International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 11 Issue 11 ǁ November 2023 ǁ PP. 55-58

Building a Scalable REST API for Customer Churn Prediction using Spring Boot and H2O Auto ML on Telco Customer Churn Dataset

Aishwarya Sura Ravi
*[*1]*

Abstract
Customer churn is a major problem for businesses, especially in the highly competitive telecommunications industry. Predicting customer churn is a critical task for businesses to retain their existing customers and reduce the cost of acquiring new ones. The Telco Customer Churn dataset is a publicly available dataset that contains information about customer demographics, services subscribed, contract details, and churn status. In this project, we aim to predict customer churn using machine learning techniques and the Telco Customer Churn dataset. We will use Spring Boot to build a REST API that takes in customer data and predicts the probability of churn using XGBoost or Recurrent Neural Networks.

Keywords: *Spring Boot,H20 Auto ML,machine learning*

Date of Submission: 01-11-2023 Date of acceptance: 10-11-2023

I. INTRODUCTION

The Telco Customer Churn dataset is a well-known and widely used resource for practicing data analysis and training machine learning algorithms. The dataset contains information about customer demographics, services subscribed, contract details, and churn status. In this project, we will use Spring Boot to build a REST API that takes in customer data and predicts the probability of churn using XGBoost or Recurrent Neural Networks. XGBoost is a tree-based ensemble method that is popular for classification and regression problems. Recurrent Neural Networks are a type of neural network that can learn complex patterns in sequential data. By using machine learning techniques and the Telco Customer Churn dataset, businesses can predict customer churn and take necessary actions to reduce it.

II. SOFTWARE USED

A. Spring Boot

Spring Boot is an open-source Java framework that simplifies and accelerates the development of production-ready, stand-alone Spring applications. It provides a set of pre-configured templates and tools, enabling developers to quickly build web applications, microservices, and RESTful APIs. Spring Boot eliminates the need for complex XML configuration and focuses on convention over configuration, making it easier to get started with Spring development. It offers embedded servers like Tomcat or Jetty, reducing the need for external server configuration. Spring Boot's ecosystem includes a wide range of extensions and integrations for building various types of applications, and it's widely used in the development of modern, scalable, and efficient Java applications.

B. ReactJS

React.js is an open-source JavaScript library developed and maintained by Facebook. It is widely used for building user interfaces for web applications. React is known for its component-based architecture, which allows developers to create reusable and modular UI components, making code maintenance and development more efficient. It utilizes a virtual DOM to improve performance by minimizing the need to directly manipulate the actual DOM, resulting in faster rendering and better user experiences. React is often used in combination with other tools and libraries, such as Redux for state management, and it has a large and active community of developers contributing to its ecosystem.

C. H20 AutoML

H2O AutoML is a user-friendly machine learning platform that automates the process of building and training machine learning models. It's designed to make it easier for non-experts to harness the power of machine learning. With H2O AutoML, you can simply provide your data, and it will automatically explore various machine learning algorithms and hyperparameters to find the best model for your task. It's particularly useful for tasks like

classification and regression. This approach saves time and effort, making machine learning more accessible to a wider audience.

D. Java

Java is a popular and versatile programming language used for building a wide range of applications. It's known for its platform independence, meaning you can write code once and run it on various devices and operating systems. Java uses a "write once, run anywhere" philosophy, making it suitable for web applications, mobile apps, and server-side development. It's an object-oriented language, which means it organizes code into reusable components called objects. Java is also famous for its robust security features, making it a good choice for building secure and reliable software.

III. ABOUT DATASET

The dataset is used to predict customer behavior for the purpose of retaining them, and it's particularly relevant in the context of a telecom company. It empowers data analysts to examine a comprehensive set of customer information, enabling the development of targeted customer retention strategies.

This dataset contains one row for each customer, with each column representing specific attributes as outlined in the metadata for that column. The data provides insights into the following aspects:

1. Churn Status: It reveals whether customers have recently departed, as indicated by the "Churn" column. This information is critical for assessing customer attrition.

2. Services Subscribed: Details of the services each customer has subscribed to, including phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies.

3. Customer Account Information: This section encompasses various aspects of a customer's account, such as their tenure as a customer, contract type, preferred payment method, inclination toward paperless billing, and records of monthly charges and total charges incurred.

