smart cities

MDPI

Article

Multitask Learning for Crash Analysis: A Fine-Tuned LLM Framework Using Twitter Data

Shadi Jaradat [1,2,*], Richi Nayak [2], Alexander Paz [3], Huthaifa I. Ashqar [4,5] and Mohammad Elhenawy [1,2]

[1] Centre for Accident Research & Road Safety-Queensland, Queensland University of Technology, Brisbane, QLD 4000, Australia; mohammed.elhenawy@qut.edu.au
[2] Centre of Data Science, Queensland University of Technology, Brisbane, QLD 4000, Australia; r.nayak@qut.edu.au
[3] School of Civil Engineering, Queensland University of Technology, Brisbane, QLD 4000, Australia; alexander.paz@qut.edu.au
[4] Civil Engineering Department, Arab American University, 13 Zababdeh, Jenin P.O. Box 240, Palestine; huthaifa.ashqar@aaup.edu
[5] Artificial Intelligence Program, Fu Foundation School of Engineering and Applied Science, Columbia University, 500 W 120th St, New York, NY 10027, USA
* Correspondence: shadi.jaradat@hdr.qut.edu.au

Highlights:

What are the main findings?

- Demonstrates the effectiveness of a novel multitask learning (MTL) framework utilizing large language models (LLMs) for real-time analysis of road traffic crashes (RTCs) through the integration of social media data.
- Fine-tuning GPT-2 for language modeling demonstrated that it outperformed baseline models, including GPT-4o mini in zero-shot mode and XGBoost, across various classification and information retrieval tasks. This study benchmarks the performance of the fine-tuned GPT-2 model against these baselines, highlighting its superior performance in these tasks.
- The study collected and curated a dataset of 26,226 RTC-related tweets from Australia over a year. This dataset extracted fifteen unique features, with six used in classification tasks and nine in information retrieval tasks.
- Developed an advanced automated labeling system using GPT-3.5, followed by rigorous expert verification to ensure the accuracy and reliability of feature extraction from tweets. The resulting meticulously curated dataset serves as a foundational resource for training and validating subsequent models, establishing a new standard for RTC analysis.

What is the implication of the main finding?

- Offers a transformative approach to traffic safety analytics, providing detailed, timely insights crucial for emergency responders, urban planners, and policymakers.
- By leveraging cutting-edge AI techniques within an MTL framework, this study demonstrates a transformative approach to real-time RTC analysis, setting the stage for future advancements in the field.
- The curated dataset generated in this research not only advances traffic safety measures but also serves as a valuable resource for extracting insights, developing models, and conducting further research. This resource provides a solid foundation for future studies aimed at enhancing road safety.

Abstract: Road traffic crashes (RTCs) are a global public health issue, with traditional analysis methods often hindered by delays and incomplete data. Leveraging social media for real-time traffic safety analysis offers a promising alternative, yet effective frameworks for this integration are scarce. This study introduces a novel multitask learning (MTL) framework utilizing large language models (LLMs) to analyze RTC-related tweets from Australia. We collected 26,226 traffic-related tweets from May 2022 to May 2023. Using GPT-3.5, we extracted fifteen distinct features categorized

Citation: Jaradat, S.; Nayak, R.; Paz, A.; Ashqar, H.I.; Elhenawy, M. Multitask Learning for Crash Analysis: A Fine-Tuned LLM Framework Using Twitter Data. *Smart Cities* **2024**, *7*, 2422–2465. https://doi.org/10.3390/smartcities7050095

Academic Editor: Pierluigi Siano

Received: 16 July 2024
Revised: 26 August 2024
Accepted: 29 August 2024
Published: 1 September 2024

into six classification tasks and nine information retrieval tasks. These features were then used to fine-tune GPT-2 for language modeling, which outperformed baseline models, including GPT-4o mini in zero-shot mode and XGBoost, across most tasks. Unlike traditional single-task classifiers that may miss critical details, our MTL approach simultaneously classifies RTC-related tweets and extracts detailed information in natural language. Our fine-tunedGPT-2 model achieved an average accuracy of 85% across the six classification tasks, surpassing the baseline GPT-4o mini model's 64% and XGBoost's 83.5%. In information retrieval tasks, our fine-tuned GPT-2 model achieved a BLEU-4 score of 0.22, a ROUGE-I score of 0.78, and a WER of 0.30, significantly outperforming the baseline GPT-4 mini model's BLEU-4 score of 0.0674, ROUGE-I score of 0.2992, and WER of 2.0715. These results demonstrate the efficacy of our fine-tuned GPT-2 model in enhancing both classification and information retrieval, offering valuable insights for data-driven decision-making to improve road safety. This study is the first to explicitly apply social media data and LLMs within an MTL framework to enhance traffic safety.

Keywords: road traffic crashes; social media data analysis; large language models; multitask learning; GPT

1. Introduction

Analyzing social networks offers a rapid and cost-effective means to gain diverse insights into various phenomena, such as violence, natural disasters, and road traffic crashes (RTCs). Traditional RTC analysis relies heavily on direct reporting and monitoring systems, often delayed and inaccurate due to under-reporting. With the advent of social media platforms like Twitter, now rebranded as X, there is a real-time alternative for capturing RTC data. These platforms enable immediate information dissemination from eyewitnesses, providing valuable insights into incidents and public reactions [1,2]. However, unstructured data's sheer volume, velocity, and veracity pose significant challenges for traditional data processing methods [3].

Recent advancements in large language models (LLMs) such as GPT-3.5 and GPT-4 have significantly transformed the data analysis landscape, particularly in applications requiring the interpretation of vast and complex datasets. These models, known for their extensive training on diverse datasets, have demonstrated remarkable adaptability and performance across a wide range of tasks [1,2]. This shift towards utilizing LLMs through prompt engineering and fine-tuning methods has challenged the traditional approach of developing specialized models for each task, suggesting a more unified framework where a single model can handle multiple tasks effectively [3,4].

The primary motivation behind our research is to explore the potential of LLMs in the critical domain of traffic safety, explicitly focusing on detecting and analyzing road traffic crashes (RTCs) using social media data. Traditional RTC analysis has primarily relied on direct reporting and monitoring systems, which are often delayed and prone to inaccuracies due to under-reporting. However, the advent of social media platforms like Twitter provides a real-time alternative for capturing RTC data, enabling immediate information dissemination from eyewitnesses and offering valuable insights into incidents and public reactions [5,6]. Despite this potential, social media data's sheer volume, velocity, and unstructured nature pose significant challenges for traditional data processing methods [7].

To address these challenges, our study employs a multitask learning (MTL) framework leveraging LLMs, specifically fine-tuning GPT-2 and utilizing GPT-4 in a zero-shot setting. This approach allows us to explore the capabilities of LLMs in handling both classification and information retrieval tasks related to RTCs, thereby enhancing the efficiency and accuracy of traffic crash analysis. While traditional methods often involve single-task models requiring extensive feature engineering, our framework integrates multiple tasks within a single model, demonstrating how LLMs can surpass conventional methods in both scope and performance [8,9]. Our framework integrates classification and information

retrieval tasks within a fine-tuned language model, specifically GPT-2. This approach enhances the detection of traffic crashes and the extraction of detailed insights from social media data, surpassing traditional methods [10,11].

In particular, our research investigates the efficacy of LLMs in processing social media data to detect the occurrence of a traffic crash, assess its severity, identify collision types, and determine public sentiment and emotions related to the incidents. Additionally, the information retrieval tasks, such as the number of injured or deceased, accident location, and contributing factors, offer detailed contextual information essential for comprehensive accident analysis and improving preventive measures [3,9]. These critical tasks provide immediate insights for emergency response teams, law enforcement, and urban planners to implement effective road safety measures [7,8,10,11]. Moreover, by comparing the performance of fine-tuned GPT-2 with GPT-4's zero-shot capabilities, we aim to establish a baseline that highlights the advantages of task-specific fine-tuning over general-purpose models [12].

This study also addresses the critical challenges of data scarcity and imbalance, which frequently impede the training of reliable ML models. By efficiently utilizing available data to simultaneously train on multiple related tasks, our model ensures that even limited or imbalanced datasets can yield highly informative and dependable results. This capability is particularly crucial in regions where RTC reporting is insufficient and conventional data collection methods are impractical [13,14].

Our research aims to develop a methodology for multitask learning using social media data analysis. By leveraging advanced NLP technologies, MTL seeks to enhance the real-time analysis of road traffic crashes by systematically examining social media platforms, particularly Twitter. This approach promises to enable more effective accident response management strategies and improve public safety initiatives by providing immediate and accurate insights into traffic incidents. Our objectives are as follows:

- Develop a multitask learning framework: design a comprehensive MTL framework that integrates classification and information retrieval tasks, surpassing traditional multi-class classifiers.
- Label tweets and verify via domain experts: implement an automated labeling system using GPT-3.5, followed by expert verification to ensure the accuracy and reliability of the extracted features. The resulting dataset is used for fine-tuning the GPT-2 model.
- Fine-tune GPT models for multitask objectives: fine-tune GPT-2 for simultaneous classification and information extraction tasks, ensuring the model can handle both types of tasks efficiently.
- Incorporate GPT-4 zero-shot as a baseline: utilize GPT-4 in a zero-shot setting to establish a performance baseline, enabling a robust comparison with the fine-tuned GPT-2 model and highlighting the effectiveness of task-specific fine-tuning.
- Evaluate and test model efficiency and applicability: rigorously test the model's performance using real-world Twitter data to assess its effectiveness and applicability in real-time RTC monitoring and analysis.
- Provide a dataset of tweets related to traffic crashes: make this dataset available for further model development or to gain insights about traffic crashes using social media.

The rest of this paper is structured as follows: Section 2 provides essential background information on multitask learning, LLMs, prompt engineering, fine-tuning, and the challenges associated with traffic crash detection and analysis. Section 3 presents related work, contextualizing our research within the broader field. Section 4 details the methodology employed, including the design and implementation of the multitask learning framework and the fine-tuning process. Section 5 offers a comprehensive overview of the experimental setup, experiments, and results, providing insights into the effectiveness of our approach. Finally, Section 6 summarizes our key findings and contributions and discusses potential avenues for future research and improvements.

2. Background

This section provides an overview of the key concepts and theoretical foundations underpinning our research.

2.1. Multitask Learning

Multitask learning (MTL) is an approach in machine learning where a model is trained simultaneously on multiple related tasks, allowing it to learn shared representations across these tasks. The primary advantage of MTL lies in its ability to leverage commonalities among tasks, which can lead to improved generalization and performance compared to models trained on each task individually [10]. By learning shared features and representations, MTL can often reduce the risk of overfitting and enhance the model's ability to generalize to new, unseen data [14].

