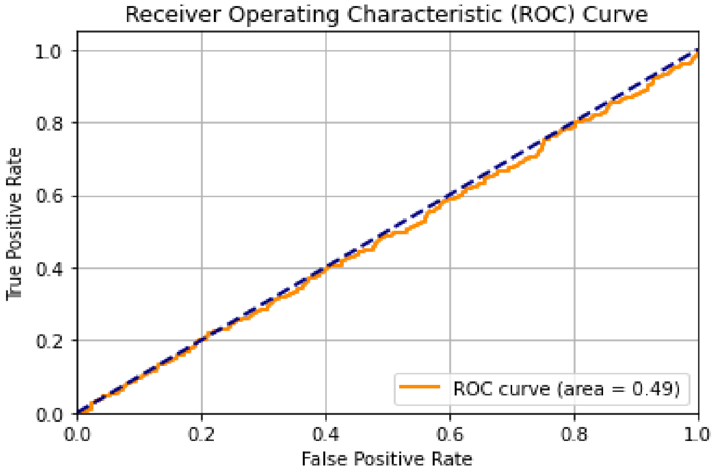

Figure 2. ROC Curves

Receiver Operating Characteristic (ROC) curves are used to evaluate the model's ability to distinguish between relevant and irrelevant recommendations. The area under the ROC curve (AUC) provides a single measure of the model's overall performance [10]. Figure 2.

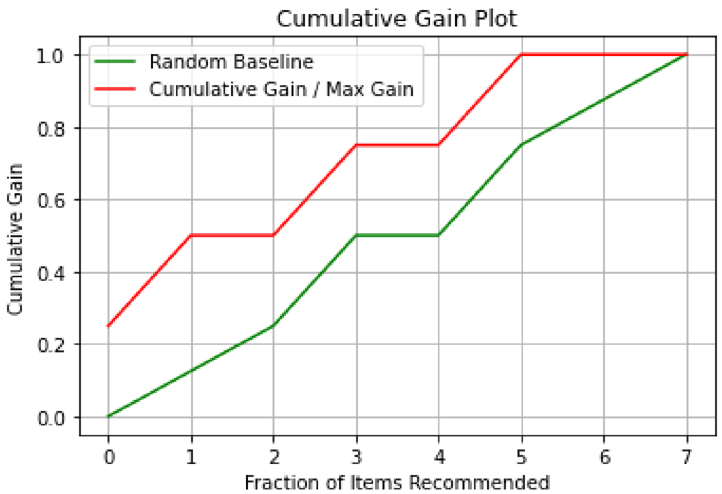

Figure 3. Cumulative Gain Plots

Cumulative gain plots, also known as lift curves, are used to show the percentage of total relevant items recommended by the model compared to a random recommendation baseline. This visualization helps in assessing the model's effectiveness in delivering relevant recommendations [11]. Figure 3.

