asTech Insights: The GenAI approach to Customized Collision Repair Recommendations

Monika Daryani
Repairify Inc.
Plano, TX
Email: monika.daryani@repairify.com

Samuel Poradish
Repairify Inc.
Plano, TX
Email: samuel.poradish@repairify.com

Abstract— **Leveraging the power of large language models, asTech Insights marks a revolutionary stride in automotive repair, skillfully interpreting complex vehicle diagnostics to produce custom repair instructions. This innovation, born from creative problem-solving, boasts an impressive accuracy rate exceeding 90% and operates over 60 times faster than conventional methods, dramatically cutting operational expenses. asTech Insights elevates service quality in the industry through using certified recommendations, ensuring swift, evidence-based, and precise repair tactics. The paper thoroughly explores the sophisticated technical design of asTech Insights, examines its real-world applications, and underscores its capacity to redefine the automotive repair and maintenance sector's future.**

I. INTRODUCTION

In auto repair shops, the vehicle repair process initiates with the establishment of a work order and the execution of diagnostic tests to pinpoint necessary repairs. Various Repairify devices, such as local scanners*, original equipment manufacturer (OEM) scanners, ADASThink*, and others,* are prevalent and trusted for these diagnostic procedures. Employed in thousands of auto repair shops, these devices' service levels are tailored according to each shop's specific contractual agreement.

Local scanning services are categorized into two levels:

- **Level 1 - Local Scan Tool Only** This service involves using a local scan device that provides a list of Diagnostic Trouble Codes (DTCs)*, their associated modules, and statuses. It's a cost-effective option but relies heavily on the mechanic's knowledge to identify and rectify faults accurately. Incorrect fixes can lead to increased costs and complications in insurance claims. This level contract demands highly skilled mechanics in the bodyshop familiar with car faults, post-repair testing, and the latest Advanced Driver-Assistance Systems (ADAS)* features that may require re-calibration. For example, if there is damage on front bumper, the mechanic should be aware that after fixing or replacing front bumper they should re-calibrate the front radar sensors.
- **Level 2 - Local Scan with Tech Support** A premium guided service where the scan tool provides DTCs, modules, statuses, and generates a request for asTech technicians'* review. Expert technicians provide a list of repair recommendations, re-calibration requirements, and post-repair tests. This aids mechanics in planning services and facilitates insurance quoting. While more comprehensive, this option is more expensive compared to Level 1 and can have a longer turnaround time.

asTech Insights has been introduced to bridge the gap between these two service levels. This solution offers initial recommendations, without the interactive element of expert technicians. Mechanics receive a clear repair plan or confirmation of their existing plan, enhancing efficiency. Key advantages of asTech Insights include:

- Speed: It delivers results in seconds compared to up to 30 minutes that an asTech technician might take.
- Certified Recommendations: To resolve potential conflicts of opinion, certain recommendations are certified for added clarity.
- Cost efficiency: This service is available at lower cost than level 2 service.

The asTech Insights[1] system marks a noteworthy advancement in the automotive industry, enhancing the efficiency of diagnostic and repair processes in auto repair shops. It stands out for its ability to combine affordability, speed, and dependability in vehicle servicing, streamlining the overall workflow.

II. RELATED WORK

The integration of Artificial Intelligence (AI) in vehicular technology, particularly in autonomous vehicles, has been a significant advancement. As outlined by Ma et al.[2], AI plays a crucial role in the three primary applications of autonomous vehicles: 1) perception; 2) localization and mapping; and 3) decision-making. Beyond autonomous vehicles, AI has also seen substantial development in car manufacturing and Advanced Driver-Assistance Systems (ADAS) [3], with numerous corporations and startups driving the field forward.

LLM's contribution to recommendation systems is also noteworthy, as highlighted in the work of Fan et al. [4] which explores the application of Large Language Models (LLMs) in three paradigms, namely pre-training, fine-tuning, and prompting. Since the introduction of ChatGPT, LLMs have expanded across various sectors. Kasneci et. al. [5] discusses Chat GPT's potential benefits and challenges in educational

*See Appendix A for definition of terms

applications. Thirunavukarasu et. al. [6] discusses potential effectiveness of LLMs in clinical, educational and research work in medicine. Li et. al. [7] discusses existing solutions and guidance for adoption of LLMs in finance.