The concept of multitask learning has its roots in the field of neural networks, where it was first applied to tasks such as speech recognition, natural language processing, and computer vision. In these domains, MTL has been shown to improve the performance of models by allowing them to benefit from the auxiliary information provided by related tasks [15]. For instance, in natural language processing, tasks like part-of-speech tagging, named entity recognition, and dependency parsing are often learned together in a multitasking framework, as they share standard linguistic features [16].

MTL is particularly effective in scenarios where there is a limited amount of labeled data for individual tasks. Combining data from related tasks allows the model to use the available information better, leading to more robust predictions [17]. Moreover, MTL can help mitigate data imbalance, where some tasks have significantly more data than others. In such cases, the shared learning process can prevent the model from being biased towards the more dominant tasks [18].

Multitask learning has been shown to improve the performance of models such as BERT and GPT by enabling them to leverage shared features across tasks like text classification, sentiment analysis, and named entity recognition [10,14]. MTL employs the advanced capabilities of the GPT-2 model, renowned for generating coherent human-like text and deciphering complex language patterns [11]. This allows the model to perform well in various natural language processing tasks by leveraging shared features across multiple related tasks, enhancing overall performance and generalization. By fine-tuning this model on a curated dataset of RTC-related tweets, we achieve dual functionality: precise classification of RTC features such as collision type and severity and extracting contextual details often overlooked by conventional models [3,19].

The application of MTL in conjunction with large language models (LLMs) like GPT-2 and GPT-4 further enhances its effectiveness. These models are capable of learning complex language patterns and representations, which can be shared across tasks in a multitasking setting, thereby improving the overall performance of the system [20].

2.2. Large Language Models (LLMs)

Large Language Models (LLMs) are advanced neural networks designed to generate, comprehend, and interpret human language. These models are distinguished by their vast number of parameters, deep architectures, and extensive training on large datasets. LLMs excel at capturing the complexities and subtleties of language, making them highly useful for a broad range of applications such as text generation, sentiment analysis, and question-answering systems [21]. A pivotal advancement in the development of LLMs was the introduction of the transformer architecture by Vaswani et al., which showcased the effectiveness of self-attention mechanisms, significantly improving performance across various natural language processing (NLP) tasks like language translation, sentiment analysis, and text generation [22].

A notable feature of LLMs is their ability to learn contextual representations of words and phrases. Unlike earlier models that used static word embeddings [23], LLMs employ dynamic embeddings, providing a more nuanced understanding of language across dif-

ferent contexts. This innovation led to the creation of landmark models such as OpenAI's Generative Pretrained Transformer (GPT), which demonstrated the potential of unsupervised learning for language comprehension and generation [9]. Subsequent models, including GPT-2 and GPT-3, expanded on these foundations, further increasing the scale and capabilities of transformer architectures [8]. Following the advancements of GPT-3, models like ChatGPT have evolved significantly, being fine-tuned specifically for conversational tasks, which illustrates the shift from general language understanding to more specialized, context-aware applications [24]. This trend highlights the growing dependence on LLMs for the development of systems and applications, reducing the need for separate machine learning models.

2.3. Prompt Engineering

Prompt engineering is a technique employed to leverage the capabilities of LLMs for specific tasks by crafting precise prompts or instructions. This method directs LLMs to generate desired outputs or perform specific tasks effectively. Various prompt engineering strategies include zero-shot, few-shot, role-playing, and chain-of-thought prompting. For instance, zero-shot prompting involves structuring a task so that the LLM can understand and generate a suitable response without prior task-specific training [25]. In contrast, few-shot prompting provides the model with a few examples to guide its responses, which is particularly effective for controlling output format but may be less suitable for tasks requiring complex reasoning [26]. Role-playing prompts involve guiding the LLM to simulate a particular role or persona, thereby enhancing its ability to produce contextually relevant responses [6]. Chain-of-thought prompting encourages the model to provide a step-by-step rationale before arriving at a final response, aiding the model in making more informed decisions by understanding the underlying reasoning behind specific actions [27]. These strategies are critical in understanding the broader applications of LLMs; however, in our study, we specifically utilized zero-shot prompting as a baseline model for multitask learning in traffic crash analysis.

2.4. Fine-Tuning LLMs

Fine-tuning is a crucial process that adapts pretrained LLMs to specialized tasks by further training them on domain-specific datasets, thereby improving their performance in targeted applications. This process customizes the general language capabilities of LLMs to excel in specific domains, such as traffic crash analysis. Typically, fine-tuning starts with a pretrained LLM like GPT-2, which has already learned a broad spectrum of language patterns and semantics from large text corpora. The model is then fine-tuned on a smaller, domain-specific dataset, transferring its general language knowledge to the specialized task [28]. This method enables LLMs to become highly effective in specific tasks while retaining their overall language comprehension. In this study, the fine-tuning of GPT-2 is concentrated on performing both classification and information retrieval tasks related to Road Traffic Crashes (RTCs), ensuring the model is adept at managing the complexities of this domain. The detailed methodology of fine-tuning is elaborated in the subsequent section.

2.5. Traffic Crash Detection and Analysis

The application of large language models (LLMs) in traffic crash detection and analysis represents a significant advancement in intelligent transportation systems. Traditionally, traffic crash detection has relied on physical sensors, camera footage, and manual reporting, which often suffer from latency and coverage limitations. The emergence of social media as a data source, coupled with the capabilities of LLMs, offers a powerful alternative for real-time traffic incident detection and analysis [6].

LLMs, such as GPT-3.5 and GPT-4, have demonstrated their ability to process and analyze vast amounts of unstructured text data, making them ideal for extracting meaningful insights from social media platforms like Twitter [9]. By employing advanced natural

language processing techniques, these models can identify relevant patterns in tweets related to road traffic crashes (RTCs), such as location, severity, and cause. For instance, a fine-tuned LLM can classify tweets to determine whether they report an accident, predict its severity, and extract key details such as the number of vehicles involved or the presence of injuries [8].

Moreover, multitask learning frameworks have enhanced the performance of LLMs in traffic crash detection by allowing the models to tackle multiple related tasks simultaneously. This approach improves the analysis's efficiency and accuracy, as the shared representations learned across tasks lead to better generalization [24]. For example, a multitask LLM might be trained to detect crashes, classify the type of incident, and retrieve detailed contextual information, such as the specific location and time of the crash. This simultaneous processing of multiple tasks speeds up the detection process and provides a richer understanding of the incident, which is crucial for timely and effective response measures [25].

Integrating LLMs with real-time social media monitoring systems offers significant benefits for traffic management and safety. By continuously analyzing social media data, these systems can quickly detect emerging traffic incidents, allowing authorities to deploy emergency services or issue public alerts more swiftly than traditional methods. Studies have shown that such systems can reduce response times and improve situational awareness, ultimately leading to enhanced road safety and reduced accident severity [5].

3. Literature Review

The utilization of social media for traffic crash detection and analysis represents a significant shift in real-time data gathering and processing in the field of road safety. Social media platforms provide a continuous stream of real-time, user-generated content that can offer immediate information about road incidents, which is invaluable for emergency response and traffic management [27]. Researchers have increasingly harnessed these data sources to detect and analyze road traffic incidents more effectively, using advanced NLP and machine learning techniques to sift through vast amounts of unstructured data.

3.1. Social Media as a Data Source for Traffic Analysis

Recent studies have highlighted the potential of social media data in various traffic-related applications. For instance, social media has been effectively used to identify activity patterns [29], monitor special traffic events [30], predict traffic flow [31,32], manage transport information [31], detect travel modes [33], and analyze destination or route choices [8]. Social media data, reflecting real-life user situations, often avoids the biases typical in traditional data collection methods [9]. Twitter's timeliness and location accuracy have been validated in accident detection studies leveraging GPS-enabled smartphones [34] and travel behavior studies validated by household travel surveys [35].

Incident detection studies have utilized tweets to deliver precise research outcomes. For example, microblogs have been used for small-scale incident detection [36], while D'Andrea et al. compared different regression models for detecting traffic incidents from Twitter streams [37]. Additionally, research has explored the relationship between tweets and traffic jams [38], and other studies have used the Naïve Bayes classification with multiple data sources to identify major incident types [39]. However, challenges like the limited volume of accident-related tweets compared to widely publicized events [32,40] and their confinement to specific areas and time frames [36] persist. The unstructured nature of tweet data and language ambiguity further complicate the analysis [41]. While support vector machines (SVM) [36,37] and natural language processing (NLP) techniques [42] have been employed, keyword detection alone often falls short due to the concise nature of tweets. Social media platforms like Twitter serve as valuable auxiliary sources for traffic safety analysis, providing real-time insights into incidents that may not be captured by traditional reporting methods [43].

3.2. Advances in NLP and Large Language Models (LLMs)

Deep learning models have shown considerable promise in addressing these challenges. Architectures such as deep belief networks (DBN), recurrent neural networks (RNN), and long short-term memory (LSTM) networks have outperformed traditional neural networks in traffic flow prediction [3]. RNNs and LSTMs, particularly effective in processing sequential data, have proven their worth in applications like speech and handwriting recognition [44,45]. These models' ability to capture long-term dependencies makes them suitable for classifying accident-related tweets, often containing sequential data.

NLP has become an indispensable tool in extracting meaningful information from the vast and varied data generated on social media platforms. The deployment of LLMs such as GPT-2, GPT-3.5, and recently GPT-4 in social media data analysis offers a sophisticated approach to understanding complex language patterns and generating human-like text responses. GPT-4, in particular, has been applied in zero-shot and few-shot settings to analyze traffic-related tweets, demonstrating its ability to handle a wide variety of tasks with minimal or no task-specific training [46,47]. These models are particularly effective in handling the informal, often unstructured text found in social media posts, which can vary significantly from formal written language [48,49]. LLMs, trained on diverse datasets comprising extensive textual content, are capable of grasping and generating nuanced language characteristic of social media communication. The application of these models in social media analysis has revolutionized how data are processed, moving beyond simple keyword searches to more complex semantic understanding and sentiment analysis [50]. This capability allows researchers to extract a richer, more detailed set of data from social media, which is critical in areas such as public sentiment analysis, market trends, and emergency situations like road traffic incidents.

To further illustrate the application of social media in traffic analysis, Table 1 summarizes recent studies that have utilized various methodologies to harness social media data for traffic incident detection and analysis. These studies highlight the evolving role of social media as a critical data source in real-time traffic management, demonstrating the effectiveness of different approaches in processing and analyzing the vast amounts of unstructured data generated on these platforms.

