ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

End-to-End Data Pipeline and Predictive Modelling for Insurance Analytics

* D Sunil Kumar [1], Sidha Meghana [2], Vittoli Sruthi [3], K Koteswara Rao[4]

[*1] Assistant Professor, KITS Akshar, Guntur, Andhra Pradesh-522019

[2,3]UG Scholar, St. Martin's Engineering College, Secunderabad, Telangana – 500100

[4] Assistant Professor, St. Martin's Engineering College, Secunderabad, Telangana – 500100

*Corresponding Author

Email: sunil1dasari@gmail.com

Abstract

Insurance telematics is an emerging and exciting field. It combines the advancements in GPS tracking, computational analytics, data processing, and machine learning into a useful tool to help insurance companies make the best product for their consumers. This is why National Indemnity looked to implement a telematics portion to their business processes of underwriting insurance policies and sponsored a School of Computing Senior Design project. In this report, we will first review existing solutions that been used to solve problems and subproblems like that we are given in this project. We then propose designs for the data pipeline and machine learning model that will be optimal in providing predictions on the risk level of drivers. National Indemnity will be able to use this project to leverage predictions to optimize insurance rates to more accurately account for risk among the insured.

Keywords: Computer Science, Machine Learning, Insurance, Telematics, Data Processing, Data Analytics

I. Introduction

In the last twenty years there has been a fast development in the technology being used for all types of businesses. This had led to ever-increasing competition in the market to stay up to date with industry standards. The areas that have particularly revolutionized by technology include data gathering, data processing, and data analysis. Recently, machine learning had emerged as an important and popular data analysis method. A broad goal of machine learning is to algorithmically discover hidden trends in datasets. The improvements in machine learning models and computational efficiency saw machine learning being used in a variety of contexts. Machine learning began to be applied to the insurance industry with increasing frequency, including in this project for National Indemnity Company. But for machine learning to be best applied, it must have quality data. For this data, it needed to be gained through a source, usually through internal resources and with third party APIs. The data in this form must then be taken and put into some sort of data storage, so it can be used later. The data is then processed in some manner to ready it for the machine learning model which needs uniform data. The model would need to be trained and evaluated extensively to ensure that the model is accurate. The goal is to have the model gain insight from the data and make it easier for the internal insurance writers to write higher quality insurance.

II. Literature Survey

Topics	Subtopics	Description	Key Papers
Data Pipelines for Machine Learning	Data Pipeline Concepts	Overview of batch and real-time	*Designing Data-Intensive*

		processing, ETL in data pipelines.	*Applications"* by Martin Kleppmann, Research on Spark, Kafka, and Hadoop.
Data Engineering and Feature Engineering for Insurance	Insurance Data Challenges	Discusses high-dimensional data, class imbalance, time-series and transactional data handling in insurance.	Articles on handling class imbalance in financial datasets, insurance-specific data engineering challenges.
Machine Learning Models in Insurance	Traditional and Advanced Models	Overview of models used in insurance, including logistic regression, decision trees, and ensemble methods.	Comparative studies on ML models in insurance, papers on deep learning applications for underwriting and claims.
Model Evaluation and Performance Metrics	Evaluation Techniques	Common metrics like accuracy, precision, recall, AUC-ROC, and RMSE for insurance models.	Studies on model evaluation frameworks specific to high-stakes industries like insurance.

