Customer Churn Prediction using AWS Kinesis and SageMaker

Review of Literature

Project Description Recap

We are aiming to create a real-time customer churn prediction system. To do this, we'll develop a machine learning model on an (emulated) real-time dataset using tools available on AWS.

Review of Literature Description

We will review tools available on AWS under topics such as data management, data profiling & visualization, and machine learning. We'll find and select which ones fit the best for our project. In this review we first describe each tool, then give its pros & cons. We also discuss its fitness to the project, and come to a conclusion on which ones we prefer to use in the project.

Data Management

Data Management entails many aspects of handling, maintaining, and processing data in a cloud service, such as uploading the data, storing it, and having it available when it's needed. In an environment where companies deal with terabytes of data, it's essential to pick a data management system that fits the needs of the company, or the specific project. Two prominent stream data management systems are Apache Kafka, and AWS Kinesis, which we discuss below.

Kafka for Data Management

Kafka is a widely used stream (real-time) data management system that can capture and process data from databases, sensors, and cloud services. Kafka stores these streams for both real-time processing and also later retrieval, allowing processes to work on both current data, and also historic data. [1] It's designed to withstand huge amounts of stream data, and create seamless transfer between the publisher - subscriber systems.

With respect to Kafka's advantages, Kafka is a well-performing and well-tested platform that many companies prefer to do data analytics with. It provides good scaling measures, and is able to handle large amounts of stream data. It is also good at data loss prevention and fault tolerance, which is essential for many cloud services. On the other hand, Kafka can become relatively complex to implement and manage clusters for new users.

AWS Kinesis for Data Management

AWS Kinesis is a data stream processing system that became popular relatively recently. Similar to Kafka, it works on capturing, handling, maintaining, and processing huge amounts of real-time data. In addition, due to the fact that it's within the AWS structure, it is also well-connected to other AWS technologies as well, which allow users to create pipelines between data streams to storage to processes. [2] AWS Kinesis offers managed services such which allow users to manage their data without implementing an infrastructure, providing an user-friendly experience.

With respect to AWS Kinesis's advantages, AWS Kinesis is a highly managed service which reduces operational costs for creating pipelines and infrastructure. It is highly scalable, and durable. It's part of the AWS environment, which enables creating seamless end-to-end data processing through efficient transfer of data between the systems. On the other hand, AWS Kinesis can cause companies to vendor-lock on their data management system choice, and limit seeking other opportunities outside AWS.

Using AWS Kinesis for Data Management provides a scalable, user-friendly, and well-connected platform that aligns with our project's requirements for real-time processing, and integrates well with the other tools we are using, such as AWS SageMaker. Additionally, although using AWS Kinesis will push us to use more AWS technologies rather than non-AWS platforms, we are not worried about vendor-locking for now.

Data Profiling

Data profiling is the act of looking over, evaluating, assessing, and condensing data sets in order to learn more about the quality of the data. A measurement of the state of data based on elements including timeliness, correctness, consistency, completeness, and accessibility is called data quality. In order to comprehend the structure, substance, and relationships of the source data, data profiling also entails reviewing the source data. The company has two high-level benefits from this assessment process: first, it gives a high-level overview of the caliber of its data sets; second, it assists in the identification of possible data projects.

Hive for Data Profiling

Apache Hive is a widely used data analytic extension to the Hadoop MapReduce framework. A convenient alternative to the MapReduce program is SQL-like language. Simple HIVE queries are able to achieve better performance than MapReduce programs that are hand-coded. For complex analytical hand-coded programs and hive queries there could be a possible gap in performance. Many significant technical advancements were made by Hive in indexing, storage format, SQL to MapReduce translator and execution engine to improvise on performance. In-depth knowledge might be required to fully use these advancements.

Hive supports a high level programming language called Hive Query Language (HiveQL). It closely resembles Structured Query Language (SQL). Queries written in HiveQL will be analyzed and converted into one or a few Hadoop MapReduce jobs. These jobs are then submitted to the underlying Hadoop cluster to run.

- **SQL-Like Interface**: Hive provides a SQL-like interface (HiveQL), making it easy for users familiar with SQL to query and analyze data.
- **Integration with Hadoop Ecosystem:** Hive is part of the Hadoop ecosystem, making it well-integrated with other Hadoop components like HDFS and Hadoop MapReduce.
- **Scalability:** Hive is designed for scalability and can handle large-scale datasets, making it suitable for big data scenarios.
- **Data Warehousing:** It is often used for data warehousing tasks, and if data is already stored in HDFS, using Hive can be a natural choice.