Table 1. Most recent studies on deep learning, multitask learning, and large language models (LLMs) for social media analysis and traffic data processing.

Study	Authors (Year)	Domain	Model Used	Data Type	Key Findings
Exploring the Potential of Social Media Data in Interpreting Traffic Congestion: A Case Study of Jiangsu Freeways [40]	[40]	Traffic Congestion Analysis	Document Frequency-Based Method	Sina Weibo Microblogs	Identified congestion-prone areas using Sina Weibo data, demonstrating the potential for traffic analysis through social media.
Social Media-Based Traffic Situational Awareness under Extreme Weather [41]	[41]	Traffic Situational Awareness	LSTM Classifier	Weibo Data	Enhanced traffic situational awareness under extreme weather with 93.8%–95.8% accuracy using an LSTM classifier.
Traffic Event Detection as a Slot Filling Problem [42]	[42]	Traffic Event Detection	CNN, LSTM, BERT	Twitter Data	Addressed traffic event detection from Twitter data as a text classification and slot filling problem, achieving high performance scores.

Table 1. *Cont.*

Study	Authors (Year)	Domain	Model Used	Data Type	Key Findings
Identification and Classification of Road Traffic Incidents in Panama City through Social Media Stream Analysis [43]	[43]	Traffic Incident Identification	SVM, Naïve Bayes, Random Forest, XGBoost	Twitter Data	Achieved high precision rates in traffic incident identification and classification using machine learning models.
Deep Learning Ensemble Model for the Prediction of Traffic Accidents Using Social Media Data [3]	[3]	Traffic Accident Prediction	GRU, CNN	Social Media Data, Bogota Climate Information	Proposed a deep learning ensemble model for traffic accident prediction, outperforming baseline algorithms.
Twitter-informed Prediction for Urban Traffic Flow Using Machine Learning [44]	[44]	Urban Traffic Flow Prediction	Random Forest, Gradient Boosting	Twitter Data, PeMS Data	Combined Twitter data with traffic and weather information to enhance traffic flow prediction accuracy.
Leveraging Large Language Models to Detect Influence Campaigns on Social Media [45]	[45]	Influence Campaign Detection	Large Language Models (LLMs)	Multilingual Social Media Datasets	Showcased superior performance in detecting and adapting to influence campaigns using LLMs incorporating user metadata and network structures.
MentaLLaMA: Interpretable Mental Health Analysis on Social Media with Large Language Models [46]	[46]	Mental Health Analysis	MentaLLaMA, LLMs	IMHI dataset, Social Media Data	Introduced MentalLLaMA, achieving state-of-the-art correctness in mental health analysis on social media, generating high-quality explanations.
Accuracy of a Large Language Model in Distinguishing Anti- And Pro-vaccination Messages on Social Media: The Case of Human Papillomavirus Vaccination [47]	[47]	Sentiment Analysis	ChatGPT, LLMs	Facebook, Twitter Data	Assessed ChatGPT's accuracy in sentiment analysis of pro- and anti-vaccination messages on social media.
Advancing Annotation of Stance in Social Media Posts: A Comparative Analysis of Large Language Models and Crowd Sourcing [48]	[48]	Stance Annotation	LLMs, Crowdsourcing	Twitter Data	The performance of LLMs with human annotators in stance annotation was compared, and it was found that LLMs perform well when human annotators do.
Multitask Prompt Words Learning for Social Media Content Generation [49]	[49]	Content Generation	Multi-modal Information Fusion	Social Media Data	Introduced a multitask prompt word learning framework, improving the quality and relevance of social media content generation.
Enhancing Mental Health Condition Detection on Social Media through Multitask Learning [50]	[50]	Mental Health Detection	BERT, Multitask Learning	Reddit, SWMH, PsySym	Multitask learning was used to enhance mental health condition detection, outperforming single-task and large language models.

Table 1. *Cont.*

Study	Authors (Year)	Domain	Model Used	Data Type	Key Findings
Multitask Learning for Recognizing Stress and Depression in Social Media [51]	[51]	Stress and Depression Detection	BERT, Attention Fusion	Reddit, Stress, and Depression Datasets	Introduced multitask learning frameworks for recognizing stress and depression, outperforming existing methods.
Multitask Learning for Personal Health Mention Detection on Social Media [52]	[52]ss	Health Mention Detection	Multitask Learning	Annotated Social Media Data	Enhanced personal health mention detection by incorporating emotional information as an auxiliary task in a multitask learning framework.

The following section delves deeper into how our study aims to fill this gap by leveraging Multi-Task Learning and Large Language Models for enhanced traffic analysis using social media data.

3.3. Addressing the Research Gap: Multitask Learning with LLMs

While significant advancements have been made in various fields using multitask learning (MTL) and large language models (LLMs), their application in road traffic crash analysis is still underexplored. The key gaps that our study addresses, with a direct contribution to road safety, are as follows:

Lack of MTL Application in Traffic Analysis:

MTL has the potential to enhance model performance by leveraging shared representations across related tasks, though its application in road traffic crash analysis remains limited and underexplored. By applying MTL, we can simultaneously analyze multiple aspects of traffic incidents, leading to more comprehensive and accurate insights that improve incident response and prevention strategies.

Underutilization of Social Media for Real-Time Traffic Safety Insights:

While social media data has been used in traffic safety research, integrating advanced LLMs with MTL for systematic data extraction and analysis remains unexplored. Leveraging social media as a real-time data source enables quicker incident detection and analysis, leading to faster response times and better-informed traffic safety decisions.

4. Methodology

This section outlines the methodologies used to retrieve, process, analyze, and classify social media data related to traffic crashes. Figure 1 illustrates the proposed system architecture. Traffic incidents are often reported on social media platforms like Twitter. This real-time data, however, is unstructured and filled with idioms and dynamic language, making it difficult to analyze using traditional methods. Our methodological framework comprises several modules: data crawling, preprocessing, labeling and verification, model fine-tuning, and evaluation. The primary goal of the proposed framework is to automate the detection and analysis of traffic accidents and events through a multitasking learning approach.

Initially, real-time data are gathered from Twitter using targeted hashtags and keywords (Step 1: Data Collection). Following data collection, preprocessing techniques are employed to clean and prepare the data for subsequent analysis (Step 2: Data Preprocessing). Then, data labeling is performed using GPT-3.5 to extract relevant features from each tweet (Step 3: Data Labeling). Domain experts then verify the labeled data to ensure accuracy (Step 4: Manual Verification and Post-Processing).

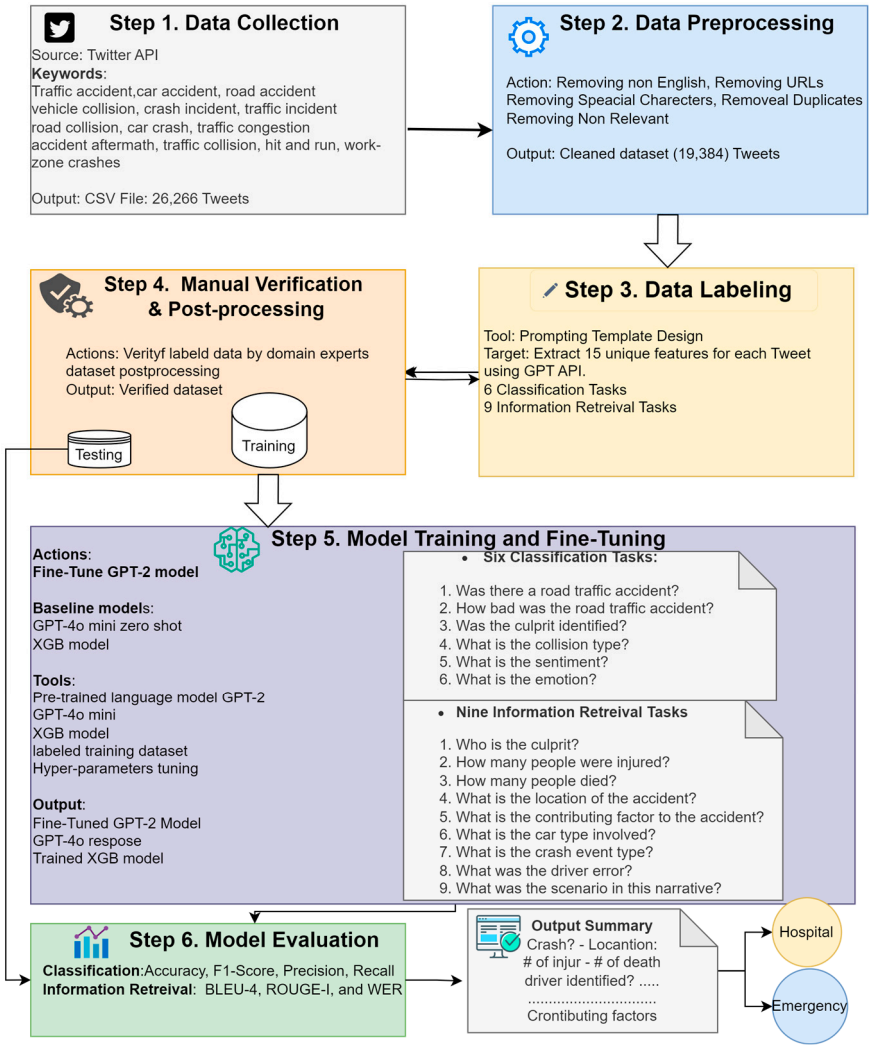

Figure 1. Proposed methodology flowchart.

Next, the GPT-2 model is fine-tuned on the labeled dataset to support multitasking learning objectives, including classification and information retrieval tasks (Step 5: Model Fine-Tuning). Two baseline models were also employed to provide a comprehensive evaluation: GPT-4o mini in a zero-shot setting and XGBoost trained individually on each task. GPT-4o mini was included to compare the effectiveness of task-specific fine-tuning versus zero-shot learning in traffic crash analysis. At the same time, XGBoost served as a representative machine learning model trained on single tasks, providing insight into how traditional ML approaches perform in this context.

Finally, the performance of all models—GPT-2, GPT-4o mini, and XGBoost—was evaluated using various metrics to ensure their effectiveness in real-world applications (Step 6: Model Evaluation). This evaluation allowed us to assess the advantages and limitations of each approach in the context of traffic crash analysis.

4.1. Data Collection

Twitter data were collected from the Australian Twittersphere, a longitudinal database of tweets from accounts identified as Australian. These accounts are characterized by a connection to Australia, as indicated in the free text fields of their profiles, such as location or bio descriptions [53]. Although tweets can include geotagging, this feature is optional and rarely used, with less than 2% of all tweets in the Australian Twittersphere containing location metadata [53]. Due to this lack of reliable geolocation data, the location of a tweet can only be inferred approximately from its content. The dataset is maintained

and continuously updated by the Digital Observatory, ensuring that the selection criteria accurately reflect Australian social media activity.

