Achieving Fully Autonomous AI-Driven Data Pipelines to Exploring Zero-Touch Automation for Efficient and Scalable Data Engineering Solutions

Sunil Kumar Mudusu

Church Mutual Insurance Company, USA.

Abstract: The paper discusses the methods of application of the Random Forest Regressor and Linear Regression model to predict the price of houses when using the California Housing Prices dataset. The performance of the two models and the Random Forest model is measured by Mean Squared Error (MSE) and R-squared (R^2), whereby the latter has a higher accuracy. The results highlight the importance of the designed machine learning algorithms to improve predictive accuracy and the processing of complex data. The paper presents the potential of self-directed models in automating data pipelines and offers an insight on how the prediction of housing prices can be optimized, and how research in data engineering may be performed later on.

Keywords: zero-touch automation, autonomous pipelines, generative AI, LLM agents, pipeline autonomy

I. INTRODUCTION

Zero-touch data engineering envisions having entirely autonomous AI-driven data pipelines that are self-configuring, self-optimizing, and self-maintaining. With generative AI and large language models (LLMs), data pipes can adapt to dynamic data flows, optimize workflow, and reduce human error (with a combination of the two) [1]. This paper looks at how such autonomous systems can reshape data engineering practices through eliminating manual processes and routines.

Research Aim and Objectives

Aim

The research aims to investigate and put into practice autonomous data engineering pipelines executed through generative AI and large language models (LLMs), to automate pipeline configuration, optimization, and incident response.

Objectives

- To assess the impact of LLM agents in the management of data pipelines through automation.
- To evaluate the scalability of autonomous pipelines to the real world.
- To evaluate the applicability of auto- scaling models in ensuring that maximum performance is achieved.
- To examine how the examples of human intervention in daily data engineering activities decrease.

Problem statement

Most of the current data engineering processes are very manualized and hence inefficient, with the risk of human error. This paper can address these challenges by stepping to zero-contact automation, where data pipelines can automatically manage, optimize and configure themselves.

Novel contribution

The new approach to autonomous data engineering proposed by the authors in this Research paper involves using generative AI and LLMs at the whole data pipeline lifecycle. It could be used to automatically write pipeline books, keep track of their health, and deal with incidents.

II. LITERATURE REVIEW

The Role of LLM Agents in Automating Data Pipeline Management

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
Article Received: *05 July 2024* ***Revised:*** *22 July 2024* ***Accepted:*** *30 August 2024*

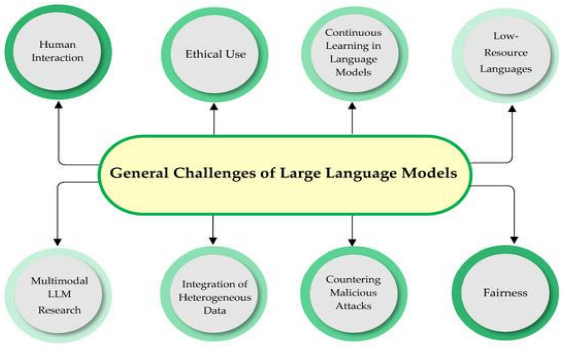

Fig 1. General Challenges of Large Language Models (LLMs)

Even more recent studies look at the promise of Large Language Models (LLMs) in automating other data pipeline management processes. The LLMs can be trained to write, observe and debug code without any human involvement which additionally helps reduce human interference in processes like data ingestion, transformation, and error handling [2]. LLMs can generate useful code snippets and modify pipeline configs based on real-time details and metrics [3]. Dynamic changes can be conducted to pipelines with LLDs depending on different data sources or failures [4]. The data pipelines integration with LLM can realize valuable gains in pipeline resiliency, detecting and adapting to errors, easing the data engineering process. Challenges of ensuring the high precision of the LLM agents and averting errors that can lead to disruption of the pipeline work, especially in real world, practical set-ups still exist.