4. Demographic Insights: The dataset also provides valuable demographic information about customers, including their gender, age range, and whether they have partners and dependents.

Overall, this dataset equips analysts and data scientists with the information necessary to make informed decisions about customer retention strategies within the telecom industry, thereby reducing customer churn.

IV. PROPOSED METHODOLOGY

1. Data Collection: the Telco Customer Churn Dataset was collected, which contained customer information and churn data. This dataset served as the foundation for the predictive model.

2. Data Preprocessing: Clean and prepare the data by handling missing values, encoding categorical variables, and normalizing numerical features. This step ensures the data is in a suitable format for machine learning.

3. Feature Selection:Identify the most relevant features that can impact customer churn prediction. This step helps in reducing complexity and improving model performance.

4. Model Building with H2O AutoML:
- Utilized H2O AutoML, a user-friendly machine learning tool, to automatically explore and evaluate various machine learning algorithms for customer churn prediction.
- Experimented with different algorithms, including XGBoost (a gradient boosting algorithm) and RNN (Recurrent Neural Network) to find the most accurate model.

5. Model Training and Evaluation:
- Train the selected models on the preprocessed data.
- Evaluated model performance using appropriate metrics such as accuracy, precision, recall, and F1-score to ensure the model's effectiveness in predicting customer churn.

6. Scalable REST API Development with Spring Boot:
- Created a scalable REST API using Spring Boot, a Java-based framework, to expose the trained model's predictions as an endpoint.

- This API will allow external systems or applications to make real-time predictions by sending customer data.

7. API Testing and Deployment:
   - Test the API to ensure it functions correctly and provides accurate predictions.
   - Deploy the API to a server or cloud platform to make it accessible to users.

8. Integration and Accessibility:
   - Integrate the deployed REST API with a user-friendly React.js interface, making it accessible and intuitive for companies to effectively monitor and manage customer churn, enabling them to take proactive measures to retain valuable customers and enhance their services. This user interface will provide a convenient and visual way for businesses to interact with the predictive model, empowering them to make informed decisions and implement customer retention strategies with ease.

9. Monitoring and Maintenance:
   - Implemented monitoring and error handling to ensure the API operates smoothly.

This methodology aims to build a reliable and scalable REST API for predicting customer churn using advanced machine learning techniques, making it accessible for real-world applications and decision-making in the telecom industry.