To ensure the relevance of the data, three distinct groups of tweets were extracted based on tailored search criteria designed to capture various aspects of Road Traffic Crashes (RTC). All tweets included in this study were published between 1 May 2021 and 31 May 2023. The data extraction process was carefully managed to focus exclusively on content relevant to the research objectives, drawing from a comprehensive dataset that represents a significant portion of Australian Twitter activity during the specified period.

Group 1: tweets including terms such as hit and run, hit-and-run, hit n run, leave (or leaving) the scene, fleeing the scene, accident flee, car crash flee, pedestrian hit and run, cyclist hit and run, and driver fled.

Group 2: tweets mentioning work zone crash, construction zone accident, road work collision, work zone safety, work zone incident, construction site crash, and road work accident.

Group 3: tweets containing traffic accident, car accident, road accident, vehicle collision, crash incident, traffic incident, road collision, car crash, traffic congestion, accident aftermath, and traffic collision.

As outlined in the methodology flowchart (Figure 1), a total of 26,226 tweets were filtered and utilized, each meticulously curated to match our research criteria. Table 2 shows the attributes of Twitter data collected for analysis. Our focus was on the text of the tweets, which was used as the primary source for annotating the dataset.

Table 2. Attributes and definitions of Twitter (X) data collected for analysis.

Attribute	Definition
tweet_id	Unique identifier of a tweet, as provided by Twitter/X.
username	Name of the Twitter account that posted the tweet.
text	Full body text of the tweet, including all hashtags.
created_at	Time the tweet was posted.
in_reply_to_tweet_id	If the tweet is a reply, this shows the ID of the tweet this replied to; otherwise, the value is blank.
retweeted_tweet_id	If the tweet is a retweet, this shows the retweeted Tweet's ID; otherwise, the value is blank.
quoted_tweet_id	If the tweet is a quote, this shows the quoted tweet's ID; otherwise, the value is blank.
favorite_count	The number of favorites this tweet received at the time of collection.
location	Location that the tweet posted; can be blank.

The data collection process, represented in Step 1 of Figure 1, also involved specific keywords and hashtags to ensure the relevancy of the tweets to traffic incidents. These included keywords such as traffic accident, car accident, road accident, vehicle collision, crash incident, traffic incident, road collision, car crash, traffic congestion, accident aftermath, and traffic collision. Hashtags used included #auspol, #7news, #9news, #springst, #covid19aus, #tlalert, #breaking, #vicvotes, #thisisnotjournalism, #keenananderson, #wearamask, #covid19, #covid, #australia, #victraffic, #abc730, #covidisnotover, #andrewsymonds, #neighbours, and #robodebtrc.

4.2. Data Preprocessing

Our data preprocessing approach prioritizes the maintenance of contextual integrity, which is essential for effective utilization of contextual word embeddings. This importance is emphasized in studies on word representation models [54,55], which highlight the significance of preserving context to improve model performance and representation accuracy.

To begin the preprocessing, we employed regular expressions to systematically remove elements that could introduce noise or irrelevant information, such as usernames (@username), hashtags (#hashtag), emojis, smileys, non-alphanumeric characters, and URLs. While part of the original tweet content, these elements do not contribute meaningfully to the analysis of traffic incidents and could skew the embedding representations if left unchecked.

Recognizing the global nature of Twitter data, we identified and removed tweets containing non-English sentences to maintain consistency in language processing, ensuring that all data passed to our models were in English. This step was critical because our language models were trained primarily on English-language data, and non-English content could adversely affect model performance.

Next, we addressed the issue of redundancy by filtering out duplicate tweets. Duplicate content can distort frequency-based analyses and affect the model's ability to generalize from the data, so this step was crucial in refining the dataset to 19,384 unique tweets. This streamlined collection was deemed ready for further analysis, guaranteeing a higher-quality dataset tailored for our research.

Our preprocessing did not stop at basic cleaning. Given that the data collection was driven by keywords and hashtags directly related to traffic incidents, such as "traffic accident", "car accident", and "road collision", it was imperative to ensure that these tweets were genuinely relevant to our study. Our fine-tuned model (GPT-2) conducted an initial classification step to enhance data quality further. This step involved verifying whether each tweet was indeed related to a traffic crash, effectively filtering out non-crash-related tweets. This additional verification step is vital for eliminating false positives—tweets that may contain relevant keywords but do not actually pertain to traffic incidents.

This layered approach ensures that our framework robustly identifies and processes relevant information, significantly improving the accuracy and applicability of our traffic event detection and analysis system.

4.3. Data Labeling

Accurate class labeling of social media data is a critical step in enhancing the performance of traffic event detection and sentiment analysis. Traditional methods heavily relied on manual labeling and conventional machine learning classifiers [56]. However, these approaches often resulted in low accuracy due to the inherent complexities and variability of social media data, which include informal language and diverse content structures. Recent advancements in deep learning, particularly the use of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), have offered more robust solutions. For example, Gutierrez-Osorio et al. [3] applied CNNs and RNNs to classify traffic event data into categories like non-traffic, traffic incidents, and traffic information. Despite these advancements, challenges remained, particularly with the brief and unstructured nature of Twitter data. Similarly, the work of D'Andrea et al. [37] and Chen et al. [57] leveraged Natural Language Processing (NLP) techniques and word embedding models such as word2vec, but further progress is needed to fully capture the nuances of social media content.

To address these challenges and streamline the data labeling process, we implemented GPT-3.5, a powerful large language model. The use of GPT-3.5 significantly reduced the reliance on manual labeling by automating the feature extraction process, thereby expediting the creation of a comprehensive and high-quality dataset for training machine learning models. This automated approach aligns with the findings of Pei et al. [9], who demonstrated that GPT-based models excel in generating and recovering data through self-supervision, thus improving the accuracy and efficiency of annotation tasks. Additionally, the work by Xu et al. [58] supports the effectiveness of GPT-3 and GPT-4 models in handling complex, multi-feature data extraction tasks across multiple languages, further validating the choice of GPT-3.5 for this study.

In our approach, each of the 19,834 tweets in our dataset was processed using GPT-3.5, with a specifically crafted prompt designed to extract 15 predefined features critical for a comprehensive understanding of each traffic incident. These features include crucial aspects such as the presence of a road accident, its severity, and the number of casualties involved, among others. The prompt engineering process was iterative, involving continuous refinement and testing to ensure the accuracy and relevance of the extracted information. Table 3 shows the GPT-3.5 prompt template, Algorithm 1 presents the dataset curation process pseudocode, and Figure 2 illustrates the Workflow for Processing and Analyzing Tweets Using GPT-3.5, together providing a detailed overview of how the labeling process was automated.

Table 3. GPT 3.5 prompt template and dataset curation process pseudocode.

```
"""
Given this tweet below:
<tweet>
{text}
</tweet>
I want you to extract information for the following fields:
- Road Accident: Yes or No
- Severity: mild, moderate, fatal
- Driver:
- Driver apprehended:
- No of injured:
- No of deaths:
- Location of accident:
- Contributing factor:
- type of car involved:
- crash event type:
- driver error:
- collision type:
- case scenario:
- sentiment: (positive, negative, or neutral)
- emotions: (pick only one of these: fear, anger, sadness, happy, neutral, disgust, love, confusion,
gratitude, sympathy or empathy)
Provide your answer in a JSON dictionary.
RULES
- You must provide a value to each field above.
- if the tweet does not contain enough information to answer any of the above field, you must set
that field to the value "None."
Your response:"""
```

Through this automated labeling process, we generated a substantial labeled dataset consisting of 297,510 samples (19,834 tweets multiplied by 15 features per tweet). The extracted features were saved in a structured JSON format for further processing, ensuring they were suitable for classification and information retrieval tasks during model fine-tuning. To manage large volumes of data efficiently and to conform to API rate limits, we structured our data processing workflow to handle tweets in batches of 128, incorporating sleep intervals of 120 s between batches.

This systematic labeling process was integral to Step 3 of our methodology (as shown in Figure 1), where the GPT-3.5 model was employed to perform detailed feature extraction across six classification tasks and nine information retrieval tasks. These tasks were designed to provide a nuanced understanding of the data, ranging from identifying the sentiment expressed in the tweets to determining the type of collision and the contributing factors involved.

Algorithm 1: Dataset Curation Process pseudocode

```
1: procedure DATASET_CURATION()
2:        D ← LOAD_RAW_DATA()
3:        D ← REMOVE(#, duplicates, emojis, etc., from D)
4:        auth ← AUTHENTICATE(API_KEY)
5:        P ← CREATE_PROMPTS(D)
6:       F ← DEFINE_FEATURES()
7:       M_config ← CONFIGURE_MODEL(GPT_3_5, F)
8:       M ← Configure_MODEL(M_config)
9:       PD ← BATCH_PROCESS(M, D)
10:       do
11:            SLEEP(120)
12:            PD ← CONTINUE_PROCESS(M)
13:       loop until ALL_PROCESSED(P.D.)
14:       if NEED_ITERATION() then
15:            go to step 3
16:       endif
17:       J ← TO_JSON(PD)
18:       C ← TO_CSV(J)
19:       FD ← FINAL_CLEAN(C)
20:       SD ← STANDARDIZE(FD)
21:       if ¬VALIDATE(SD) then
22:            raise ERROR("Validation Failed")
23:       endif
24:       return SD
25: end
DATASET_CURATION
```

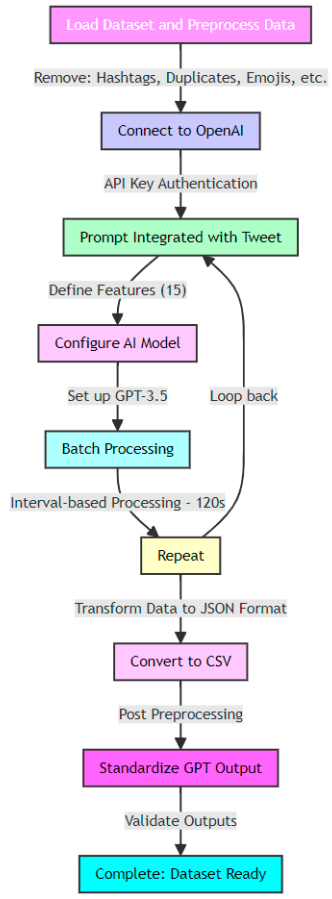

Figure 2. Workflow for Processing and Analyzing Tweets Using GPT-3.5.