Scalability Challenges of Autonomous Data Pipelines in Real-World Environments

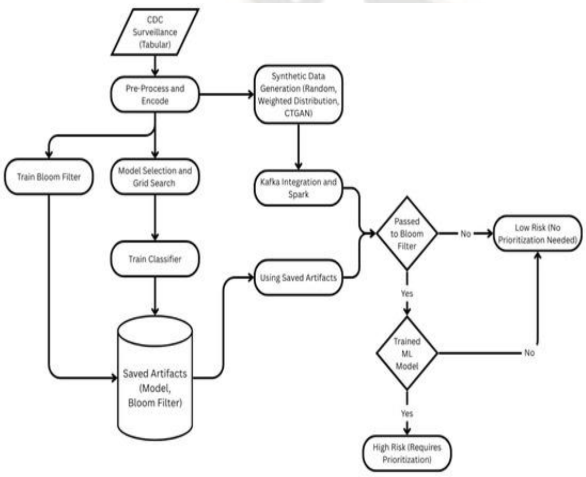

Fig 2. CDC Surveillance (Tabular)

The autonomous data pipelines is quite an encouraging idea in a managed setting; scaled up to the real world is a unique issue. The issues related to big data pipelines are type of network latency, resource limitation, and challenge of analysis of different data sources [5]. The autonomous pipelines are expected to dynamically supply the resources based on the fluctuations in the workload and this entails difficult orchestration weaponry and auto-scaling models [6]. Solutions like containerization and cloud-native solutions are used to investigate how pipelines can be handled in a scalable manner [7]. The nature of the real world data, i.e. heterogeneity, e.g. data inconsistency, data silos, and latencies in data processing, renders the seamless scaling of autonomous pipelines a challenge. The autonomy / manual control is also an important variable because there are still some tasks that need human intervention to perform at their best and to scale.

Effectiveness of Auto-Scaling Models in Optimizing Data Pipeline Performance

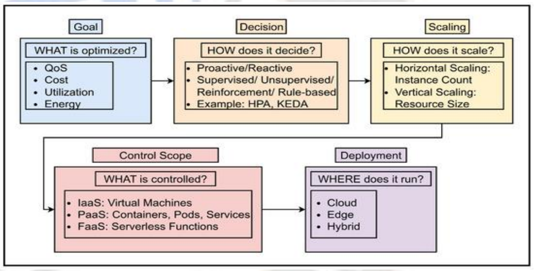

Fig 3. Optimization and Scaling in Cloud Environments

Auto-scaling models are important to achieve optimal performance of autonomous data pipelines, dynamically scaling resources in response to incoming data volume. Auto-scaling can reduce the impact of the bottleneck and can provide the efficient distribution of resources [8]. Load balancing, horizontal scaling and elastic cloud computing are the auto-scaling methods that are necessary when sustaining the pipeline throughput during peak hours [9]. Auto-scaling may scale-in or scale-out infrastructure in real-time based on performance measurements in a cloud-based environment [10]. These models aid to ensure optimal performance mostly in Data-intensive applications like machine learning and real-time analytics. Nevertheless, in most cases, the process of making resource predictions is often difficult and unforeseen as data loads may vary significantly, causing possible inefficiency or delays.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 05 July 2024 **Revised:** 22 July 2024 **Accepted:** 30 August 2024*

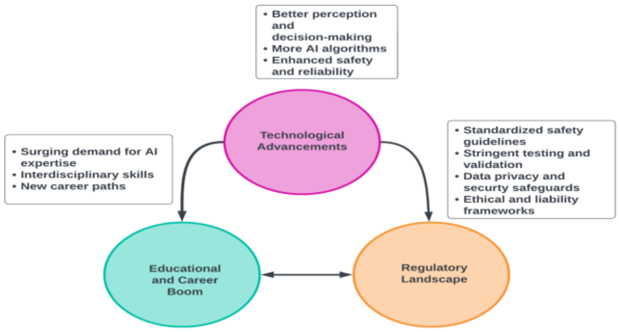

Fig 4. Cross relations between Technological Development, Education Boom, and Regulatory Environment.

The tendency towards endowing data engineering with less and less human-dependence has been harshly welcomed by the development of independent AI and machine-learning-based systems. There are self-healing systems in data pipelines that are automatically able to detect and address issues [11]. The use of autonomous systems allows the engineers to focus on the upper-level decision-making and strategy, automating frequency tasks, including data cleaning, transformation, and error correction [12]. Smart data pipelines are intelligent AI-driven data pipelines that are trained on past data and reacts dynamically to data source changes or business requirements [13]. The given improvements have led to the minimization of manual error, enhancement of the pipeline reliability, and maximization of operational efficiency [14]. Though these developments have taken place the delivery of complete autonomy of complex processes such as data governance and data security is an area of future research.