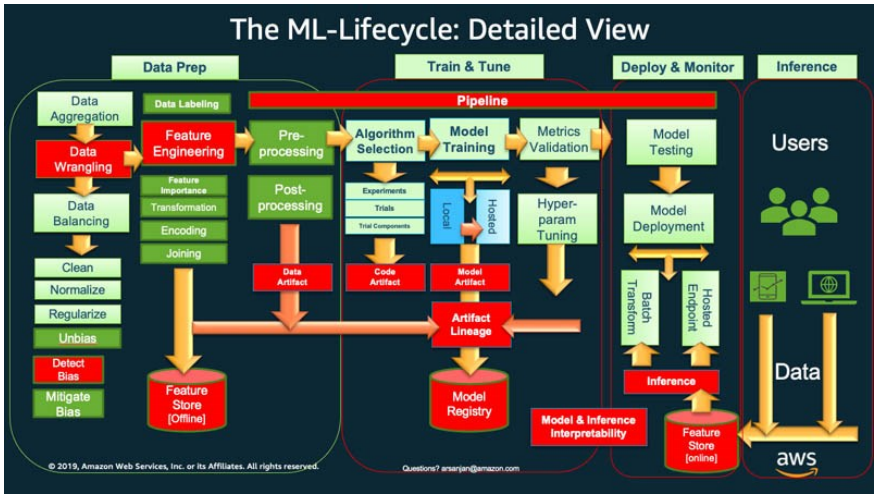
III. Proposed Methodology

In order to understand what would work best for our system, we first looked at the literature of similar projects to see how to best implement the highest quality system. We will take sources covering a variety of topics from data ingestion, data storage, data processing, and machine learning models. The majority of the sources used are studies discussing these features in relation to the insurance industry. There are also textbooks to give evidence to some of the more general features of a data processing and machine learning project. The insurance industry provides its own unique issues and circumstances for learning problems that are handled in a range of complex and unique solutions.

In machine learning, the data is the most crucial part. The data is generated by a variety of sources and must be ingested into a cohesive system where it can be further processed 1 and used later to train the model. With insurance data, ingestion can be done in a variety of methods. These methods are determined by what kind of data the pipeline will be handling. The data coming in is normally through an outside source. The data is then received will be put into datastorage for later use. Data will also be gathered from several different sources. For insurance, these include but are not limited to telematics providers, mobile applications, other devices connected to the internet, and internal data. For ingestion purposes, each source will likely have its own format and security methods. Security risks on both sides of the connection should be managed with passwords, API keys, and other various security measures. Each source will need to be handled individually to ensure that the data is ingested properly.

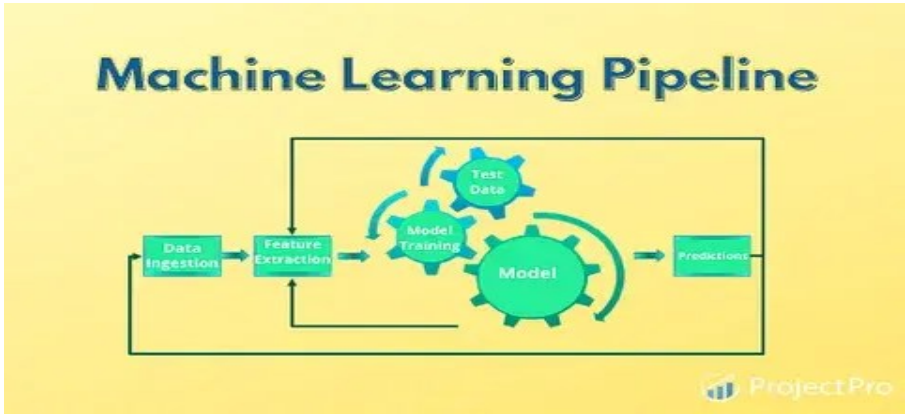
Figure 1: ML-Lifecycle

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT
ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 1, 2025



Data in large quantities is stored by various means. To have an efficient system, the data storage mechanism must be well optimized for the data and for what the data will be used. These require careful tuning to build complex storage architectures. Common structures include data warehouses and data lakes. Additionally, there are additional architecture types such as NoSQL, SQL, and cloud storage. Data warehouses are best practice when there is a large majority of structured data. This would be data that all comes in the same forms such as CSV or another rigidly structured data type. The data is typically stored in tables like formats and have strict relationships with other tables. This allows for higher speeds but lower flexibility. Data lakes are better when the data contains semi-structured or unstructured data. This wouldbe for file types that have little to no structure like JSON or text documents. Data lakes are best for these data types because they allow for more flexibility with data types. Structured data is also allowed in data lakes but would have less efficient access. Data lakes are best for handling raw and heterogeneous data [MPT20]. Each of these data models is best suited for certain types of data and the application that will be accessing them. In general NoSQL uses fewer standard queries compared to relational databases seen with SQL databases. This makes the databases less able to port to other service providers and integrate with certain data types [MK19]. Cloud storage is another potential solution for data storage. Storing data on the cloud allows for easy and scalable storage for a company's data. It also allows efficient access from multiple locations at once. One downside is potential latency as the data is being accessed through an outside network. It is much more dependent on a larger external network. Additionally, entrusting data on an outside network with another company can be a security concern in some instances [KSF+20].