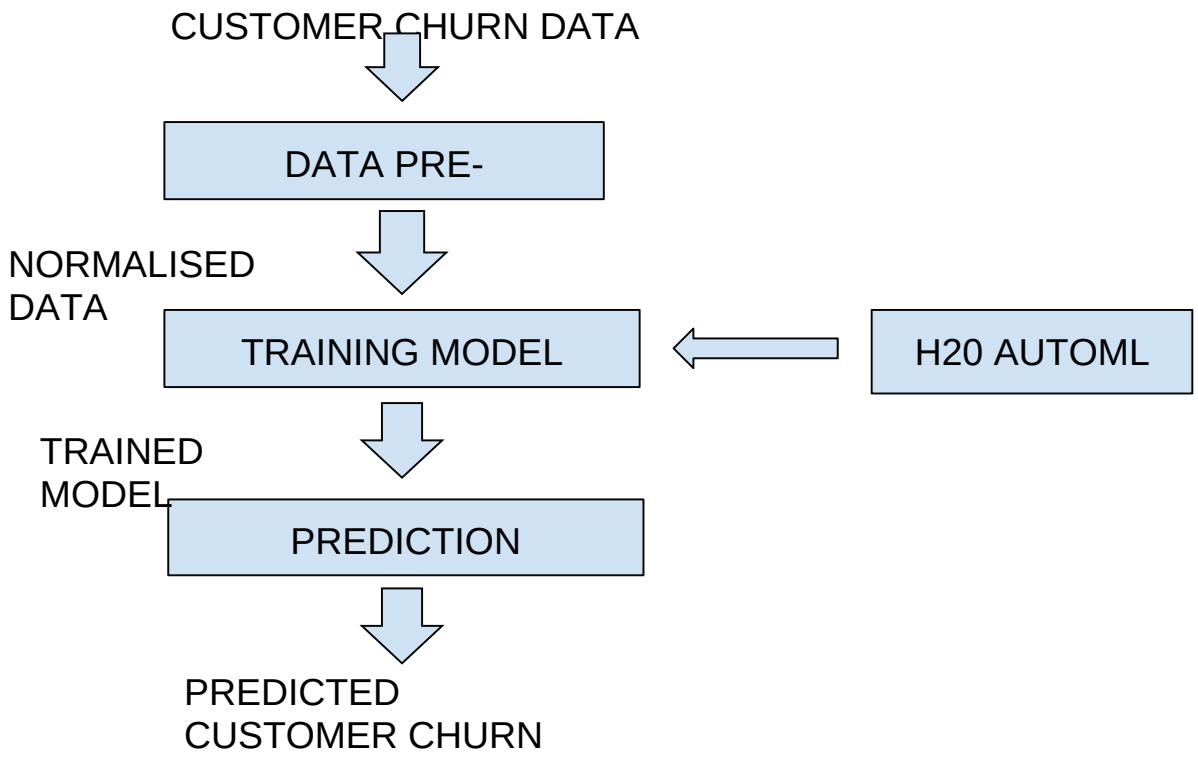

Fig 1: System Architecture

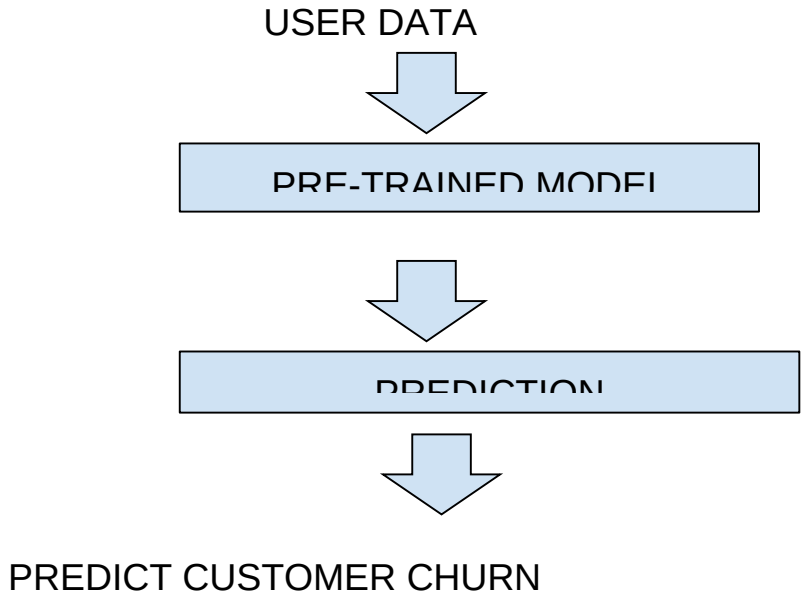

Fig 2: Workflow of Application

V. CONCLUSION

In conclusion, when it comes to predicting customer churn, XGBoost and Recurrent Neural Networks (RNNs) both offer valuable insights. XGBoost excels with its exceptional accuracy, making it a strong choice for traditional machine learning applications. On the other hand, RNNs can capture more complex patterns in the data but may require a larger dataset and more computational resources to achieve similar accuracy. The decision between XGBoost and RNNs should consider the available data, computational capacity, and the trade-off between model complexity and performance. Ultimately, the choice depends on the specific needs of the customer churn prediction task.

REFERENCES

[1]. Backerholm, Stina, and Malin Börjesjö. "Multi-Class Classification for Predicting Customer Satisfaction: Application of machine learning methods to predict customer satisfaction at IKEA." (2023).
[2]. Nataliya, Matveeva, and Mokhov Kurulo. "Using Spring Framework to develop web applications." System technologies 1, no. 144 (2023): 44-50.
[3]. Musib, Somnath. Spring Boot in Practice. Simon and Schuster, 2022.
[4]. Walls, Craig. Spring in action. Simon and Schuster, 2022.
[5]. Jarray, Noureddine, Ali Ben Abbes, Manel Rhif, Hanen Dhaou, Mohamed Ouessar, and Imed Riadh Farah. "SMETool: A web-based tool for soil moisture estimation based on Eo-Learn framework and Machine Learning methods." Environmental Modelling & Software 157 (2022): 105505.