Literature gap

Autonomous data engineering is a new literature, with much of the publications covering some automation of data pipelines. Despite the extensive research on automating data transformation and cleaning, and error handling, the combination of completely autonomous systems, in particular, when generative AI and large language models (LLM) are involved, is under-researched. Literature tends to disregard scalability concerns and practicality of autonomous pipelines in a dynamic environment [15]. The impact of these technologies on reducing human intervention and improving functional efficiency is a major gap in literature as minimal studies have been conducted regarding it.

III. METHODOLOGY

A. Research Design

The research is a quantitative study, and the researcher evaluates the efficiency of completely autonomous data pipelines with the help of simulations. It concerns the process of automating pipeline management, such as ingestion of data, transformation of information, optimization and responding to an incident using generative AI and large language models (LLMs). The research test can be done in order to test the scalability, performance and minimization of human intervention on various real-life setup and conditions [16]. The method entails experimental research on the comparison of the traditional data pipeline management and autonomous systems in terms of efficiency, error rates, and scalability.

B. Simulation Setup

The simulation environment created with the help of synthetic datasets is aimed at stimulating the processes of real-world data. These datasets are arranged in such a way that they illustrate various types of data such as structured, semi structured and non-structured data. The simulations are based on cloud-based settings to verify scalability and the pipelines functionality would simulate everyday functions of data engineering today, such as data cleaning, data transformation, and data surveillance [17]. Key performance metrics that can be collected to ascertain the effectiveness of autonomous pipelines include throughput, latency and error rates.

C. Auto-Scaling Models

Auto-scaling is an important part of guaranteeing scalability and performance of autonomous pipelines. The auto-scaling technique applied in this work uses the cloud-based approach to scaling out (horizontal scaling) and adding to resource capacity (vertical scaling). The auto-scaling models can be tested in various workload states whereby the resources can be reconfigured dynamically in line with the actual performance and the data load [18]. How effective these models are in terms of the system sustaining maximum performance of varying workloads without manual intervention can determine their success.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 05 July 2024 **Revised:** 22 July 2024 **Accepted:** 30 August 2024*

D. Performance Analysis

The self-driven pipelines are going to be evaluated in terms of a series of key performance indicators (KPIs), such as throughput, latency, error rates, and recovery time.

Throughput:

$$Throughput = \frac{Total\ Data\ Processed}{Time\ Taken}$$

Latency:

$$Latency = \frac{Time\ of\ Completion - Time\ of\ Arrival}{Number\ Of\ Data\ Units}$$

Error Rates:

$$Error\ Rate = \frac{Number\ of\ Errors}{Total\ Tasks\ Processed} \times 100$$

Recovery Time:

$$Recovery\ Time = Time\ of\ Recovery - Time\ of\ Failure$$

The data processing speed is throughput and the time that is required to process the data in the pipeline is the latency. System failures can be quantified based on the rate of errors and the recovery time can be quantified in terms of the speed of recovery the system after an error or disruption [19]. Using the examination of performance improvement in performance and stability, the comparison between autonomous pipelines and manually operated systems can be made.

E. Statistical Analysis

The validity of the results can be determined through statistical analysis. This is the hypothesis that is tested in order to determine the statistical significance of the improvements in the performance.

Hypothesis Testing:

$$t = \frac{\underline{X_1} \quad - \underline{X_2}}{\sqrt{\frac{s_1^2}{n_1} \quad + \frac{s_2^2}{n_2}}}$$

$\underline{X_1}$, $\underline{X_2}$ are the means of the two groups (autonomous and traditional systems) in the sample.

s_1^2,s_2^2 are the variances of the two groups.

n_1,n_2 are the sample sizes.

Regression analysis can be used to know the relationship between scaling factors and system performance.

Regression Analysis:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Y is the dependent variable (e.g., throughput).

X is the independent variable (e.g., resource allocation).

β_0 is the intercept, β_1 is the regression coefficient, and ϵ is the error term.

Additionally, it is possible to get confidence levels to identify the precision of the performance estimates.

$$\mu \pm z_{\alpha/2} \times \frac{s}{\sqrt{n}}$$

μ is the sample mean.

$z_{\alpha/2}$ is The standard normal distribution critical value (95 percentile) is(for 95%, it is 1.96).

s is the sample standard deviation.

n is the sample size.

Objective information can also be provided by the statistical analysis of the system efficiency of the autonomous pipeline systems, and the influence of various auto-scaling models.