Figure 2: Machine Learning Pipeline



For large data models, there is often extraneous and duplicate data. Keeping all this data in can lead to a bloated and slow model. One key step to inputting data is to reduce the size of the data. This will boost computation speed but could lead to less accurate results. The key to reducing data is to find a

method that extracts relevant data while discarding extraneous data. There are three broad categories to reduce data size. They are dimensionality reduction, numerosity reduction, and data compression. Dimensionality reduction is done to reduce the number of fields input to the model to reduce the computational burden of the model. The most common methods are Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Generalized Discriminant Analysis (GDA). PCA uses orthogonal mapping to map a large dimension dataset to a lower dimension set. It is good to be used for mapping relationships between variables. The goal of PCA is maximize variance across the dataset. The next most common method is LDA. It works by projecting a dataset onto a lower set of variables. It is quite similar to PCA but has a goal of maximizing differences between classes. This allows the ML model to more easily determine classes. In numerosity reduction, the data is represented in smaller forms to reduce the volume of the data.

IV. Implementation

This project required large amounts of data from third party providers. Agreements were made with the companies and relevant credentials were given to be used in the project. Since the data was coming from different providers, different data ingestion pipelines will be used to get the data into data storage. For data storage, see section 3.2 for more information on implementation. There are generally two categories of data ingestion implementations. The first is to stream in the data. This means that data is ingested in real time from the provider. This is done when a project is time-sensitive and a delay in input would have a significant negative impact on the project. The second category is to ingest the data in batches. This method does not ingest data as soon as it is ready, rather it processes data at set intervals. This project was better suited to ingesting data in batches. The machine learning model was not trained in real-time. It did not need the data as soon as possible. Additionally, the amount of data ingested from the providers would be more than what would be cost-effective for a streaming method. Streaming adds a much higher degree of complexity to the implementation. Batch ingestion allows for relatively standard processing times, data size, and frequency. This allowed for an optimal ingestion method for this project. For this project, the data used had already been gathered without a set ingestion scheme. It will be up to future projects to fully realize the implementation of the ingestion. Since the project did not require any complex ingestion method, the common applications for data ingestion more than sufficed. Azure Data Factory was the best for this project as it allowed for seamless integration to other Azure products that were used in the later parts of the pipeline.

For companies like National Indemnity that are not in the tech industry, it is common to use third parties for data storage as it is extremely expensive to establish their own servers and operating systems. National Indemnity has the preference of Azure and Microsoft services so that is the primary software lender the project used. For insurance data for our machine learning system, a data lake was the best option. The data consisted of text documents, tables, and third-party telematics data in JSON format. The data was also coming into our system raw and unprocessed, so our system needed to be able to handle the storage of all the stages of the data in the pipeline. A data lake was the best option for our project because data lakes can handle the input data is given, as well as handling the data in intermediary processing steps. The data lake would also be able to store the testing results without being constrained by a strict data format. Since the project did not have fully integrated ingestion, the data storage was also not fully implemented. This will also be done by future projects to ensure efficient use of data within the project. The best option for this would be using Azure Data Lake Storage as it would provide the necessary resources to build an efficient data lake that can be integrated well with the data ingestion and other systems operated by National Indemnity.

Machine learning is a broad field with a variety of different models to choose from. Each model has its own unique benefits and challenges. Every problem in the real world contains its own unique circumstances that must be carefully considered. The model must be chosen to best fit the problem.