In autonomous driving, LLMs are gaining traction. Cui et al.[8] recently demonstrated how LLMs can enhance the driving capabilities of autonomous vehicles. Additionally, their work [9] delves into the effects of Multimodal Large Language Models on autonomous driving.

Despite these advancements, the application of AI or LLM in the diagnosis of vehicle issues and understanding of car sensor data remains relatively unexplored. Given AI's proven effectiveness in diagnosing cancer in early stages as discussed in Fitzgerald et. al. [10] and issues in other fields like e-commerce to detect frauds as discussed in Daryani et. al. [11], there is a strong potential for similar success in the automotive diagnosis realm. This area presents an opportunity for groundbreaking work that could revolutionize the approach to vehicular diagnostics and maintenance.

III. METHODOLOGY

A. Data Collection and Processing

We begin with setting up a data warehouse environment suitable for large-scale data processing. A team of analysts and architects took years of application data and built a star schema* relational database with millions of records capable of supporting analytical work. We performed SQL data mining focused on creating a JSON data structure, encompassing tables for normalized training sets and temporary staging from natural language technician inputs in four general phases:

- Parse and extract DTC information for the scanreading report, including description, module, symptom byte, and status byte. This had to support all viable fault formats across manufacturers and handle ambiguities due to different technician input styles.
- Use a combination of RegEx rules to normalize and segment blocks of recommendation text, which often contained multiple recommendations for different sets of DTCs across modules in many different input formats. These rules combined grammatical analysis with recognition of references to fault codes/canned text to create logical partitions.
- Map the fault code information from step 1 to viable recommendation blocks from step 2 for each work order; a dynamic, ranked ten-stage matching process was used to ensure only the highest desirable match for the available data type and quality would prevail.
- Clean the resultant matched recommendations sentence-by-sentence to prevent undesirable text from coming through using sets of RegEx patterns to look for any artifacts or grammatical tendencies and either remove or replace them to create a more cohesive dataset verbiage.

Input for SQL script:

```
Fault C2122 indicates that transmitter 2
is not communicating with ...

Complete a vehicle road test of at least
5-6 miles, making ...
```

*See Appendix A for details

Output for SQL script / Input for python script:

```
"FAULTS": [
        {
        "DESCRIPTION": "Cannot Receive
        a Data From the Transmitter ID2",
        "DTC": "C2122",
        ...
        }
        ],
        "MODULE": "Tire Pressure
        Monitoring System Module"
      ],
"RECOMMENDATION": "Verify proper operation
of transmitter 2. If transmitter 2 is
found to be defective, replace ..",
```

Python Scripting: The Python script complemented the previous stage by further refining the data for machine learning applications, by taking SQL JSON output as input. The script performed general data cleaning and transformation, focusing on removing irrelevant columns, handling duplicates, and aggregating data based on specific criteria. We also removed redundant work orders and prioritized recommendations that were certified by our tech content authors. and prioritised certified recommendations provided by our tech content authors. When we encountered a conflict, i.e., multiple recommendations for the same car issue, we prioritized highly repeated recommendations based on the assumption that repeated recommendation should have resolved the given issue when used the first time. The script also incorporated an approach to combine rows of data into a structured string format (as shown below) suitable for the generative model, ensuring the data was machine-learning-ready.

Output for python script:

```
"source_text" : "Recommend for Car is a
Lexus NX 300. It has a Current fault,
represented by DTC code C2122, which ..."

"target_text": "Fault C2122 indicates
that transmitter 2 is not
communicating with ...
Verify proper operation of
transmitter 2. If transmitter 2 is
found to be defective, replace ...

Complete a vehicle road test of at least
5-6 miles, making ..."
```

The script's modular design allowed for flexibility in data handling, accommodating various car makes and adjusting to different data sizes.

Together, these scripts facilitated a thorough and efficient process of turning raw automotive diagnostic data into a structured and clean dataset, ready for advanced analysis and machine learning model training.

B. Model Training and Fine-Tuning

The model training and fine-tuning phase leveraged the capabilities of PyTorch, PyTorch Lightning, and the Huggingface Transformers library. The integration of these powerful tools allowed for an efficient and scalable approach to training our AI models, specifically tailored for the T5/Flan-T5 models.