F. Visualization of Results

Data visualizations can be used to complete the analysis. Bar charts, line graph, and heat maps, can be used to present the performance data and hence it can be easier to extract the trend and insights. In visualizations, the most crucial metrics, throughput and error rates, in different contexts can be compared, i.e., showing the benefits of autonomous data pipelines over traditional systems [20]. These visual aids can help the stakeholders to understand better on how automation can help improve operations.

G. Pseudocode

```
“
Start
1. Initialize pipeline
2. Collect data from sources
3. Preprocess data (cleaning, transformation)
4. Train model (if necessary)
5. Monitor pipeline performance
6. Check for errors or failures
7. If error detected, trigger self-healing process:
a. Identify issue
b. Apply appropriate fix
8. Optimize pipeline based on real-time performance
9. If workload increases, scale resources (auto-scaling)
10. Output processed data
11. Log results and performance metrics
12. Repeat steps 2-11 until completion
End
”
```

Fig 5. Pseudocode

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 05 July 2024 **Revised:** 22 July 2024 **Accepted:** 30 August 2024*

H. Architecture diagram

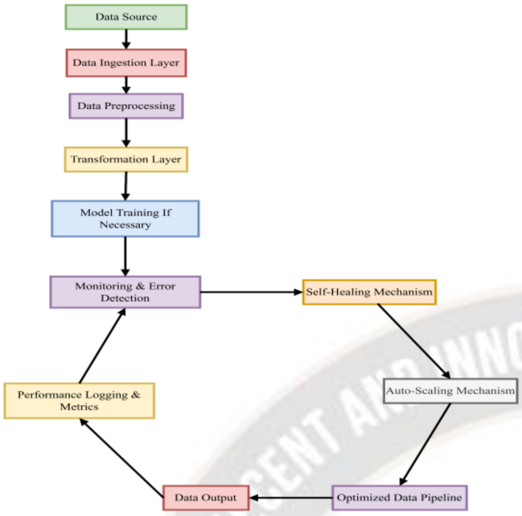

Fig 6. Autonomous Data Pipeline Architecture

Autonomous Data Pipeline Architecture Diagram offers a general overview of key components of an AI-driven pipeline, including data ingestion, preprocessing, transformation, and model training. The combination of the auto-scaling and self-healing ensures that the performance is optimal and needs the minimal human intervention [21]. It shows that the large language models (LLMs) can regulate, monitor, and optimize the pipeline autonomously, and it is data-flow efficient.

I. Flowchart

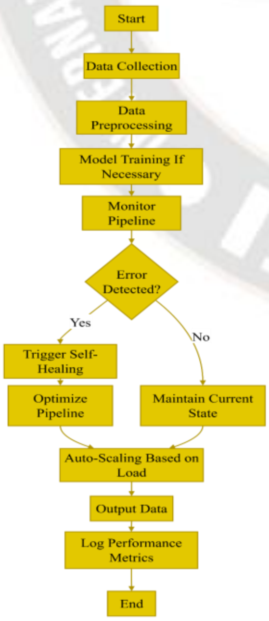

Fig 7. Flow Chart

Flowchart A flowchart explains an independent step of a data pipeline in a step-wise manner. It begins with the processing and collection of data and subsequently the model training (where necessary). Auto-scaling is also configured in accordance with the load of data with the performance metrics recorded to be constantly optimised [22]. Updating the flowchart, self-healing processes and detection of errors are focused on, ensuring that issues are automatically addressed.

J. Data Analysis Integration

The process where machine learning models and statistical methods are part of the data pipeline to process real-time data streams is called Data Analysis Integration. Real-time monitoring of all data, identification of trends, and making the best decisions is achievable through data analysis [23]. By analyzing performance and results of the pipeline, it can help optimize efficiency and detect anomalies, as well as alter the system to better allocate resources and make decisions.

