Flipkart

GRiD 5.0

Problem Statement Title:
Personalized Product Recommendations

Team Name:
Apocalypse

Team members details

Team Name	Apocalypse		
Institute Name/Names	Graphic Era Deemed to be University		
Team Members >	**1 (Leader)**	**2**	**3**
Name	Geetansh Verma	Sahil Gupta	Vishal Kumar
Batch	2020-24	2020-24	2020-24

Deliverables/Expectations for Level 2 (Idea + Code Submission)

"Imagine a recommendation system that anticipates your needs without bothering you for inputs. Our application achieves just that. It silently observes your interactions on the website, understanding your preferences effortlessly offering more tailored shopping experience."

- Easy-to-use and hassle-free experience because there is no need for user input.
- With observational intelligence, the system learns from observing how users interact with your website, ensuring a user-centric approach.
- Advanced Methods: For precise product recommendations, we make use of potent AI technologies like GANs, Neural Collaborative Filtering, and Transformers.
- Recommendations are personalized based on your past behavior and product preferences thanks to data-driven insights.

- Build on the MERN stack of technology, with Flask API for seamless backend interaction and communication with the recommendation model.

- Architecture that can be easily transferred between different technology stacks because it is not dependent on any one particular machine.

- Plug and Play: E-commerce behemoths like Flipkart can quickly and easily integrate our system while customizing recommendations based on their dataset.

- User-Friendly Design: The application is laid out simply to make it usable by a variety of users.

- Engineered to handle heavy usage, making it suitable for integration with leading e-commerce platforms, this feature is known as "scalability ready."

- Please refer to the Readme file in the ZIP folder for more information on the detailed documentation.

Use-cases

- The primary use case of this application is to personalize the recommendation system.
- It is easily extensible and modifiable for use in other industries, such as cooperative tools to cut costs and focus on ROI (Return On Interest) rather than maintaining historical data and user ratings.
- Marketing firms can make better use of their marketing budget by using it to increase traffic.
- It can be used by social media giants to make their sites interactive.

Solution statement/ Proposed approach

- Due to the scarcity of review data in specific categories to train deep learning model.Studies have employed Generative based augmentation to synthetically expand the dataset.

- For this reason, in our paradigm, we try to use a small static portion of category specific generative augmented learning.

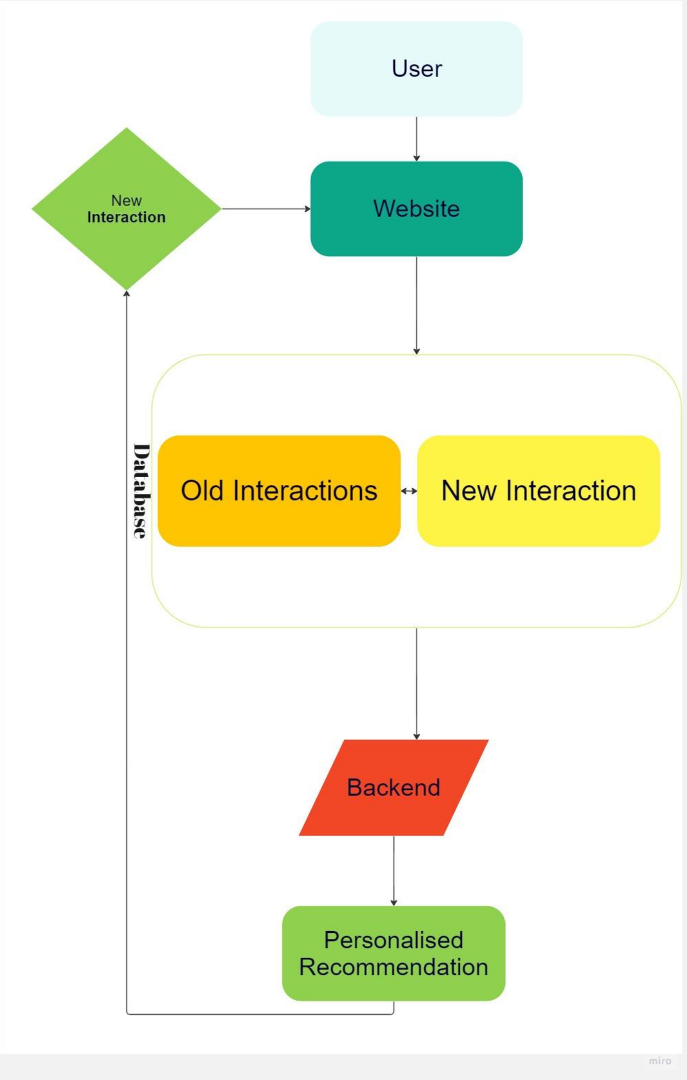
- However, as in the case of minority over sampling with traditional methods, it doesn't focus on the specific category thus being ineffective and inducing an unhealthy amount of synthetic data into the final dataset.