4.4. Manual Verification and Post-Processing

Following the automated data labeling process, the dataset underwent a comprehensive post-processing phase aimed at standardizing and cleansing the output values extracted by GPT-3.5. This step was essential to ensure the reliability and accuracy of the data before they were used for model fine-tuning and evaluation. The primary objectives of this phase were to resolve inconsistencies, standardize the data outputs, and address any errors or anomalies detected during the initial labeling. Standardization was crucial for producing a dataset that is not only consistent but also aligned with the structured requirements of machine learning models.

To ensure the high quality of the synthesized dataset, a subset of the data was manually reviewed. This manual verification process involved cross-checking the labels generated by GPT-3.5 with the original tweet content to confirm their accuracy and relevance. This step was critical in identifying and correcting any misclassifications or inaccuracies that might have occurred during the automated labeling process.

The post-processing and expert review involved several key activities:

Standardization: This included reconciling different spellings and formats, such as standardizing variations of terms like "hit-and-run" to a consistent format. For example, Table 4 provides illustrative examples of how various GPT-generated labels were standardized across different features, ensuring consistency and clarity in the dataset.

Table 4. Illustrative examples of standardization.

Feature	GPT Label	Standardized Value
No of Deaths	"1 (pregnant mate)"	"1"
	"at least 12"	"≥12"
	"dependent on scenario"	"unknown"
	"one million Americans"	"unknown"
Driver Apprehended	"one male detained, second tracked by pd falco"	"yes"
	"varies"	"unknown"
	"no information"	"unknown"
	"not applicable"	"no"
Crash Event Type	"hit my parked car"	"single-vehicle crashes"
	"reckless joyride and crashing into a house"	"vehicle crashing into a building"
	"road accidents"	"traffic accidents"
	"roll-over"	"vehicle rollover"
Collision Type	"animal-vehicle"	"single-vehicle crashes"
	"back collision"	"backing collision"
	"broadside collision"	"broadside collision"
	"hit and run accidents"	"hit and run accidents"
Severity	"catastrophic"	"fatal"
	"mild to moderate"	"moderate"
	"serious non-life threatening"	"moderate"
	"tragic"	"fatal"
Driver Error	"abandoning critically injured passenger"	"failure to render aid"
	"attempted hit-and-run"	"hit-and-run attempt"
	"confusion"	"not applicable"

Table 4. *Cont.*

Feature	GPT Label	Standardized Value
	"dangerous driving, impaired, impaired 80+"	"impaired driving"
Sentiment	"confusion"	"negative"
	"shocked"	"negative"
	"unknown"	"neutral"
Emotions	"amusement"	"happy"
	"annoyed"	"angry"
	"anxious"	"fear"
Contributing Factor	"alleged shooting"	"not applicable"
	"allegedly caused a sickening three-car crash"	"possible contributing factor"
	"bad interview"	"not applicable"
	"banana peel thrown onto road"	"possible contributing factor"
Type of Car Involved	"conservation police officer's vehicle and another vehicle"	"police officer's vehicle, another vehicle"
	"grey, 2 door car with dark tinted windows, loud exhaust, and black hardtop"	"grey, 2 door car with dark tinted window and black hardtop"
	"unidentified"	"unidentified"
No of Injured	"1 (deputy)"	"1"
	"at least 5"	"≥5"
	"dozens"	"12+"
	"none"	"0"

Verification: A manual review of a subset of the data was conducted to ensure that the automated labels were accurate and reliable. This was particularly important for ensuring that the model correctly interpreted complex or ambiguous tweet content.

Cleaning: This involved removing redundant or nonsensical label values and ensuring consistency across all labels. The cleaning process was essential to ensure that the dataset was free from errors that could compromise the integrity of subsequent analyses.

Finalization: The data were prepared for subsequent training and evaluation phases, ensuring it met the necessary quality standards.

Given the diverse nature of the tweets and the complex data extracted, post-processing was essential to reconcile discrepancies and standardize values across various features. For instance, as shown in Table 4, the standardization process addressed variations in how features like the number of deaths, driver apprehension, and collision types were labeled by GPT-3.5, ensuring consistency across the dataset.

A key step in our methodology involved the initial filtering of tweets to ensure relevance to road traffic crashes (RTCs). Keywords related to traffic crashes were used for initial filtering through the Twitter API, focusing on terms such as "hit and run", "road accident", and "traffic collision." However, recognizing the limitations of keyword-based filtering, we further refined the dataset using a fine-tuned GPT-2 model to perform an additional classification task. This task involved verifying whether a tweet was genuinely related to a traffic crash before proceeding to a detailed analysis.

This additional step was crucial in enhancing the accuracy of the dataset by filtering out non-crash-related content that may have been erroneously captured by the API filters. Table 5 presents examples where the API filters misclassified tweets as crash-related, but the

GPT-2 model correctly identified them as non-crash-related. These examples demonstrate the effectiveness of the fine-tuned GPT-2 model in refining the dataset and ensuring that only relevant tweets were included for further analysis.

Table 5. Examples of API vs. fine-tuned GPT-2 model classification.

Serial Number	Tweet	API Filter Classification	GPT-2 Model Classification	Context
1	R.T. @9NewsGoldCoast: The Maroons will travel from the Gold Coast to Western Australia today aiming to execute a hit-and-run mission in the west to regain the Origin Shield. #9News https://t.co/qfEJem4RJO (accessed on 7 November 2023)	Crash-related	Non-crash-related	The tweet uses "hit-and-run" in the context of a sports mission, not a traffic crash.
2	If she's still hung over some guy whether for good or for bad my guy flee the scene. No try fix wetin you no spoil (accessed on 7 November 2023)	Crash-related	Non-crash-related	The phrase "flee the scene" is used in a relationship context, not related to a traffic incident.
3	Monday—Mixed 4's: Slap That Ace beat Hit n Run (23–15) https://t.co/n0PAcPwi5f (accessed on 7 November 2023)	Crash-related	Non-crash-related	The term "Hit n Run" is part of a sports score update, unrelated to any traffic crash.
4	The best interview I have seen Laura do, and I'll admit I'm not always a fan. A car crash of an interview for the PM. Gave me absolutely no faith in sorting out the country's issues. @bbclaurak (accessed on 7 November 2023)	Crash-related	Non-crash-related	"Car crash" is used metaphorically to describe a disastrous interview, not an actual traffic incident.

This hybrid approach, combining automated standardization with manual verification, significantly improved the quality of the dataset. This meticulous process ensured that the data were consistent and reliable and prepared it for effective use in training and evaluating machine learning models, as illustrated in Step 4 of Figure 1.

The detailed post-processing approach highlighted the complexity and extensive effort required to manage and prepare the data for subsequent modeling tasks, ensuring that the final dataset was robust and applicable to real-world scenarios.

4.5. Final Curated and Annotated Dataset Description

The resulting dataset consists of 297,510 samples derived from the 19,834 tweets, with 15 features extracted from each tweet. This dataset was divided into training and testing sets to evaluate the model's performance effectively. Specifically, 287,010 samples were allocated for training, while 10,500 samples were reserved for testing. This allocation was carefully designed to meet the requirements for each of the 15 targeted outputs, ensuring a robust and balanced dataset for model evaluation. Table 6 below provides an overview of the data distribution and statistics, detailing the number of samples, features, and their allocation between training and testing sets. Appendix A provides a few samples of the labeled dataset and the extracted values from RTC tweets.

Table 6. Dataset statistics.

Statistic	Value
Number of samples	26,226
Number of samples after preprocessing	19,834
No. of features generated for each tweet	15
Total number of samples	19,384 × 15 = 297,510
Training samples	287,010
Testing samples	10,500

To provide further clarity on the dataset's structure, Table 7 shows the distribution of data labels across the six classification tasks. This table illustrates how the data are segmented between different categories for both the training and testing sets, highlighting the distribution of labels that are critical for accurate model training and evaluation.

Table 7. Dataset split for the six classification tasks.

Was There a Road Traffic Accident?								
Dataset		Yes		No		Unknown		Total
Training Set		16,702		2431		1		19,134
Testing Set		605		95		0		700
Total		17,307		2526		1		19,834
Severity: How bad was the road traffic accident?								
Dataset	Fatal	Mild	Moderate	None	Severe	Critical	Unknown	Total
Training Set	6824	6736	2302	2269	619	287	97	19,134
Testing Set	277	228	83	86	13	9	4	700
Total	7101	6964	2385	2355	632	296	101	19,834
Was the culprit driver identified?								
Dataset	Yes	No	Unknown	Reportedly	Some		Total	
Training Set	2339	15,682	1106	6	1		19,134	
Testing Set	102	559	39	0	0		700	
Total	2441	16,241	1145	6	1		19,834	
What is the collision type?								
Dataset			Training Set		Testing Set		Total	
Hit-And-Run			3049		110		3159	
Pedestrian			1486		46		1532	
Broadside			390		16		406	
Single-Vehicle Crashes			179		6		185	
Stationary Object			133		9		142	
Chain Reaction			127		6		133	
Rear-End			101		0		101	
Type of Car			103		4		107	
Rollover			32		0		32	
Intersection			10		0		10	
Sideswipe			3		0		3	
Side-Impact			2		0		2	
Backing			2		0		2	
Not Applicable			13,517		503		14,020	
Total			19,134		700		19,834	
What is the sentiment?								
Dataset	Negative		Positive		Neutral		Total	
Training Set	13,348		2388		3398		19,134	
Testing Set	490		83		127		700	
Total	17,307		2526		1		19,834	

Table 7. *Cont.*

What is the emotion?			
Dataset	Training Set	Testing Set	Total
Sadness	6788	249	7037
Anger	4980	183	5163
Fear	3481	112	3593
Neutral	2187	87	2274
Happy	922	36	958
Disgust	309	12	321
Gratitude	207	11	218
Confusion	202	5	207
Sympathy	30	2	32
Love	24	1	25
Curiosity	3	1	4
Empathy	1	1	2
Total	19,134	700	19,834

4.6. Model Training and Fine-Tuning

The selection of models in this study is driven by a balance of performance, computational efficiency, and domain-specific relevance, ensuring that the chosen models align with the study's objectives and the nature of the data. GPT-2 Fine-Tuning was selected due to its demonstrated capability to provide strong language model performance while maintaining computational efficiency and open source, making it a suitable choice for tasks where resource constraints are considered. Additionally, its success in similar domain-specific tasks, such as analyzing specialized text data, underscores its relevance and applicability in this context [11]. GPT-4 Zero-Shot was chosen for its superior language processing capabilities, enabling it to perform complex tasks without the need for task-specific training, thereby offering a robust baseline for comparison in scenarios where task-specific fine-tuning is not feasible [8]. Finally, XGBoost was selected for its effectiveness in handling structured data, particularly in classification tasks where it has been shown to deliver strong performance across various NLP domains. Its widespread recognition and consistent results in predictive modeling further justify its inclusion as a key component of the study's methodology [57].