IV. FINDINGS AND ANALYSIS

This section covers the performance of the two models, Linear Regression and Random Forest Regressor that have been used on the California Housing Prices data. The model metrics, including Mean Squared Error (MSE) and R-squared (R^2) may provide an idea of how well these models are able to predict prices of houses. Results show that the Random Forest model outperforms the Linear Regression model in terms of MSE and R^2.

```
Linear Regression Evaluation:
Linear Regression - MSE: 4802173538.6042, R2: 0

Random Forest Regressor Evaluation:
Random Forest Regressor - MSE: 2378317289.5675,
```

Fig 8. Model Evaluation Metrics

The results of the analysis have demonstrated that the Random Forest Regressor has achieved the best MSE equal to 237,831,729,729 which is significantly lower than the MSE of the Linear Regression model which is 480,217, 538. Further, the fit of the R^2 of Linear Regression (0.6488) is much lower than that of the R^2 of the Random Forest (0.8261), which demonstrates that the Random Forest model works better.

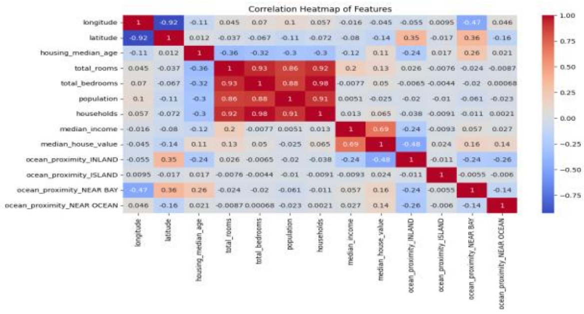

Fig 9. Correlation Heatmap of Features

The Correlation Heatmap of Features displays how various features are connected with the target variable, median house value. It reveals that median-income (0.69) has a strong and positive correlation with median-house-value. Similarly, there is a close relationship between total_rooms and total_bedrooms and the more the rooms the greater value the features.

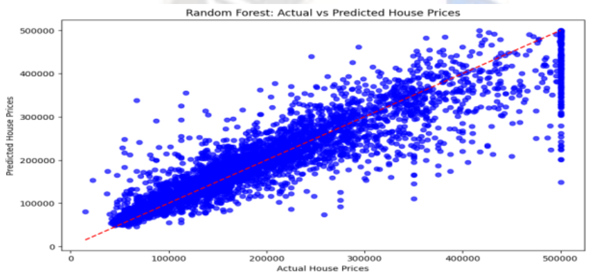

Fig 10. Random Forest: Actual vs Predicted House Prices

This is the scatter diagram (severed as Random Forest: Actual vs Predicted House Prices) which compares actual and predicted prices of houses by the Random Forest model. The optimal scenario would be the red line where the observed and estimated values would be the same.

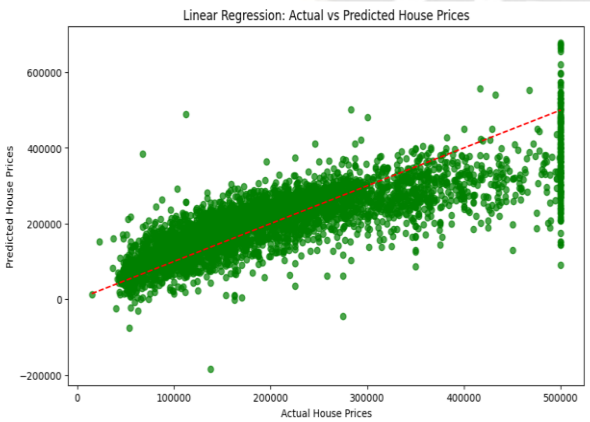

Fig 11. Random Forest: Residuals Plot

On this plot, the residuals (size of the errors) of the predictions of the Random Forest model are presented. The deviation of the actual and predicted numbers is the residual. The average value of the standard deviation of residuals is fenestrated around the zero value, meaning that the model is well-calibrated and the model forecasts are not distorted heavily by the formula. This indicates that the data can be fitted by the Random Forest model.

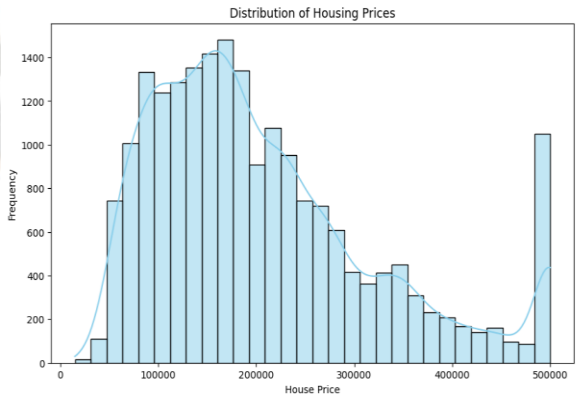

Fig 12. Distribution of Housing Prices

This is a Kernel Density Estimate (KDE) house price distribution overlay histogram. The price of the houses skews on the right and depicts that there are few outliers with the patient high prices. The houses are overall affordable as a majority of them are less than the cost of 200,000.