PyTorch provided the foundational framework for deep learning operations. It facilitated dynamic computation graphs and efficient memory usage, essential for training large models like T5/Flan-T5 (Imambi et al., 2021)[12]

PyTorch Lightning streamlined the training process. It enabled efficient batch processing, maximized GPU utilization, and simplified the training loop with advanced features such as early stopping, which significantly enhanced performance monitoring and optimization (Sawarkar et al., 2022) [13]

The Huggingface Transformers library was crucial for accessing pre-trained models and managing various language model operations. It provided extensive support for Transformer-based models like T5/Flan-T5, simplifying implementation and reducing the need for extensive coding (Wolf et al., 2019) [14]

We developed Custom data handling and preprocessing classes to optimize data structuring and tokenization, aligned with the T5/Flan-T5 models' requirements. This ensured that the data fed into the models was in the most efficient format for training and inference.

The training module, designed using LightningModule, adeptly managed the intricacies of training, validation, and testing phases. It included mechanisms for calculating loss and logging performance metrics. The AdamW optimizer was chosen for its efficiency in handling weight optimization, particularly in fine-tuning pre-trained models (Loshchilov & Hutter, 2017) [15]

Additionally, the training framework supported adjustable command-line arguments for versatile model configuration, enabling dynamic modifications in training parameters like batch size, epochs, and token lengths. This adaptability was essential for customizing the training process to suit various dataset sizes and characteristics.

Our experimental approach involved segmenting the dataset based on vehicle make and setting up multiple models. We tested different configurations, including batch sizes, epoch counts, and choosing between T5 and Flan-T5 models. The most effective model emerged with a configuration of 3 epochs, a batch size of 4, using Flan-T5, and setting maximum source and target token lengths to 512. The training on g5.12xlarge instance took 4 hours to train for each model. This optimized setting led to significant improvements in the model's performance and accuracy.

C. Integration of LLM into Repairify's System

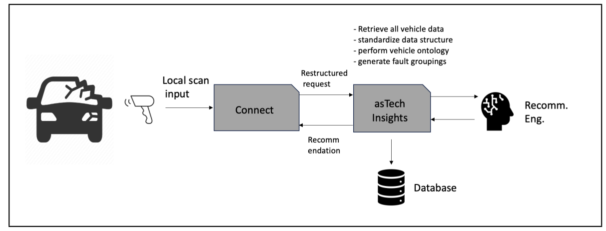

Fig. 1. End-to-end asTech Insights API flow

The process flow of asTech Insights, as depicted in Figure 1, initiates with a local vehicle scan. The scan tool transmits data to asTech connect, the asTech scan interface, which reformats the data into the desired JSON structure. This data is then sent via a POST request to asTech Insights, initiating a series of operations culminating in a complete data entry in the database. Concurrently, the Insights API triggers a step function to retrieve additional relevant data and preprocess it, ensuring it aligns with our training data format.

1) Retrieve all vehicle data and restructure: Utilizing the vehicle's year, make, and model, the system identifies corresponding Vehicle Configuration database (VCDB) Make ID, VCDB Model ID, and Base ID. In certain cases, the vehicle's VIN (Vehicle Identification Number) is used for these details. We add this data and convert the whole JSON into standardized keys, ensuring consistency throughout the process. The system maps aftermarket descriptions and statuses to OEM terminology, adding additional keys with these values where applicable.

2) Generate fault groupings: During the data collection phase, as detailed in Section III-A, we organized our training data into clusters of Diagnostic Trouble Codes (DTCs) that commonly occur together. However, in the production environment, such pre-defined groupings were not available. To overcome this challenge, we introduced an additional lambda function dedicated to each work order, specifically designed to generate groupings of DTCs.

This process was facilitated by employing the FPgrowth (Frequent Pattern Growth) algorithm, as cited in Han et al.[16], to discern frequent patterns and associations among the DTC groupings. (Frequent Pattern Growth Algorithm is the method of finding frequent patterns without candidate generation.) Subsequently, Depth First Search (DFS) was utilized to apply these associations in creating a structured JSON file that maps interconnected DTCs. An example of this structure is demonstrated below:

```
{
"B261E": ["B261C", "B261F", "B261D"],
"B261F": ["B261C", "B261E", "B261D"],
"B2638": ["B2641"],
"B2641": ["B2638"],
...
```

To fine-tune the FPGrowth algorithm, we focused on the confidence metric. Our approach involved plotting graphs to compare the confidence score against the current data groupings. Through the application of the elbow method or the "knee of the curve" technique, we were able to identify the most effective confidence score for our algorithm.