- In order to achieve this, we trained a different model for each of the seven categories and chose the two categories with the lowest accuracy (Gift Cards and Jewelry Categories).

- Using a GPT-2 based architecture, we then only augmented the two worst categories while leaving the other categories intact (No augmentation).

- The synthetic data has to be cleaned before being used for training because we can't rely on augmented data with closed eyes.
- So, we cleaned it using Power BI.

- Then, using this dataset, we built the recommendation system by combining the original data with the augmented data.



ARCHITECTURE:

GenAI Augmentation using GPT-2:

- "This code automates the generation of synthetic customer reviews of products. It works in a systematic manner:

- Data generation: Randomly chosen data points from a dataset are used to create synthetic reviews. Information about customers and products is among these points.
  Text Generation: Based on the information gathered, a text generator generates product ID, product title, timestamp and rating that offers a thorough analysis.

- Data Structuring: To create structured review data, extracted information from the generated text is combined with customer and product identifiers.

- Synthetic review dates are generated and assigned to the reviews, similar to what was stated above.

- The structured data and review dates are arranged into DataFrames and then combined to create a consolidated dataset.

- Data cleaning: To improve the dataset, data is cleaned, missing values are handled, and new data is merged.

- Final Result: The enriched dataset that was produced includes artificial product reviews with various attributes.

- By eliminating the need for manual data collection, this method makes it possible to create a comprehensive dataset for analysis, modeling, or testing.

Transformer based recommendation:

- The exploration of a robust recommender system using PyTorch Lightning is covered in the algorithm.

- At its core, the Recommender model uses embedding layers to transform item IDs into useful embeddings.

- For the purpose of unraveling complex patterns in user-item interactions, it uses a stack of transformer encoder layers. Items are ranked by a linear output layer, and generalization is ensured by dropout.

- Utilizing unique loss functions and optimization strategies, training involves minimizing errors.

- The model's performance is assessed during validation and testing, enabling real-time deployment with saved checkpoints.

- This strategy makes use of transformer models and deep learning to deliver personalized recommendations for increased user engagement.

Neural Collaborative filtering recommendation:

