AI-DRIVEN PROFIT OPTIMIZATION

RECOMMENDATION ENGINE

Introduction

The Product Recommendation Framework provides a blueprint to implement personalised product recommendations that deliver measurable impact on revenue generation. The recommender engine leverages Azure cloud technologies and sophisticated reinforcement algorithms to intelligently predict which products or content should be suggested to customers. By analysing patterns and trends within large datasets, the recommender engine can identify subtle correlations and preferences that might otherwise go unnoticed. Customer interactions are linked with feedback loops to continuously improve algorithmic accuracies.

The unique coding framework on which the recommender engine is based, allows companies to leverage their customer and sales data at scale to generate customized recommendations. Solution integration is available via Rest APIs or Batch model serving.

The recommendation framework is also designed to solve critical modelling challenges faced in the industry such as the cold start problem and data sparsity leveraging unique data modelling techniques.

Studies show that 91% of consumers are more likely to shop with brands that remember their preferences and provide relevant recommendations. Companies such as Amazon drive about 35% of its eCommerce revenue from product recommendations while Netflix drive 75% of its revenue from these type of recommender engines. To maintain a competitive edge in these evolving markets, it is essential for your company to prioritize investment in the integration of recommendation engine technologies.

Attainable KPI's

This Recommender engine play a pivotal role in driving key performance indicators (KPIs) across various dimensions of business success. Through personalized recommendations tailored to individual preferences, this engine can significantly impact metrics such as conversion rates, customer engagement and loyalty as well as improved customer sentiment.

- ### Increasing Sales and Conversion Rates

The Recommendation engine allows retailers to suggest relevant products to customers based on their purchase behaviour, browsing history and demographics. This system is also capable of accounting for user states, such as, at what time of day, week, month, year was the transaction made and how should the recommendations be augmented to account for these nuances. By serving these personalized recommendations on product pages, banners and emails, retailers can effectively upsell and cross-sell products. This in turn leads to increased product sales and conversion rates.

- ### Enhancing Customer Engagement and Loyalty

Integrating the recommendation engine into business operations not only cultivates brand loyalty in a competitive market but also lays the foundation for forging enduring customer relationships. Achieving personalized experiences through product recommendations tailored to customer interests, past purchases, and browsing behaviour drives enhanced engagement, encourages repeat purchases, and fosters unwavering customer loyalty.

- ### Improving overall customer Sentiment

When customers receive product recommendations that resonate with their preferences, interests, or needs, they experience increased satisfaction with the shopping experience. This personalized approach demonstrates the brand's attentiveness to their individual desires, fostering a deeper connection and sense of value. As a result, customers develop a more positive perception of the brand, leading to enhanced overall customer sentiment. This positive sentiment not only encourages repeat purchases but also fuels word-of-mouth recommendations and advocacy, further bolstering the brand's reputation and success in the market.

Solving Critical Modelling Challenges

Addressing issues such as the cold start problem and data sparsity in this recommender engine is crucial for maximizing the effectiveness and impact of personalized recommendations. By overcoming these challenges, the recommender engine can provide more accurate and relevant suggestions to users, even in scenarios where limited data is available.

- ### The Cold Start Problem

Recommenders typically struggle when dealing with new users or items that lack sufficient historical data. The cold-start problem occurs when there isn't enough information to make accurate recommendations for these entities. One approach used to overcome this challenge is to segment retail customers. For new users where no historical product sales exist, optimized recommendations linked to a users' respective segment is proposed. In doing so, the recommender allows the new users to start interacting with the recommendations. With each interaction, the underlying product distributions are updated allowing for more personalised recommendations over time. A random selection algorithmic approach is also integrated into the solution to account for the addition of new items within the recommender framework. This algorithm allows for random exploration of new items unfamiliar to the users, while updating the data distributions as the users interact with the respective items.

- ### Scalability

As the digital landscape continues to expand and the volume of data generated by consumers escalates exponentially, the demand for efficient recommender systems capable of handling large datasets becomes increasingly imperative. Scalability emerges as a pivotal consideration to ensure that recommendation engines can process and serve batch or real-time recommendations without sacrificing performance or responsiveness. Leveraging cutting-edge technologies such as Azure Databricks, PySpark, and MLFlow, businesses can develop robust recommendation engines equipped to manage vast repositories of customer and transactional data with ease and efficiency. The flexibility inherent in these technologies allows recommendation engines to tailor their outputs according to client requirements, whether generating recommendations on a per-user basis or aggregating insights to deliver recommendations at a segment level. By leveraging the scalability and versatility afforded by Azure Databricks, PySpark, and MLFlow, businesses can develop recommendation engines that not only accommodate the growing volume of data but also deliver highly personalized and actionable recommendations that drive engagement, satisfaction, and ultimately, business success.

The Solution Approach

The recommendation engine leverages a diverse range of data sources, including sales frequency, recency, and monetary value, as well as channel and browsing activity, customer demographics, and product preferences. Furthermore, temporal factors such as year, month, week, day, and seasonality are incorporated into the modelling framework to provide comprehensive contextual information. The primary objective of the recommender engine is to deliver personalized recommendations to customers, ensuring that critical user context is not

- overlooked to avoid suboptimal recommendations.

To generate State-IDs for the recommendation engine, an unstructured machine learning approach is employed. Previous data points serve as input for this framework, with the models producing user segments as output. These segments are then transformed into State-IDs and utilized as input for the Contextual Multi-Arm Bandit algorithm, which governs the decision-making process regarding which products or product lists should be recommended to specific States or Users.

As part of the implementation process, we utilize an A/B testing or Champion-Challenger framework, augmented by the inclusion of potential hold-out groups. By leveraging these testing frameworks, we ensure the effective monitoring and evaluation of the recommendation engine's performance.

Product Features

The recommendation engine offers a comprehensive recommender framework aimed at enhancing customer engagement and boosting overall company revenue. Notable features encompass:

- Customer segmentation and analytics for targeted marketing strategies.
- Tailored product recommendations specific to individual segments or users.
- Monitoring of recommendation engine KPIs and quantification of value generated.
- Flexibility through Rest API or Batch serving capability, facilitating seamless integration with business applications.

Choose from a Core or Enhanced Implementation

The Core plan includes the ingestion, processing and parsing of customer data. Historical analysis and trends as well as customer segmentation. The outcomes of the Core plan are the development of a “Customer Like You” strategy to inform the company of the type of personas within the customer base and potential strategies to optimize user engagement.

The Enhanced plan includes everything in the Core plan PLUS the development of the Recommender Engine. This entails setting up State Definitions, Architecture setup, Algorithm selection and training, API or Batch serving and KPI tracking/ monitoring.

Both plans include model monitoring and maintenance services, documentation as well as frequent updates/ ecosystem improvements.