Most work orders typically have a single fault grouping. However, if a work order has multiple fault groupings, then multiple recommendation calls are made for each grouping. Our recommendation engine, a Flan-T5 model hosted on Amazon Sagemaker, receives these recommendation calls. By deploying the model on SageMaker, we are able to demonstrate its practical application in a cloud environment and make it scalable.

Once the above-mentioned steps are completed, which usually takes around 7 seconds, all the processed information is carefully updated in the database. After that, the updated data is posted back to the asTech Connect interface.

IV. APPLICATION

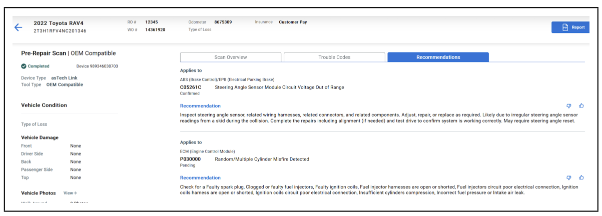

Fig. 2. Real life case study : Irregularities in steering angle sensor

Figure 2 shows a real life example for irregularities in steering angle sensor and misfires in engine cylinders. This recommendation identifies that the irregularity in signal data reported by the steering angle sensor and details for the consumer what additional steps are required to diagnose and resolve the issue. In this instance the procedure is a calibration, which requires specialized skills and equipment to complete. By identifying this early in the repair process the shop is provided with the opportunity to accurately quote and plan for required repairs.

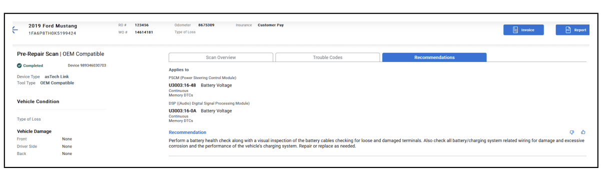

Fig. 3. Real life case study: battery Voltage issue

Figure 3 shows another real life example. In this example two DTCs have been addressed as a group because they both represent symptoms of a single root issue, in this case a battery voltage issue. Addressing the DTCs as a collection helps the consumer understand their relationship to one another and allows the LLM to generate a more detailed explanation of the required repairs. This recommendation contains important information about the components required to complete the repair, as well as precautions to ensure the procedures are completed safely.

V. RESULTS

A. *Efficiency Analysis*

Average time taken by AI Diagnosis is 7.1 seconds, while average manual diagnostic time is 7 minutes 29 seconds.

Using an LLM to generate diagnostic recommendations significantly reduces the human labor invested in the process for both our technicians and customers. On average, an asTech technician specializing in diagnostics spends 7 minutes and 29 seconds interpreting the scan readings and authoring a diagnostic report, while the LLM is capable of producing comparable results in an average of 7.1 seconds. This substantially reduces the cost associated with providing the service and allows it to be offered at a more competitive price point. The near instantaneous response also enables consumers to integrate the usage of the recommendations in their workflow without any meaningful extension of cycle time, which is a key performance metric in automotive repair.

B. *Accuracy Evaluation:*

TABLE I

AUDIT RESULT FOR PRODUCTION MODEL

Make	Audit Precision	Ratio in dataset
Ford	0.909	0.111
GM	0.907	0.144
Stellantis	0.912	0.114
Toyota	0.912	0.191
Nissan	0.898	0.164
Honda	0.906	0.215
Hyundai & Kia	0.944	0.060
Overall	0.9093	1.000

We performed an audit of the production data, with results presented in TABLE I. The data was meticulously verified by our content authors, who are highly skilled technicians. Precision was calculated using the formula:

$$Precision = \frac{No.\ of\ accepted\ recommendations}{All\ recommendations} \quad (1)$$

The findings reveal an impressive precision rate of **90.93%** for our recommendations compared to those made by content autors. (We assume human content author recommendations as 100% correct.) Efforts are ongoing to enhance this precision further through data curation and the implementation of other strategic improvements.