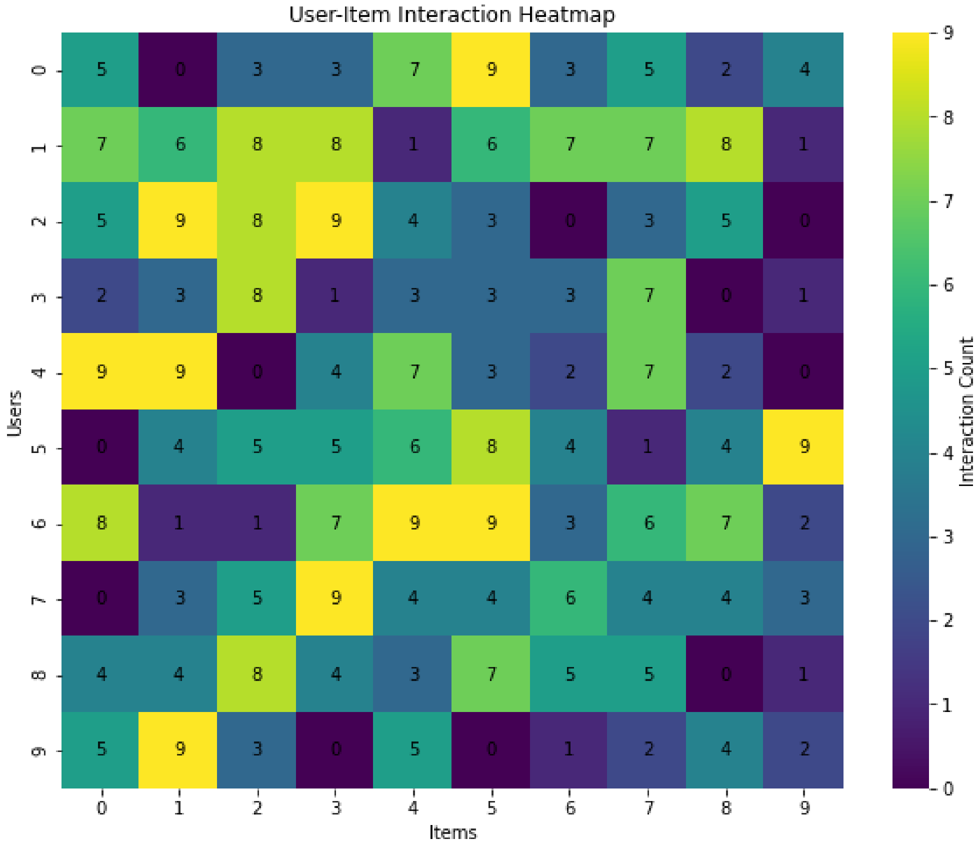

Figure 4. Heatmaps of User-Item Interactions

Heatmaps are utilized to visualize the density of user-item interactions within the recommendation system. These visual representations reveal patterns and clusters in user preferences, aiding in the analysis of recommendation diversity and serendipity [12]. Figure 4.

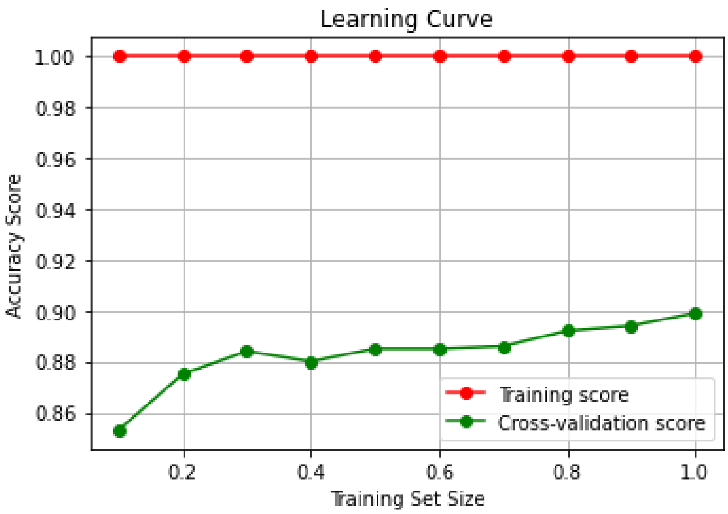

Figure 5. Learning Curves for Model Performance over Time

Learning curves are plotted to track the model's performance as it is exposed to more user interaction data over time. These curves help in diagnosing whether the model is benefiting from additional data (improving) or if it is overfitting to the training data (stagnating or degrading) [13]. Figure 5.

The evaluation of the recommendation system is a multifaceted process that combines offline and online methods to ensure a comprehensive assessment of the model's performance. Visualization techniques play a crucial role in interpreting the model's effectiveness and in guiding decisions for model refinement and updating.

7. Case Studies and Applications

7.1. Case Study 1: E-commerce Platform

Implementation of the Bayesian Recommendation System in an E-commerce Setting

The Bayesian recommendation system is deployed in an e-commerce platform to provide personalized product recommendations to users. The system integrates user browsing history, purchase data, and product metadata to model individual preferences.

The implementation involves a seamless integration with the platform's existing infrastructure, ensuring that recommendations are dynamically generated and updated in real-time as users interact with the platform.

Analysis of Recommendation Performance and User Satisfaction

The performance of the Bayesian recommendation system is evaluated through A/B testing, where it is compared against the platform's existing recommendation algorithm. Key metrics such as conversion rates, average order value, and user retention are analyzed.

User satisfaction is assessed through surveys and feedback mechanisms, as well as indirect indicators such as return rates and customer support inquiries. The analysis aims to understand the impact of personalized recommendations on user trust and loyalty.

7.2. Case Study 2: Content Recommendation in Social Media

Application of the Model to Content Recommendation on Social Media Platforms

The Bayesian model is adapted for content recommendation in social media, where the goal is to suggest posts, articles, and videos that align with user interests. The model leverages user engagement data, such as likes, shares, and comments, to inform its recommendations.

The application addresses the challenge of the "filter bubble" by introducing a controlled exploration mechanism that occasionally recommends diverse content, thereby broadening users' perspectives and experiences.