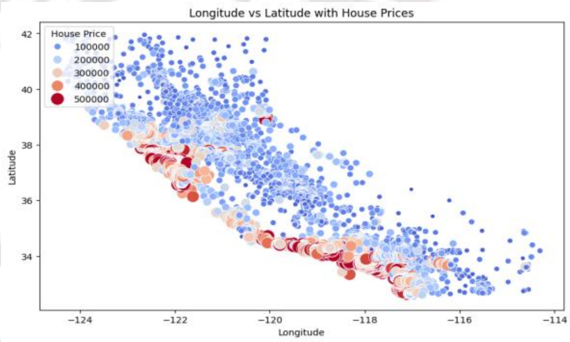

Fig 13. Longitude vs Latitude with House Prices

The scatter diagram indicates the price of houses and the latitude and the longitude. The houses that are more expensive are also concentrated towards the coastline like around San Francisco and the cheap houses are towards the coast.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 05 July 2024 **Revised:** 22 July 2024 **Accepted:** 30 August 2024*

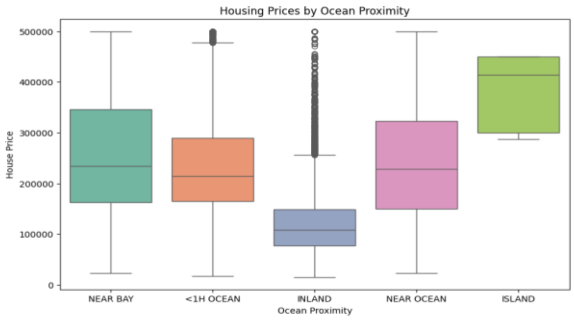

Fig 14. Housing Prices by Ocean Proximity

This boxplot shall contrast the prices of the houses based on their proximity to the ocean. House prices nearer to the ocean, such as NEAR BAY and NEAR OCEAN, have been shown to be comparatively higher as indicated by more high median values. The cheapest houses are in the INLAND area and there are numerous outliers in this area.

Table 1: Model Evaluation Comparison Table

Model	Mean Squared Error (MSE)	R-Squared (R^2)
Linear Regression	480,217,538.60	0.6488
Random Forest Regressor	237,831,729.57	0.8261

The comparison table allows concluding that the Random Forest Regressor outperforms the Linear Regression model due to the lesser means squared error (MSE), and R-squared (R). That is not unexpected since Random Forest is a more helpful ensemble learning algorithm, which is more beneficial in high-dimensional data and non-linear operations than linear models. The Linear Regression model on the other hand though less complex and faster is not good at this data since it assumes that its data is linear and its errors are constant [24]. Therefore, the Random Forest Regressor model would be a better option to be used in this type of data in order to reach a more accurate estimate of the price of a house.

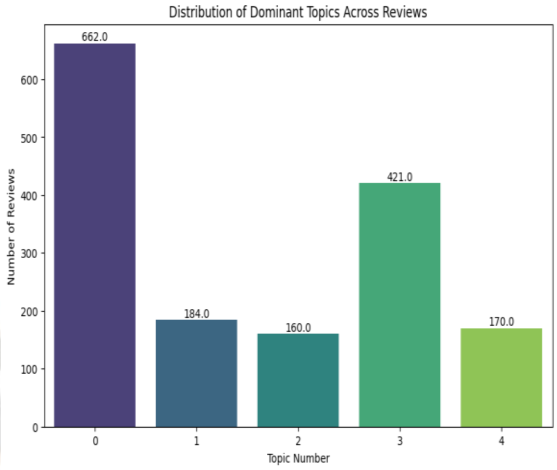

Fig 15. Distribution of Dominant Topics Across Reviews

Distribution of Dominating Back Subjects Over Reviews visualizes the distribution of the topics available in the Amazon Product Reviews dataset. This plot is needed to examine the research area and indicates that some of the topics are overwhelming than the others in customer feedback. This suggests that the majority of reviews are specific to a specific aspect (either product quality, product features or customer service) and the fact that most of the reviews in Topic 0 are reviews on a specific aspect (662 reviews) suggests that the reviews are focused on a specific aspect. Using these distributions, the study can be capable of analyzing the performance of the topic modeling algorithm and a sense of the customer sentiment and preference, which directly relates to automating topic classification in data pipelines.