4.7. Model Evaluation

In evaluating the performance of models, particularly in natural language processing (NLP) and machine learning tasks, several metrics are commonly employed to measure the accuracy and effectiveness of the models. Accuracy is a fundamental metric used to determine the proportion of correctly predicted instances out of the total instances and is widely applied in classification tasks due to its straightforward interpretation [59]. However, accuracy alone can be misleading, especially in cases of imbalanced datasets. Therefore, precision, recall, and F1-score are often utilized alongside accuracy to provide a more comprehensive assessment. Precision measures the proportion of true positive predictions out of all positive predictions, emphasizing the model's ability to avoid false positives. Recall, on the other hand, calculates the proportion of true positives out of all actual positives, focusing on the model's capacity to identify all relevant instances. The F1-score is the harmonic mean of precision and recall, providing a single metric that balances the trade-off between these two, particularly useful when the cost of false positives and false negatives is significant [59].

For information retrieval and text generation tasks, specific metrics such as ROUGE, BLEU, and Word Error Rate (WER) are widely employed. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) primarily measures the overlap of n-grams between the generated text and reference text, making it effective for summarization tasks [60]. BLEU (Bilingual Evaluation Understudy) is another metric that evaluates the precision of n-grams but is often used for machine translation tasks, measuring how many n-grams in the can-

didate translation are present in the reference translations [61]. Lastly, WER is utilized to assess speech recognition systems by calculating the number of insertions, deletions, and substitutions needed to convert the output text into the reference text, with a lower WER indicating a more accurate transcription [62]. These metrics, collectively, provide a robust framework for evaluating models across various NLP and information retrieval tasks, ensuring that the models not only perform accurately but also meet the specific requirements of the task at hand.

5. Experimental Setup

This section outlines the computing environment, model configurations, and fine-tuning procedures applied to adapt the GPT-2 model for classification and information retrieval tasks related to road traffic accident (RTC) data. Additionally, GPT-4o mini and XGBoost models were employed as baseline models to benchmark the performance of GPT-2.

5.1. Dataset

The dataset used in this study comprises 297,510 samples derived from 19,834 RTC-related tweets, with 15 features extracted from each tweet. The detailed process of converting these tweets into a structured format is described in Sections 4.1–4.5. To ensure effective model evaluation, the dataset was divided into training and testing sets, with 287,010 samples allocated for training and 10,500 samples reserved for testing. This distribution was carefully designed to meet the requirements for each of the 15 targeted outputs. Table 6 provides an overview of the dataset's statistics.

5.2. Computing Environment

Our experiments were conducted using Google Colab, a cloud-based Jupyter notebook environment that provides access to both GPU and CPU resources, leveraging the extensive functionalities offered by the Hugging Face Transformers library. Specifically, we used an NVIDIA Tesla T4 GPU with approximately 12 GB of available memory. This setup was sufficient for fine-tuning the GPT-2 medium model, although careful management of memory resources was necessary. To address memory constraints, especially during the fine-tuning of larger models, we implemented gradient accumulation, which allows effective training with larger batch sizes by accumulating gradients over multiple steps before updating the model's parameters.

While Google Colab also provides access to TPUs, the PyTorch-based implementation of GPT-2 (version 2.4.0), used in this study, did not fully support TPU usage at the time of the experiments. Therefore, GPU-based fine-tuning was chosen.

5.3. Fine Tuning of GPT-2

For our experiments, we selected the GPT-2 Medium model, which contains 355 million parameters. The GPT-2 model comes in several sizes, each with a different number of parameters. The original sizes provided by OpenAI include 124 M, 355 M, 774 M, and 1.5 B parameters. These models vary in their capacity and performance, with larger models typically delivering better results but requiring more computational resources. In our experiments, we utilized the gpt2-medium model, which contains 355 M parameters. This model strikes a balance between computational efficiency and the ability to generate high-quality outputs, making it suitable for our tasks involving road traffic accident-related tweets.

GPT-2 was chosen for its robustness and reliability in domain-specific fine-tuning tasks. First, GPT-2, despite being an earlier model, remains a robust and reliable option for fine-tuning in domain-specific tasks. Its smaller size and lower computational requirements compared to more recent models make it a practical choice for research settings with limited resources, allowing for efficient adaptation to specialized tasks such as those involving RTC-related tweets [11]. While GPT-3.5 and GPT-4 represent more advanced architectures, their computational demands are significantly higher, making GPT-2 a viable option where

resource efficiency is critical [8]. Additionally, GPT-2's architecture has been extensively validated in various NLP applications, demonstrating strong performance when fine-tuned on specific datasets, thus providing a solid foundation for our multitask learning framework [11].

5.3.1. Data Preparation

The dataset used for fine-tuning was derived from tweets related to road traffic accidents (RTC). It comprised 15 distinct tasks, including six classification tasks (e.g., determining if a tweet is related to an RTC) and nine information retrieval (IR) tasks (e.g., extracting the number of injuries from a tweet). The data were formatted as prompt-response pairs, where the prompt posed a question and the response provided the corresponding answer.

In this structure, the prompt represents the question (e.g., "What is the sentiment?"), and the response is the expected output (e.g., "positive"). This prompt-response structure was crucial in preparing the model for both classification and IR tasks.

5.3.2. Tokenization and Input Preparation

The input data were processed using the GPT-2 tokenizer, which is essential for converting the raw text into a format that the model can understand and process. Tokenization involves breaking down the text into individual tokens, which are then converted into numerical representations that the model can work with. The key steps in this process are as follows:

1. Truncation and Padding:
   The input sequences were truncated to a maximum length of 256 tokens. This truncation was necessary to ensure that all sequences fit within the model's input capacity, allowing for efficient processing and reducing computational overhead.
   Padding was applied to shorter sequences to create uniform input lengths across all sequences, ensuring that the model received inputs of consistent size, which is crucial for batch processing during training.
2. Label Preparation for Classification Tasks:
   For classification tasks, labels were assigned to the prompts. The input sequences were tokenized and paired with their corresponding labels. The model was then trained to predict the correct labels based on the input tokens, enabling it to perform tasks such as classifying whether a tweet is related to a road traffic accident (RTC).
3. Handling Long Inputs:
   If an input prompt exceeded the maximum token length of 256 tokens, it was truncated to fit within the model's input capacity. The truncation was performed carefully to preserve the essential parts of the input, ensuring that the context required for accurate predictions was maintained.
4. Input Preparation for Information Retrieval (IR) Tasks:
   In IR tasks, the model was fine-tuned to generate the correct answer based on a provided prompt. The prompt and expected response were tokenized together, with the model learning to predict the sequence of tokens that correspond to the correct answer. This approach allowed the model to handle a wide range of queries related to RTCs, such as extracting the number of injuries or identifying the location of an accident.
5. Tokenization Process:
   The entire process of tokenization and input preparation can be represented by Equation (1). This formula illustrates how the prompt and response are combined and tokenized into a format suitable for GPT-2, ensuring that the input is ready for fine-tuning. Figure 3 shows a flowchart showing the step-by-step process from input tweets and questions to the tokenized format ready for GPT-2 fine-tuning.

input_tokens = tokenizer(prompt+"\n"+response+tokenizer.eos_token, truncation=True, max_length=max_length, return_tensors="pt")["input_ids"].squeeze() (1)

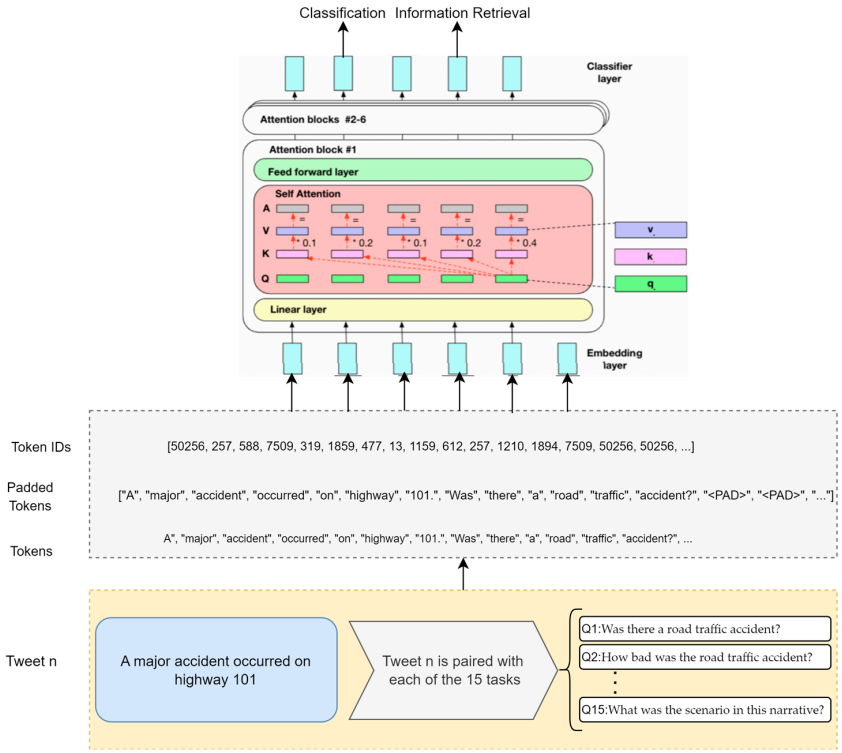

Figure 3. Tokenization and input preparation process.

5.3.3. Model Training Process

The GPT-2 medium model comprises 24 layers, 1024 hidden units, and 16 attention heads. These specifications enable the model to handle complex language tasks while maintaining a reasonable level of computational demand. Initialized with pre-trained weights from a broad internet text dataset, the model provided a strong foundation for language understanding. Fine-tuning was conducted to adapt GPT-2 for both classification and information retrieval (IR) tasks using a specially curated dataset derived from RTC-related tweets. These tasks included six classification tasks (e.g., determining whether a tweet is related to a road traffic accident) and nine IR tasks (e.g., extracting the number of injuries or the accident location). The fine-tuning process focused on optimizing GPT-2's language generation capabilities to accurately classify tweets and extract relevant information, such as crash severity or the number of injuries. This was accomplished by carefully adjusting hyperparameters, including learning rate, batch size, and the number of training epochs, to maximize performance across the specific tasks.