VI. DISCUSSION

A. *Implications for Automotive Repair Industry*

Integrating generative AI in the automotive repair industry represents a pivotal shift, promising to transform traditionally manual processes with sophisticated, data-driven solutions. This technological leap is particularly relevant as modern vehicles increasingly embody complex data systems and advanced technologies. GenAI stands out for its ability to meticulously analyze comprehensive diagnostic data, which is key to formulating more precise and effective repair strategies. Such an advancement is poised to revolutionize automotive repair in several ways:

- Reduction in Diagnostic Time: With AI-driven analysis, vehicular issues diagnosis time could be significantly reduced, enabling quicker responses to repair needs.
- Enhanced Repair Accuracy: The precision of GenAI in interpreting complex data can lead to more accurate repair recommendations, minimizing the likelihood of errors or misdiagnoses.
- Increased Workshop Efficiency: The overall efficiency of automotive workshops can witness an upsurge, as AI-driven recommendations streamline the repair process, potentially reducing labor hours and operational costs.
- Upskilling of Technicians: As GenAI handles the data-intensive aspects of modern vehicles, technicians can focus on upskilling and adapting to advanced vehicle technologies. This not only improves their capability but also elevates the service standards and reliability of the automotive repair industry.
- Customer Satisfaction and Trust: With quicker and more reliable services, customer satisfaction is likely to increase, fostering greater trust in automotive repair services.
- Home Diagnostics Potential: The technology also opens avenues for home-based diagnostics and solutions, making vehicle maintenance more accessible for individuals.

In essence, GenAI's application in automotive repair aligns with the industry's progression towards embracing digital transformation, offering a future where technology and human expertise coalesce to redefine automotive service excellence.

B. Challenges and Limitations

Implementing generative AI in automotive repair involves several complex challenges and limitations.

First is the quality and comprehensiveness of the training data. The effectiveness of AI recommendations depends on the data quality, and incomplete or biased datasets can lead to substandard or incorrect advice. We performed extensive data mining and cleaning for this project to ensure high-quality input data,

In automotive field, technicians may offer different steps and viewpoints for the same issue. To manage this, we certified a number of recommendations and prioritized the textual content within them for the recommender system. Increasing the number of these certified recommendations and emphasizing those frequently utilized by technicians helps to converge on the most effective solutions.

The complex nature of the model requires significant computational resources, particularly for updating and fine-tuning with new data. We are exploring Retrieval Augmented Generation (RAG) [17] to minimize the need for extensive re-training for minor adjustments.

Incorporating advanced AI technology into existing systems and processes demanded considerable engineering efforts. In real-world scenarios, unforeseen vehicle issues or context-specific challenges may occur, which the model may not fully capture. In such cases, the invaluable role of human expertise becomes evident.

There's a risk of the AI model generating incorrect or irrelevant information (hallucinations) when adequate context for the input is missing. Currently, we are stopping generation of recommendations which cases cause hallucinations, case-by-case basis. It's essential to take a balanced approach that combines advanced technology with human expertise and continuous data refinement.

Hence, while generative AI holds transformative potential for the automotive repair industry, these challenges highlight the need for careful consideration and a balanced approach.

VII. Conclusion

In conclusion, this paper represents a groundbreaking stride in automotive repair. By harnessing large language models, this system introduces an efficient, data-driven approach to vehicle diagnostics and repair. This paper detailed the development, implementation, and integration of asTech Insights, showcasing its potential to enhance repair accuracy, reduce diagnostic time, and improve overall workshop efficiency. The model, while highly effective, also encounters challenges such as data quality and computational demands, highlighting the need for continuous improvement and integration of human expertise. This study demonstrates the model's impressive capabilities in the automotive repair industry and sets a precedent for future AI-driven innovations in various fields.

Appendix

A. Glossary of terms

DTC: A DTC, short for Diagnostic Trouble Code, is a code used to diagnose malfunctions in a vehicle. DTCs are also called engine vehicle fault codes codes, and can be read with a scanner that plugs directly into the port of a vehicle.

ADAS: Advanced Driver Assistance Systems (ADAS) are electronic systems which assist drivers in order to avoid accidents. ADAS uses automated technology, such as sensors and cameras, to detect nearby obstacles or driver errors, and respond accordingly. ADAS can enable various levels of autonomous driving, depending on the features installed in the car. They include features like pedestrian detection, lane departure warning, adaptive cruise control, and automatic parking.