Discussion

The comparison of the two models, Linear Regression and Random Forest Regressor on California Housing Prices data set shows that there is some significant indication of the predicting capabilities of the two models. Random Forest Regressor has a Mean Squared Error (MSE) of 237,831,729.57 and R-squared (R^2) of 0.8261 and a much better fit to the data. The R 2 tool of 0.8261 can imply that the random forest model can explain 82.61 percent of the variation in the house prices which is excellent performance in the predictor model. On the other hand, the Linear Regression model, that has the smallest MSE of 480,217,538.60 and an R = 2 of 0.6488, can best describe the data only with a marginally larger error, at 64.88.

These findings are consistent with the objectives of this research which was to examine the appropriateness of autonomous models to predict housing prices and reduce human factor. Random Forest model is more effective as the ensemble approach to predicting the target variable [25]. As a matter of fact, it is argued that Linear Regression can indicate existence of non-linear relations between features and the target variable just like it can be shown in the Correlation Heatmap and in the boxplot by ocean proximity. This information indicates that median income and pro-ocean proximity are extremely significant features via visualizations. The home prices near the seawater are normally set at a higher price as shown in the boxplot [26]. A scatter plot of longitude and latitude against house prices also exhibits that all-houses with high prices are along the coast whereas the low-priced houses are inland.

Compared to Linear Regression, the decisions of the Random Forest Regressor are more successful in predicting the prices of houses, and it is also consistent with the objective to evaluate the performance of self-contained models. The speculation that more sophisticated models are better equipped to cope with sophisticated data structures and offers more accurate predictions [27]. This kind of success shows that a more advanced model, like Random Forest, should be used in automated systems to observe data pipeline performance.

V. CONCLUSION

This research paper has concluded that the Random Forest Regressor has been more effective in predicting house prices in California Housing Prices data than the Linear Regression, with a higher predictive and stronger model fitting. The findings reveal the applicability of the autonomous models in handling complex data and non-linear relationships. Different techniques like the Random Forest can significantly improve the performance of the data pipelines with an increasing need of the automated data engineering.

Future Scope

Experiments can be implemented further to determine various other applications of more advanced machine learning algorithms such as XGBoost or Neural Networks to enhance the accuracy of predictions even more. Practical work on incremental pipelines, Scalability of autonomous pipelines with dynamic data would open new possibilities to optimize data workflows and resources.

VI. REFERENCES

[1] Coronado, E., Behravesh, R., Subramanya, T., Fernandez-Fernandez, A., Siddiqui, M.S., Costa-Pérez, X. and Riggio, R., 2022. Zero touch management: A survey of network automation solutions for 5G and 6G networks. *IEEE Communications Surveys & Tutorials*, *24*(4), pp.2535-2578.

[2] Sethupathy, U.K.A., 2023. Zero-touch DevOps: A GenAI-orchestrated SDLC automation framework. *World Journal of Advanced Engineering Technology and Sciences*, *8*(2), pp.420-433.

[3] De Sousa, N.F.S., Islam, M.T., Mustafa, R.U., Perez, D.A.L., Rothenberg, C.E. and Gomes, P.H., 2022. Machine Learning-Assisted Closed-Control Loops for Beyond 5G Multi-Domain Zero-Touch Networks: NFS de Sousa et al. *Journal of Network and Systems Management*, *30*(3), p.46.

[4] Angui, B., Corbel, R., Rodriguez, V.Q. and Stephan, E., 2022, January. Towards 6G zero touch networks: The case of automated Cloud-RAN deployments. In *2022 IEEE 19th annual consumer communications & networking conference (CCNC)* (pp. 1-6). IEEE.

[5] Zhou, W., Jiang, Y.E., Li, L., Wu, J., Wang, T., Qiu, S., Zhang, J., Chen, J., Wu, R., Wang, S. and Zhu, S., 2023. Agents: An open-source framework for autonomous language agents. *arXiv preprint arXiv:2309.07870*.

[6] Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K. and Zhang, S., 2023. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.

[7] Nascimento, N., Alencar, P. and Cowan, D., 2023, September. Self-adaptive large language model (llm)-based multiagent systems. In *2023 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)* (pp. 104-109). IEEE.