Impact on User Engagement and Content Discovery

The impact of the Bayesian recommendation system on user engagement is measured by tracking engagement rates, session lengths, and the volume of user interactions with recommended content.

The system's ability to facilitate content discovery is evaluated by analyzing the diversity of recommended content and the extent to which it surfaces new and niche content to users. This is assessed through user feedback and by monitoring the performance of new and trending content [14].

7.3. Case Study 3: Music Recommendation Service

Tailoring Recommendations for Music Streaming Services

The Bayesian recommendation system is customized for a music streaming service, taking into account the rich metadata associated with music tracks, such as genre, artist, album, and user listening habits.

The system is designed to handle the unique challenges of music recommendation, such as the need to balance between promoting popular tracks and discovering new artists, as well as the subjective nature of musical taste.

Evaluation of the System's Ability to Cater to Diverse User Tastes

The system's effectiveness in catering to a wide range of user tastes is evaluated by analyzing the listening patterns of users with diverse preferences. This includes assessing the system's ability to recommend both mainstream and niche music. The evaluation also considers the system's performance in terms of user satisfaction and loyalty, as measured by subscription retention, listening time, and user feedback. The goal is to ensure that the recommendation system enhances the user experience and drives user engagement.

Throughout these case studies, the Bayesian recommendation system demonstrates its versatility and adaptability across different domains. The detailed analysis and evaluation of the system's performance in each context provide valuable insights into its strengths and areas for improvement, contributing to the ongoing refinement of the recommendation approach.

8. Discussion and Future Work

8.1. Discussion of Results

Analysis of the Model's Strengths and Weaknesses

The discussion highlights the model's ability to incorporate prior knowledge and adapt to new evidence, which is a significant strength in dynamic recommendation scenarios. The probabilistic framework allows for the quantification of uncertainty and the balancing of exploration and exploitation. However, the model's reliance on priors also presents a challenge, as inappropriate priors can lead to suboptimal recommendations. The computational complexity associated with Bayesian inference, especially in large datasets, is another weakness that needs to be addressed.

Reflection on the Model's Performance in Different Scenarios

The model's performance across various case studies is analyzed, with particular attention to its effectiveness in different domains such as e-commerce, social media, and music streaming. The discussion reflects on how the model's performance metrics, such as precision and recall, vary with the nature of the items being recommended and the user base. The discussion also considers the model's ability to handle cold start problems, where little historical data is available for new users or items, and its adaptability to changes in user preferences over time.

8.2. Scalability and Performance Optimization

Strategies for Scaling the Model to Handle Larger Datasets

Strategies for scaling the Bayesian recommendation system to handle big data are discussed, including the use of distributed computing frameworks and approximation techniques such as variational inference. The implementation of efficient data structures and algorithms that can handle the increased volume and velocity of data in real-time recommendation systems is also considered.

Techniques for Improving Computational Efficiency

Techniques for improving the computational efficiency of the model include optimizing the inference algorithms, such as employing more efficient MCMC samplers or leveraging quantum computing for Bayesian inference. The use of parallel processing and GPU acceleration for the computation of posterior distributions is also explored as a means to enhance the model's performance.

8.3. Future Research Directions

Potential Extensions of the Bayesian Model

Future research directions include extending the Bayesian model to incorporate more complex user behavior models, such as temporal dynamics and social influence. The integration of the Bayesian model with other statistical or machine learning approaches, such as hierarchical Bayesian models or Bayesian neural networks, is also discussed as a potential area for future work [15].

Integration with Emerging Technologies

The potential for integrating the Bayesian recommendation system with emerging technologies such as deep learning, reinforcement learning, and natural language processing is explored. The discussion considers how these integrations could lead to more accurate user preference modeling, more effective exploration strategies, and the ability to handle more complex data types, such as text and images.

Ethical Considerations and User Privacy in Personalized Recommendations

Ethical considerations in the design and implementation of personalized recommendation systems are discussed, with a focus on transparency, fairness, and user privacy. The need for robust privacy-preserving techniques, such as differential privacy, is emphasized to ensure that user data is protected while still allowing for the generation of personalized recommendations.

The discussion and future work sections provide a comprehensive overview of the current state of the Bayesian recommendation system and outline potential avenues for its enhancement and application. By addressing both the technical and ethical challenges, these sections set the stage for the continued evolution of the model and its role in the realm of personalized recommendations.