ADASThink: ADASThink is used by bodyshop technicians for initial damage assessment and repair estimation. It utilizes a backend vehicle equipment database to provide a preliminary report, aiding in insurance processes.

asTech Device: After creating a work order, technicians use the asTech device for vehicle scanning, retrieving Diagnostic Trouble Codes (DTCs) that highlight module errors. The completed scan report, prepared by asTech technicians, includes repair recommendations and is shared with bodyshop technicians. A significant feature of the asTech system is its capability to detect both current and potential ADAS issues. While asTech is used pre- and post-repair, ADASThink is generally used once for initial assessments when a car arrives.

OEM Scanner: Original Equipment Manufacturer (OEM) scanners, supplied by car manufacturers, are used for diagnostics and supported remotely by asTech.

asTech Technician: asTech technicians, are remote technician support provided by asTech. They support remote scanning of automotive vehicles and provide recommendations to fix them. They also clarify any doubts that the mechanic might have. They are expert car technicians with years of hands-on experience.

Local Scanner: Local scanners are on-site scan tools used in auto repair shops to identify DTCs directly from vehicles.

Star Schema: Star schema is a multi-dimensional data model used to organize data in a database used for organizing and analyzing large data sets in databases, particularly in data warehouses.

ACKNOWLEDGMENT

Our gratitude extends to Josh for his exceptional data mining efforts. His proficiency in SQL was a cornerstone in making this project a reality. We also extend our thanks to Matthew for his engineering acumen, which was instrumental in transitioning the model to production.

We thank James Smith and Fadi Oweis for their crucial role in assessing the model's performance and offering valuable feedback for data improvement. Special appreciation is also due to Sue and Nick for their skilled project management. Additionally, we are grateful to Jon Delgado and Chris Hollingsworth for their consistent support and guidance throughout the project's progression. Their collective expertise significantly contributed to the project's success.

REFERENCES

[1] (2023) astech insights. [Online]. Available: https://astech.com/our-solutions/astech-insights/

[2] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: a survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.

[3] A. Schroer and J. Powers. (2024) Artificial intelligence in cars: Examples of ai in the auto industry. [Online]. Available: https://builtin.com/artificial-intelligence/artificial-intelligence-automotive-industry

[4] W. Fan, Z. Zhao, J. Li, Y. Liu, X. Mei, Y. Wang, J. Tang, and Q. Li, "Recommender systems in the era of large language models (llms)," *arXiv preprint arXiv:2307.02046*, 2023.

[5] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günnemann, E. Hüllermeier, S. Krusche, G. Kutyniok, T. Michaeli, C. Nerdel, J. Pfeffer, O. Poquet, M. Sailer, A. Schmidt, T. Seidel, M. Stadler, J. Weller, J. Kuhn, and G. Kasneci, "Chatgpt for good? on opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103, p. 102274, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1041608023000195

[6] A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan, and D. S. W. Ting, "Large language models in medicine," *Nature medicine*, vol. 29, no. 8, pp. 1930–1940, 2023.

[7] Y. Li, S. Wang, H. Ding, and H. Chen, "Large language models in finance: A survey," in *Proceedings of the Fourth ACM International Conference on AI in Finance*, 2023, pp. 374–382.

[8] C. Cui, Y. Ma, X. Cao, W. Ye, and Z. Wang, "Drive as you speak: Enabling human-like interaction with large language models in autonomous vehicles," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 902–909.

[9] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang, J. Chen, J. Lu, Z. Yang, K.-D. Liao, *et al.*, "A survey on multimodal large language models for autonomous driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 958–979.

[10] R. C. Fitzgerald, A. C. Antoniou, L. Fruk, and N. Rosenfeld, "The future of early cancer detection," *Nature medicine*, vol. 28, no. 4, pp. 666–677, 2022.

[11] M. Daryani and J. Caverlee, "Identifying hijacked reviews," in *Proceedings of The 4th Workshop on e-Commerce and NLP*. Association for Computational Linguistics, 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.ecnlp-1.9

[12] S. Imambi, K. B. Prakash, and G. Kanagachidambaresan, "Pytorch," *Programming with TensorFlow: Solution for Edge Computing Applications*, pp. 87–104, 2021.

[13] K. Sawarkar, *Deep Learning with PyTorch Lightning: Swiftly Build High-performance Artificial Intelligence (AI) Models Using Python*. Packt Publishing Ltd, 2022.

[14] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019.

[15] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[16] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '00. New York, NY, USA: Association for Computing Machinery, 2000, p. 1–12. [Online]. Available: https://doi.org/10.1145/342009.335372

[17] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," 2021.