[8] Zheng, Q., Xu, Z., Choudhry, A., Chen, Y., Li, Y. and Huang, Y., 2023. Synergizing human-AI agency: a guide of 23 heuristics for service co-creation with

LLM-based agents. *arXiv preprint arXiv:2310.15065*.

[9] Xu, F., Zhang, J., Gao, C., Feng, J. and Li, Y., 2023. Urban generative intelligence (ugi): A foundational platform for agents in embodied city environment. *arXiv preprint arXiv:2312.11813*.

[10] Tarkoma, S., Morabito, R. and Sauvola, J., 2023. AI-native interconnect framework for integration of large language model technologies in 6G systems. *arXiv preprint arXiv:2311.05842*.

[11] Bariah, L., Zou, H., Zhao, Q., Mouhouche, B., Bader, F. and Debbah, M., 2023, December. Understanding telecom language through large language models. In *GLOBECOM 2023-2023 IEEE Global Communications Conference* (pp. 6542-6547). IEEE.

[12] Musa, A.A., Hussaini, A., Qian, C., Guo, Y. and Yu, W., 2023. Open radio access networks for smart IoT systems: State of art and future directions. *Future Internet*, *15*(12), p.380.

[13] Pamisetty, V., 2023. Leveraging artificial intelligence for strategic decision-making in tax administration and policy design. *Available at SSRN 5276644*.

[14] Zhou, W., Jiang, Y.E., Li, L., Wu, J., Wang, T., Qiu, S., Zhang, J., Chen, J., Wu, R., Wang, S. and Zhu, S., 2023. Agents: An open-source framework for autonomous language agents. *arXiv preprint arXiv:2309.07870*.

[15] Huang, Y., Chen, Y. and Li, Z., 2023. Applications of large scale foundation models for autonomous driving. *arXiv preprint arXiv:2311.12144*.

[16] Zharikova, D., Kornev, D., Ignatov, F., Talimanchuk, M., Evseev, D., Petukhova, K., Smilga, V., Karpov, D., Shishkina, Y., Kosenko, D. and Burtsev, M., 2023, July. DeepPavlov dream: platform for building generative AI assistants. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)* (pp. 599-607).

[17] Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K. and Zhang, S., 2023. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.

[18] Goertzel, B., 2023. Generative ai vs. agi: The cognitive strengths and weaknesses of modern llms. *arXiv preprint arXiv:2309.10371*.

[19] Jeong, C., 2023. A study on the implementation of generative AI services using an enterprise data-based LLM application architecture. *arXiv preprint arXiv:2309.01105*.

[20] González-Santamarta, M.Á., Rodríguez-Lera, F.J., Guerrero-Higueras, Á.M. and Matellán-Olivera, V., 2023. Integration of large language models within cognitive architectures for autonomous robots. *arXiv preprint arXiv:2309.14945*.

[21] Vemprala, S., Chen, S., Shukla, A., Narayanan, D. and Kapoor, A., 2023. Grid: A platform for general robot intelligence development. *arXiv preprint arXiv:2310.00887*.

[22] Yang, Z., Jia, X., Li, H. and Yan, J., 2023. Llm4drive: A survey of large language models for autonomous driving. *arXiv preprint arXiv:2311.01043*.

[23] Huang, Q., Vora, J., Liang, P. and Leskovec, J., 2023. Benchmarking large language models as ai research agents. In *NeurIPS 2023 foundation models for decision making workshop*.

[24] Tarkoma, S., Morabito, R. and Sauvola, J., 2023. AI-native interconnect framework for integration of large language model technologies in 6G systems. *arXiv preprint arXiv:2311.05842*.

[25] Valmeekam, K., Marquez, M., Sreedharan, S. and Kambhampati, S., 2023. On the planning abilities of large language models-a critical investigation. *Advances in neural information processing systems*, *36*, pp.75993-76005.

[26] Nascimento, N., Alencar, P. and Cowan, D., 2023. Gpt-in-the-loop: Adaptive decision-making for multiagent systems. *arXiv preprint arXiv:2308.10435*.

[27] Wang, Y., Jiao, R., Zhan, S.S., Lang, C., Huang, C., Wang, Z., Yang, Z. and Zhu, Q., 2023. Empowering autonomous driving with large language models: A safety perspective. *arXiv preprint arXiv:2312.00812*.