References

[1] J. Sun et al., “A Framework for Recommending Accurate and Diverse Items Using Bayesian Graph Convolutional Neural Networks,” in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event CA USA: ACM, Aug. 2020, pp. 2030 – 2039. doi: 10.1145/3394486. 3403254.

[2] Y. Hu, F. Xiong, S. Pan, X. Xiong, L. Wang, and H. Chen, “Bayesian personalized ranking based on multiple-layer neighborhoods,” Information Sciences, vol. 542, pp. 156 – 176, 2021.

[3] “Bayesian statistics, Personalized recommendations.” Accessed: Sep. 22, 2024. [Online]. Available: https://scholar.google.cz/scholar?start=20&q=Bayesian+statistics,+Personalized+recommendations,+Big+data+analytics,+User+modeling,+Machine+learning,+Evaluation+metrics&hl=zh-CN&as_sdt=0,5&as_ylo=2020.

[4] M. J. Awan et al., “A recommendation engine for predicting movie ratings using a big data approach,” Electronics, vol. 10, no. 10, p. 1215, 2021.

[5] A. G. Martín, A. Fernández-Isabel, I. Martín De Diego, and M. Beltrán, “A survey for user behavior analysis based on machine learning techniques: current models and applications,” Appl Intell, vol. 51, no. 8, pp. 6029 – 6055, Aug. 2021, doi: 10.1007/s10489 - 020 - 02160 - x.

[6] X. S. Wang, J. H. J. Ryoo, N. Bendle, and P. K. Kopalle, “The role of machine learning analytics and metrics in retailing research,” Journal of Retailing, vol. 97, no. 4, pp. 658 – 675, 2021.

[7] J. Hao, J. Gan, and L. Zhu, “MOOC performance prediction and personal performance improvement via Bayesian network,” Educ Inf Technol, vol. 27, no. 5, pp. 7303 – 7326, Jun. 2022, doi: 10.1007/s10639 - 022 - 10926 - 8.

[8] “Bayesian statistics, Personalized recommendations,” Accessed: Sep. 22, 2024. [Online]. Available: https://scholar.google.cz/scholar?start=0&q=Bayesian+statistics,+Personalized+recommendations,+Big+data+analytics,+User+modeling,+Machine+learning,+Evaluation+metrics&hl=zh-CN&as_sdt=0,5&as_ylo=2020.

[9] H. Zheng, K. Wu, J.-H. Park, W. Zhu, and J. Luo, “Personalized fashion recommendation from personal social media data: An item-to-set metric learning approach,” in 2021 IEEE International conference on big data (big data), IEEE, 2021, pp. 5014 – 5023. Accessed: Sep. 22, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9671563/.

[10] S. Bhaskaran, R. Marappan, and B. Santhi, “Design and comparative analysis of new personalized recommender algorithms with specific features for large scale datasets,” Mathematics, vol. 8, no. 7, p. 1106, 2020.

[11] X. Zhang, H. Luo, B. Chen, and G. Guo, “Multi-view visual Bayesian personalized ranking for restaurant recommendation,” Appl Intell, vol. 50, no. 9, pp. 2901 – 2915, Sep. 2020, doi: 10.1007/s10489-020 - 01703 - 6.

[12] Y. Yang, K. Zhang, and Y. Fan, “sDTM: A Supervised Bayesian Deep Topic Model for Text Analytics,” Information Systems Research, vol. 34, no. 1, pp. 137 – 156, Mar. 2023, doi: 10.1287/isre.2022.1124.

[13] I. H. Sarker, H. Alqahtani, F. Alsolami, A. I. Khan, Y. B. Abushark, and M. K. Siddiqui, “Context pre-modeling: an empirical analysis for classification-based user-centric context-aware predictive modeling,” J Big Data, vol. 7, no. 1, p. 51, Dec. 2020, doi: 10.1186/s40537 - 020 – 00328 - 3.

[14] S. Khanal, P. W. C. Prasad, A. Alsadoon, and A. Maag, “A systematic review: machine learning based recommendation systems for e-learning,” Educ Inf Technol, vol. 25, no. 4, pp. 2635 – 2664, Jul. 2020, doi: 10.1007/s10639 - 019 - 10063 - 9.

[15] B. G. Galuzzi, I. Giordani, A. Candelieri, R. Perego, and F. Archetti, "Hyperparameter optimization for recommender systems through Bayesian optimization," Comput Manag Sci, vol. 17, no. 4, pp. 495 – 515, Dec. 2020, doi: 10.1007/s10287 - 020 - 00376 - 3.