Additionally, there is a multitude of further adaptations and hyperparameters to choose from that make the decision extremely hard to get right. The goal of this project was to determine the risk of a given driver. The dataset used in the project contained speed, direction, time, and other telematics data. After processing, the data is ready to be fed to a model. Since the project was looking to predict a data point, that narrows down the models to supervised models. Purely unsupervised models like clustering were ruled out. The most used models for similar problems are decision trees, random forests, boosting, support vector machines, and neural networks. Though a purely clustering model would not be useful for the current project, using a clustering model and then a supervised model on top would be possible. This option would first group the dataset into related groups. Then a supervised model could be trained on each of the clusters individually, providing more tailored results on each cluster. The issues with this approach would be that it could result in overfitting since each model instance would only be trained on a small subsection of the data. This method could also be computationally expensive as well as disjointed by breaking up training intounsupervised and supervised on multiple clusters. This approach was deemed to be not optimal for this project. Support Vector Machines are efficient models that can be used in a variety of applications in the insurance industry. They are typically highly accurate, especially in high dimensional spaces. The dataset used in this project had a number of dimensions but does not have such a high amount that support vector machines would be able to provide a relative advantage to other models. Support vector machines can also be expensive, especially on multiclass problems. The project will predict a risk score which would need additional complexity in a support vector machine. A support vector machine would not be the best choice for a problem like this. Neural networks are good for the project because they are typically highly accurate. The issues with them are that they are typically costly in terms of time and energy. The project does not need to train the model frequently but keeping cost low is always a high priority. Additionally, neural networks are hard to understand. They work as a black box that takes in inputs and does a number of intermediary steps before outputting a prediction. While the input and output layers are well defined and easy to understand,everything that happens in between is largely incomprehensible for a human. This means neural networks are hard to fine tune if the model is not performing up to standards. This lack of understanding also poses a problem to the business aspect since it is hard to explain to non-technical users what the model is doing.

The XGBoost model that has been created takes in the processed data from previous steps and trains on various features to be able to get the best results. It can accurately predict a driver risk score. The model can be tuned so that it focuses more on features like location, speeding, or braking that would need to be changed depending on what the user finds valuable given the context of the business and other environmental factors. The project consisted of a graphical user interface that could be used by those at National Indemnity to train the model. The interface allowed the user to choose the data that the model will train on. Additionally, the user could change the parameters to the model to finely tune the model to the specific dataset of the user. The flexibility and interpretability of XGBoost allows the interface to take advantage of this to make it so that non-technical users can adequately harness the power of the model.

V. Experimental Results

1. Pipeline Performance and Data Quality

- Data Processing Time: Measure the time taken at each stage (e.g., data ingestion, cleaning, transformation). Visualize processing times for different data volumes to show scalability.
- Data Quality Metrics: Show the impact of data cleaning steps by tracking changes in metrics like missing value counts, consistency checks, and outlier counts.

2. Model Performance Metrics

- Prediction Accuracy: Present standard metrics (e.g., accuracy, precision, recall, F1 score, AUC) based on your model's predictions on the test data.
- Loss and Convergence: Display loss curves over training epochs to demonstrate convergence.
- Feature Importance: Identify the most influential features in the model using techniques like permutation importance or SHAP (Shapley Additive Explanations) values.

3. Comparative Analysis

- Model Comparison: Compare your primary model's performance against baseline models (e.g., logistic regression, decision tree). Visualize with bar charts or ROC curves.
- Hyperparameter Tuning Results: Show how tuning parameters like learning rate, depth, and regularization improved performance, ideally with a grid search or random search heatmap.

4. Error Analysis

- Confusion Matrix: Present confusion matrices for the model's classification results to analyze misclassifications.
- Error Distribution: Plot residuals to understand error patterns and identify any consistent biases.

5. Business Impact

- Financial Gains/Loss Reduction: Calculate potential savings or revenue increase by improving the prediction accuracy.
- Customer Retention/Loss Prediction: If applicable, show how the model could help with customer retention strategies by accurately predicting at-risk customers.