5.3.4. Multitask Learning Objective

In this study, the GPT-2 model was fine-tuned using a multitask learning framework, which allows the model to learn multiple tasks simultaneously by sharing representations across these tasks. This approach not only improves generalization but also optimizes the model's ability to handle varied tasks efficiently.

The objective function for this multitask learning framework is a weighted sum of the individual task loss functions. Specifically, the language modeling loss L_{LM} is calculated as the negative log-likelihood of the predicted token given the preceding tokens (Equation (2)):

$$\mathcal{L}_{LM} = -\sum_{t=1}^{T} \log P(x_t | x_{<t}; \theta), \quad (2)$$

where

- x_t is the token at position t;
- $x_{<t}$ represents the sequence of tokens before position t;
- θ denotes the model parameters;
- T is the total number of tokens in the sequence;
- $P(x_t \mid x_{<t};\theta)$ is the probability of token x_t given the preceding tokens $x_{<t}$ and model parameters θ.

The classification loss (L_{CLS}), typically using cross-entropy, measures the difference between the predicted and actual class distributions (Equation (3)):

$$\mathcal{L}_{CLS} = -\sum_{i=i}^{N} \sum_{c=1}^{C} y_{ic} \log(\hat{y}_{ic}), \quad (3)$$

where

- N is the number of examples in the dataset;
- C is the number of classes in the classification task;
- y_{ic} is the true label, for example, i and class c;
- $\hat{y}_{ic}$ is the predicted probability, for example, i and class c.

The combined loss (L) for the multitask learning framework is a weighted sum of the language modeling and classification losses, allowing the model to balance between generating accurate text and classifying the inputs correctly (Equation (4)):

$$\mathcal{L} = \alpha \mathcal{L}_{LM} + \beta \mathcal{L}_{CLS}, \quad (4)$$

where

- α is the weighting factor for the language modeling loss; and
- β is the weighting factor for the classification loss.

5.3.5. Training Setup and Configuration

The GPT-2 was fine-tuned on a meticulously curated dataset to support our multitask learning objectives, which include both classification and information retrieval tasks. This dual approach not only classifies relevant tweets but also extracts valuable information, crucial for enhancing traffic safety measures [28]. The model handles various classification and information retrieval tasks. The features extracted for fine-tuning the model are detailed in Table 8.

Table 8. The extracted features for fine tuning.

No.	Feature	Task
1	Was there a road traffic accident?	Classification
2	How bad was the road traffic accident?	Classification
3	Was the driver identified?	Classification
4	What is the collision type?	Classification
5	What is the sentiment?	Classification
6	What is the emotion?	Classification
7	Who is the driver?	Information Retrieval (IR)
8	How many people were injured?	Information Retrieval (IR)

Table 8. *Cont.*

No.	Feature	Task
9	How many people died?	Information Retrieval (IR)
10	What is the location of the accident?	Information Retrieval (IR)
11	What is the contributing factor to the accident?	Information Retrieval (IR)
12	What is the car type involved?	Information Retrieval (IR)
13	What is the crash event type?	Information Retrieval (IR)
14	What was the driver error?	Information Retrieval (IR)
15	What was the scenario in this narrative?	Information Retrieval (IR)

The hyperparameters selected for the training phase are detailed in Table 9, ensuring transparency and reproducibility of our methods [8,11]. This approach aligns with the best practices in machine learning for natural language processing, as highlighted by Vaswani et al. [22] in their foundational paper on transformers, Wolf et al. [63] in their comprehensive guide on utilizing the Hugging Face Transformers library, and Radford et al. [11], who demonstrate the potential of language models like GPT-2 to learn a variety of language processing tasks in an unsupervised manner.

Table 9. GPT-2 hyperparameter values.

Hyperparameter	Value
Model Name	GPT-2 Medium
Max Length	256 tokens
Batch Size	6
Learning Rate	5×10^{-5}
Model/Tokenizer Name	GPT2-medium
Weight Decay	0.0
Number of Epochs	12
Optimizer	AdamW
Gradient Accumulation Steps	128
Warmup Steps	100
Evaluation Strategy	Steps
Save Steps	32
Eval Steps	32
Logging Steps	4
Output Directory	"/path/to/output_dir"

5.3.6. Optimization and Learning Rate Scheduling

The optimization process used the Adam optimizer with a cosine annealing learning rate schedule. This helps in gradually reducing the learning rate to a minimum value over the training period. The gradient descent update and learning rate schedule are described by Equations (5) and (6):

$$\theta t + 1 = \theta t - \eta \nabla_\theta \mathcal{L}, \tag{5}$$

where

- θ: the parameters or weights of the model;
- η: the learning rate, a hyperparameter that controls how much to change the model parameters in response to the gradient of the loss function;
- ∇_θL: the gradient of the loss function L with respect to the parameters θ;
- L: the loss function that measures how well the model's predictions match the actual data.

$$\eta_t x = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})\left(1 + \cos\left(\frac{t}{T}\pi\right)\right), \tag{6}$$

where

- η_t: the learning rate at time t;
- η_{min}: the minimum learning rate;
- η_{max}: the maximum learning rate;
- t: the current time step or iteration;
- T: the total time or the period over which the learning rate schedule is applied;
- π: the mathematical constant pi, approximately equal to 3.14159.

5.4. Baseline Benchmark Models

Given the limited studies exploring multitask learning or advanced large language models (LLMs) for multitask objectives, our research aimed to establish a comprehensive benchmark. To achieve this, we included GPT-4o mini in a zero-shot setting and XGBoost as baseline models. GPT-4o mini, renowned for its advanced language understanding capabilities, serves as an ideal comparison against fine-tuned models, offering insights into performance without task-specific training [58]. On the other hand, XGBoost, a traditional and widely respected machine learning model, was selected for its efficiency and robust performance in classification tasks [57]. This approach provides a strong foundation for evaluating the effectiveness of the GPT-based models in handling RTC-related data.

5.4.1. Prompt Engineering Using Zero-Shot GPT-4o Mini

To benchmark the performance of the fine-tuned GPT-2 model, we employed GPT-4o mini in a zero-shot setting. Zero-shot learning enables the model to handle tasks without task-specific training, offering insights into the model's ability to generalize across various tasks inherently. For this experiment, we used the same 700-sample test set, formatted in JSON structure as prompt-response pairs, that was previously employed for GPT-2 fine-tuning. Utilizing this consistent dataset ensured a direct and fair comparison between the models.

The prompts for GPT-4o mini were designed to elicit precise responses to specific queries within the tweets. Each prompt was aligned with the 15 predefined features used in the GPT-2 fine-tuning process. These prompts were meticulously crafted with clear instructions to ensure the model's output adhered to the required format and avoided unnecessary verbosity. Figure 4 illustrates examples of prompts used to determine whether a tweet described a road accident and its severity. The complete set of prompts for the various tasks is provided in Appendix B.

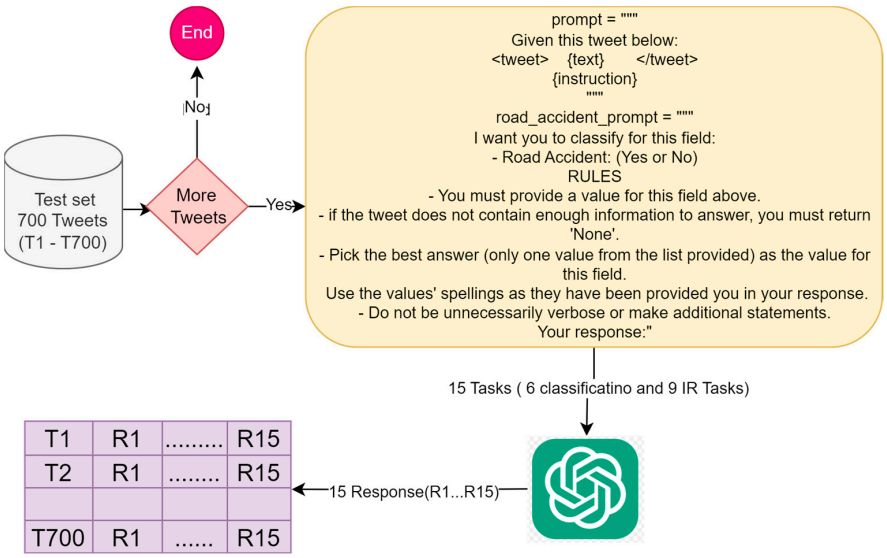

Figure 4. Prompting GPT-4o mini and handling responses.

5.4.2. XGBoost Model Training

XGBoost was included as a traditional machine learning baseline to compare its performance with that of the GPT models. XGBoost is renowned for its speed and effectiveness in classification tasks, making it a suitable benchmark against more complex models like GPT-2 and GPT-4o mini.

For each classification task, the same dataset used for GPT-2 fine-tuning was employed to train the XGBoost model. The data were preprocessed by converting the text features into numerical representations using TF-IDF (Term Frequency-Inverse Document Frequency) vectors, ensuring that the textual data were appropriately formatted for XGBoost's gradient boosting algorithm.

We conducted a search through the literature to identify the best hyperparameter values for XGBoost. Based on this search, we determined the optimal final configuration for our model: n_estimators=500, use_label_encoder=False, and eval_metric='mlogloss.' These values were selected to balance performance and computational efficiency, ensuring a robust baseline for comparison.

It is important to note that XGBoost was trained on specific individual tasks, focusing solely on six classification tasks. This was due to the model's inherent design as a traditional machine learning approach, which contrasts with the multitask learning capability of GPT-2. The focus on classification tasks only was determined by the nature of the RTC-related data and the comparative purpose of this experiment.

6. Analysis and Results

6.1. Fine-Tuned GPT-2 Performance

To comprehensively assess the effectiveness of our models across both classification and information retrieval (IR) tasks, we employ a suite of widely recognized statistical metrics tailored to each task type. For the classification tasks, we utilize accuracy, F1-score, recall, and precision to evaluate how effectively the model performs the classification tasks. For IR tasks, which focus on extracting or generating relevant information from the data, we measure performance using BLEU-4 [61] as shown in Equation (7), which measures the precision of 4-g between the model output and reference text, ROUGE-I [60], which evaluates the recall of unigrams and is shown in Equation (8), and Word Error Rate (WER) [62] as shown in Equation (9), which measures the errors in the generated text compared to the reference text. These metrics help assess how closely the model's outputs align with human-generated reference texts. BLEU-4 measures the correspondence of n-grams between the model's output and reference texts, focusing on precision. Generally, the higher, the better. ROUGE-I focuses on recall by comparing the overlap of unigrams between the generated text and the reference. Focusing on recall in the evaluation. Generally, the higher, the better. Word Error Rate (WER) indicates the proportion of errors in the generated text compared to the reference. It is determined by counting the number of substitutions, insertions, and deletions required to convert the system output into the reference text, then normalizing this count by the total number of words in the reference text. Generally, the lower, the better. These metrics are instrumental in evaluating the quality and relevance of information retrieved or generated by the model, as described by [61] for BLEU-4, [60] for ROUGE-I, and [62] for WER.

$$BLEU - 4 = BP. \exp\left(\sum_{n=1}^{4} w_n log\ p_n\right), \tag{7}$$

where BP is the brevity penalty, w_n are the weights, and p_n is the precision of n-grams.

$$ROUGE - I = \frac{Number\ of\ Overlapping\ Unigrams}{Total\ Number\ of\ Unigrams\ in\ Reference}, \tag{8}$$

$$WER = \frac{S + D + 1}{N}, \tag{9}$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the number of words in the reference.