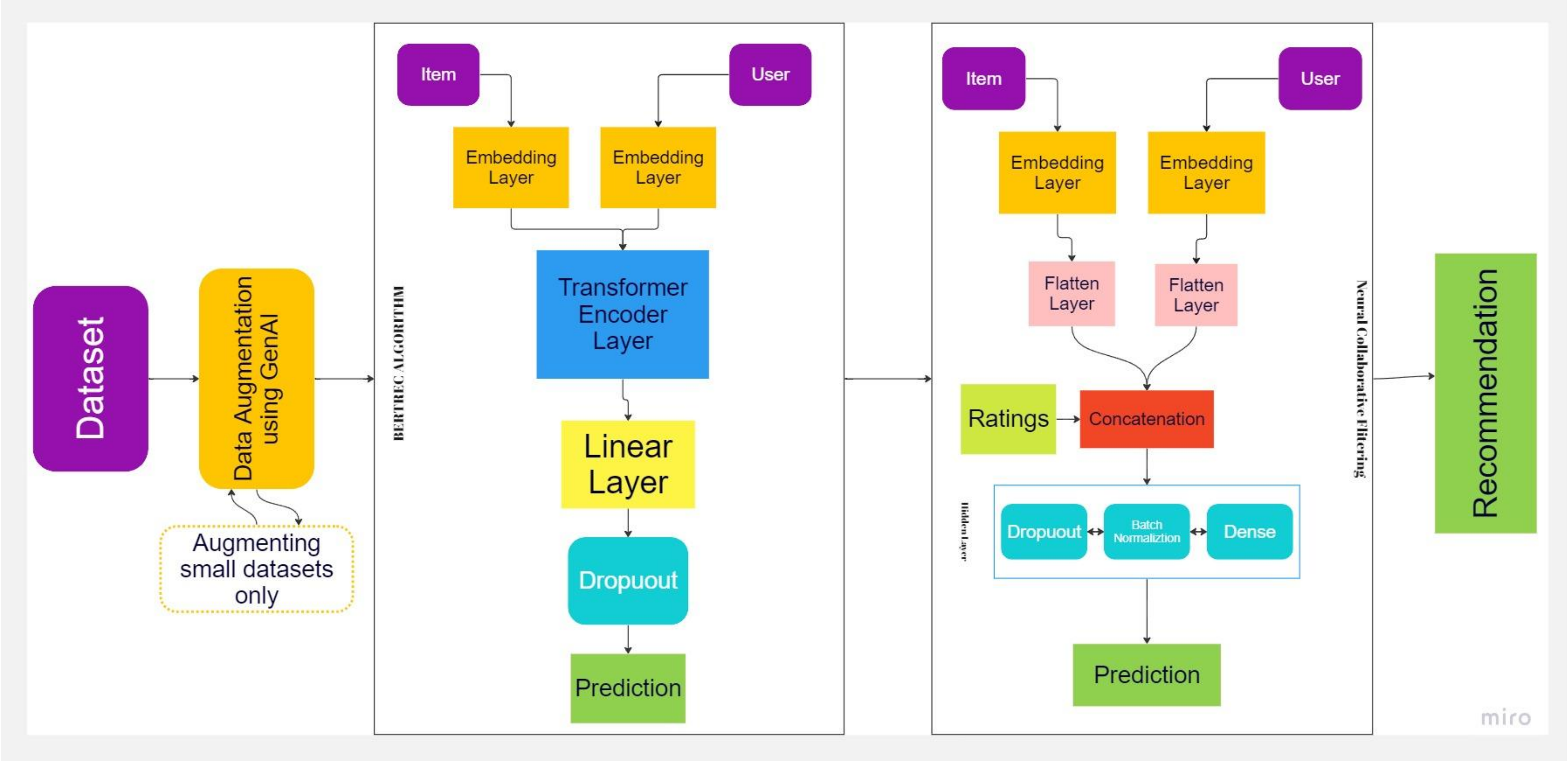
- The development of a collaborative filtering recommender system using Keras is outlined in this algorithm.

- The system's goal is to provide individualized recommendations by examining user preferences and behavior.

- The input, embedding, flatten/concatenate, hidden, and output layers make up the model architecture. The model produces predictions and is trained and evaluated using mean squared error.

- These forecasts are rated and then interpreted using the original IDs. The system's effectiveness is demonstrated by its capacity to anticipate and suggest ratings, improving the quality of recommendations.

- The importance of evaluating recommendations against actual ratings, the dependence of collaborative filtering on user behavior, and the role of deep learning in improving recommendations are highlighted as key takeaways.



With the sequential fusion of all these algorithms we built a model that is recommending personalized products to the user.

Limitations

- We were unable to expand the model's capabilities by adding more categories due to a lack of computational resources.
- Due to scarcity of time, we had to train several different recommendation models; however, in the future, we'll concentrate on creating a streamlined, lightweight model.
- The frontend and backend components are both waiting to be optimized, and our plan makes use of AWS Lambda to create standardized robustness.

Future Scope

- More product categories will be added so that users can select from a wider range of products with better recommendations.

- We will maintain the historical analysis of each individual person using better database-based paradigms, and we will use a time series forecast model to amplify the outcome.

- We will attempt to implement a system that incorporates our model so that e-commerce websites like Flipkart can integrate it into their system.

Flipkart

GRiD 5.0

Thank You