PySpark for Data Profiling

PySpark is the Python API for Apache Spark, an open-source, distributed computing system used for big data processing and analytics. Apache Spark is designed for speed and ease of use, providing an interface for programming entire clusters with implicit data parallelism and fault tolerance. Apache Spark is one of the most prominent and highly valued big data frameworks. It was developed by people from the University of California and written in Scala. The performance of Apache Spark is fast because it has in-memory processing. It does real-time data processing as well as batch processing with a huge amount of data and requires a lot of memory, but it can deal with standard speed and amount of disk.

- **Programming Flexibility:** PySpark provides a more flexible and expressive programming interface using Python, enabling complex data profiling tasks.
- **Rich Ecosystem:** PySpark is part of the Apache Spark ecosystem, which includes libraries for machine learning (Spark MLlib), streaming (Spark Streaming), and graph processing (GraphX).
- **Performance Optimization:** Spark's in-memory processing can lead to better performance, especially for iterative algorithms and interactive data analysis.
- **Real-Time and Batch Processing:** PySpark supports both batch and real-time processing, making it versatile for a range of use cases, including real-time churn prediction projects.

Using PySpark for data profiling provides a powerful and versatile platform that aligns with our project's requirements for real-time processing and integrates well with the broader Spark ecosystem. It offers the flexibility and scalability needed for a proof of concept in the context of customer churn prediction.

Visualization

- **Using Python Visualization Libraries:** With the collected data, use Python visualization libraries like Matplotlib, Seaborn, or Plotly to create charts, graphs, and visualizations. These libraries provide a wide range of options for static and interactive visualizations.
- **Creating Dashboards:** To create interactive dashboards, tools like Plotly Dash or Bokeh can be integrated into your Python workflow. These tools allow users to create web-based dashboards that update in real-time.

Machine Learning

Machine Learning is a subset of Artificial Intelligence, that focuses on mathematical algorithms and the use of data to make predictions, classifications, or image / language generation based on the training dataset provided to the models. In recent years with the ease of accessibility to big data and higher computing power, the machine learning models have started to rely on huge amounts of data to achieve very high accuracies, and creativity. With the increase of demand in huge datasets, the demand for infrastructures that can maintain big data has increased as well.

AWS SageMaker for Machine Learning

AWS SageMaker is a fully managed Machine Learning service that provides the tools for building, training, and testing models at the scale of large amounts of data. AWS SageMaker allows users to work on the machine learning models end-to-end, and provides corresponding computation power required for training with huge datasets. It also provides built-in algorithms for ML tasks, and allows custom algorithms as well. It’s highly compatible with the AWS services, enabling seamless data transfer between S3 buckets, and other data management services. [6]

Using AWS SageMaker for Machine Learning provides a well-maintained experience and deeply connected services while working on the Machine Learning models. Since we are basing the data management, profiling and analysis on AWS services, it is natural to make use of AWS SageMaker for the model building, training and testing on Data Stream.

References

[1] https://kafka.apache.org/intro

[2] https://docs.aws.amazon.com/streams/latest/dev/introduction.html

[3] Dai, WEI & Wardlaw, Isaac & Cui, Yu & Mehdi, Kashif & Li, Yanyan & Long, Jun. (2016). Data Profiling Technology of Data Governance Regarding Big Data: Review and Rethinking. 10.1007/978-3-319-32467-8_39.

[4] D. Kamath, P. Srinivas, A. Gopal, B. V. Lanchana and V. Suma, "A profiling tool for apache hive run-time query," 2017 International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2017, pp. 502-507, doi: 10.1109/ICCMC.2017.8282740.

[5] Y. K. Gupta and S. Kumari, "A Study of Big Data Analytics using Apache Spark with Python and Scala," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 2020, pp. 471-478, doi: 10.1109/ICISS49785.2020.9315863.

[6] https://docs.aws.amazon.com/sagemaker/latest/dg/whatis.html

Group Members

- Avinash Sankar - A20525471
- Taufeeq Ahmed Mohammed - A20512082
- Ali Guzelyel - A20454373
- Hariprasaath Velampalayam Veerakkumaar - A20528175