VI. Conclusion

In this paper, we outlined the implementation of an effective data pipeline and machine learning model for assessing driver risk in the auto insurance industry. Through efficient data ingestion, storage, and processing stages, we ensured that raw data from a multitude of sources was able to be transformed into a usable format for training our machine learning model. We prepared a rich dataset capable of feeding into our machine learning model effectively. To do this we utilized a data lake for storage and employed processing techniques including imputation, aggregation, feature engineering, and normalization, In selecting a suitable machine learning model, we evaluated various options including clustering, support vector machines, neural networks, gradient boosting machines, decision trees, and random forests. Ultimately, we determined that gradient boosting and random forests, with XGBoost as our preferred implementation, offered the optimal combination of accuracy, efficiency, and interpretability for our project's needs. The XGBoost model developed in this study not only demonstrates accuracy in assessing driver risk but also provides a user-friendly interface for non-technical stakeholders to fine-tune parameters based on their domain expertise. This adaptability ensures that the model remains relevant and effective amid evolving business requirements and environmental factors for the company. By integrating robust data pipeline implementation with a sophisticated machine learning model, this project lays a foundation for enhancing risk assessment strategiesin the insurance industry, contributing to improved decision-making and operational efficiency within the field.

References:

[1]Shady Abdelhadi, Khaled Elbahnasy, and Mohamed Abdelsalam. A proposed model to predict auto insurance claims using machine learning techniques. Journal of Theoretical and Applied Information Technology, 98(22):3428–3437, 2020.

[2][Alp20] EthemAlpayd. Introduction to Machine Learning. MIT Press, Cambridge, MA, 2020.

[3][ARGAT20] JaberAlwidian, Sana Rahman, MaramGnaim, and Fatima Al-Taharwah. Big data ingestion and preparation tools. Modern Applied Science, 14:12, 08 2020.

[4][BAL+20] Sarah Benjelloun, Mohamed El Mehdi El Aissi, YassineLoukili, YounesLakhrissi, SafaeElhaj Ben Ali, Hiba Chougrad, and Abdessamad El Boushaki. Big data processing: Batch-based processing and stream-based processing. In 2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS), pages 1–6, 2020.

[5] [Cea09] S. Chakrabarti and et al. Data Mining: Know It All. Morgan Kaufmann, Burlington, Massachusetts, 2009. [FJS16] Kuangnan Fang, Yefei Jiang, and Malin Song. Customer profitability forecasting using big data analytics: A case study of the insurance industry. Computers Industrial Engineering, 101:554–564, 2016.

[6]Kuangnan Fang, Yefei Jiang, and Malin Song. Customer profitability forecasting using big data analytics: A case study of the insurance industry. Computers Industrial Engineering, 101:554–564, 2016.

[7] [HM21] Mohamed Hanafy and Ruixing Ming. Machine learning approaches for auto insurance big data. Risks, 9(2), 2021. [

[8]KSF+20] Christian Kaiser, Alexander Stocker, Andreas Festl, MarijaDjokic-Petrovic, EfiPapatheocharous, Anders Wallberg, Gonzalo Ezquerro, Jordi OrtigosaOrbe, Tom Szilagyi, and Michael Fellmann. A vehicle telematics service for driving style detection: Implementation and privacy challenges. In VEHITS, pages 29–36, 2020.

[9][MK19] Andreas Meier and Michael Kaufmann. SQL & NoSQL databases. Springer, 2019. [MP18] AndreeaM˘at˘acut¸˘a and C˘at˘alinaPopa. Big data analytics: Analysis of features and performance of big data ingestion tools. InformaticaEconomica, 22(2), 2018.

[10]B´alintMoln´ar, Galena Pisoni, and Adam Tarcsi. Data lakes for insurance industry: Exploring challenges and opportunities for customer behaviour analytics, risk assessment, and industry adoption. 07 2020.

[11] [PGPZ23] Thomas Poufinas, PeriklisGogas, Theophilos Papadimitriou, and EmmanouilZaganidis. Machine learning in forecasting motor insurance claims. Risks, 11(9), 2023.

[12][PNGA19] Jessica Pesantez-Narvaez, Montserrat Guillen, and Manuela Alca˜niz. Predicting motor insurance claims using telematics data—xgboost versus logistic regression. Risks, 7(2), 2019. [sci] scikit-learn Contributors. Preprocessing data. https://scikitlearn.org/stable/modules/preprocessing.html. Accessed: May 6, 2024.