Table 10 provides a detailed breakdown of accuracy, precision, recall, and F1-score, enabling a thorough evaluation of the model's ability to predict diverse aspects of traffic crashes. Figure 5 presents the confusion matrices for these six classification features, visually illustrating the model's accuracy in predicting each feature.

Table 10. Model evaluation per class.

Feature	Class Value	Precision	Recall	F1-Score
Was there a road traffic accident?	Yes	0.825	0.842	0.833
	No	0.975	0.972	0.974
What is the severity of the road traffic accident?	Fatal	0.933	0.953	0.943
	Moderate	0.638	0.530	0.579
	Critical	1.000	0.778	0.875
	None	0.877	0.744	0.805
	Mild	0.754	0.833	0.792
	Unknown	0.000	0.000	0.000
	Severe	0.812	1.000	0.897
What is the collision type?	Single vehicle	0.400	0.333	0.364
	Multiple Vehicle	0.000	0.000	0.000
	Broadside	0.647	0.688	0.667
	Chain reaction	0.000	0.000	0.000
	Hit-and-run	0.589	0.509	0.546
	Stationary object	0.500	0.333	0.400
	Pedestrian	0.342	0.283	0.310
	Not applicable	0.852	0.913	0.881
What is the sentiment?	Negative	0.901	0.967	0.933
	Positive	0.887	0.759	0.818
What is the emotion?	Sympathy	0.000	0.000	0.000
	Gratitude	0.750	0.545	0.632
	Love	1.000	0.000	1.000
	Disgust	0.500	0.167	0.250
	Anger	0.723	0.814	0.766
	Neutral	0.643	0.517	0.573
	Confusion	0.000	0.000	0.000
	Curiosity	0.000	0.000	0.000
	Sadness	0.784	0.892	0.835
	Empathy	0.000	0.000	0.000
	Happy	0.698	0.833	0.759
	Fear	0.765	0.580	0.660
Was the driver identified?	Yes	0.958	0.902	0.929
	No	0.936	0.971	0.953
	Unknown	0.500	0.308	0.381

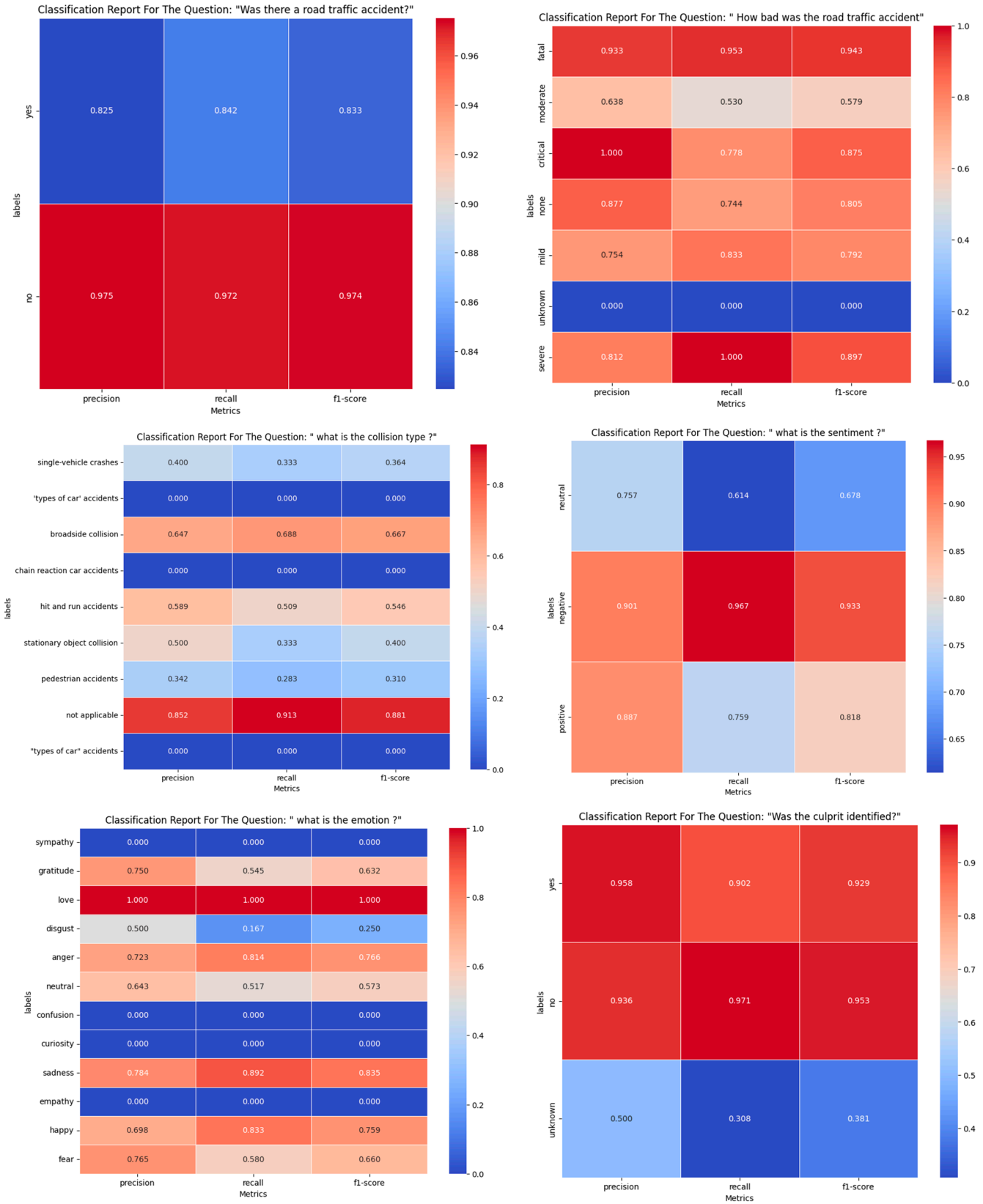

Figure 5. Confusion matrices for the six classification tasks.

Results showed that our model exhibits strong performance in predicting the absence of a road traffic accident, with a high precision of 0.975 and a recall of 0.972, leading to an impressive F1-score of 0.974. This indicates a high degree of accuracy in identifying negative

cases, ensuring that false positives are minimal. Additionally, the model demonstrates reliable performance for identifying the presence of an accident with a precision of 0.825, recall of 0.842, and an F1-score of 0.833. These results affirm that the model is proficient and dependable in classifying both positive and negative instances of road traffic accidents.

Our model's performance is particularly notable for the "fatal" and "severe" accident categories. For "fatal" accidents, the model achieves high precision (0.933), recall (0.953), and an F1-score of 0.943, reflecting its strong capability to accurately identify the most severe accidents. Similarly, for "severe" accidents, the model achieves a precision of 0.812, recall of 1.000, and an F1-score of 0.897, ensuring that severe cases are correctly identified. These results underscore the model's effectiveness in assessing accident severity, providing critical insights for emergency response and resource allocation.

The model performs exceptionally well in identifying collision types, with high precision, recall, and F1-score for some classes. For "broadside collisions", the model achieves a balanced performance with a precision of 0.647, recall of 0.688, and an F1-score of 0.667, demonstrating its ability to handle more complex classification tasks effectively.

The sentiment analysis model shows commendable performance, particularly for "negative" sentiment, with a precision of 0.901, a recall of 0.967, and an F1-score of 0.933. This high level of accuracy ensures that the model can reliably detect negative sentiments, which is crucial for understanding public perception and feedback. The model also performs well for "positive" sentiment, with a precision of 0.887, recall of 0.759, and an F1-score of 0.818, indicating its applicability in sentiment analysis tasks.

Our model also demonstrates outstanding performance in identifying "love" with perfect precision, recall, and an F1-score of 1.000. This highlights the model's capability to accurately detect strong, positive emotions. Additionally, the model shows good performance for "sadness", with a precision of 0.784, recall of 0.892, and an F1-score of 0.835, and for "anger", with a precision of 0.723, recall of 0.833, and an F1-score of 0.774. These results emphasize the model's reliability in emotion classification, which can be instrumental in various applications, such as customer feedback analysis and mental health monitoring.

The model excels in predicting both "yes" and "no" cases for culprit identification, with high precision (0.958 and 0.936), recall (0.902 and 0.971), and F1-scores (0.929 and 0.953), respectively. This indicates a strong capability in distinguishing between cases where the culprit was identified and where they were not. Such reliable performance is crucial for applications in law enforcement and incident reporting.

The classification tasks demonstrate a high level of performance across multiple categories, underscoring the reliability and applicability of our models. The models exhibit exceptional accuracy in identifying severe accidents, negative sentiments, and specific emotions such as "love" and "sadness", showcasing their robustness in handling complex classification tasks. The consistently high precision, recall, and F1-scores across various tasks highlight the models' dependability and consistency in real-world applications. These results affirm the strengths of our models and their potential to provide valuable insights across different domains, ensuring that our work is both reliable and highly applicable.

Nonetheless, evaluating the model's performance on IR tasks presented a unique challenge. Unlike classification tasks with predefined categories, these tasks demanded assessing the model's ability to generate coherent and informative natural language responses to open-ended questions. Table 11 summarizes the model's performance using different metrics.

The highest BLEU-4 score is achieved for the query "What is the crash event type?" with a score of 0.47, suggesting that the model excels at accurately retrieving specific details about the type of crash event. Other queries, such as "What was the location?" and "What was the driver error?" also show respectable BLEU-4 scores of 0.26 and 0.23, respectively, highlighting the model's ability to handle questions requiring precise information retrieval. The highest ROUGE-I score is 0.87 for the query "How many people died?" This indicates that the model is particularly effective at retrieving information related to fatalities. Other tasks, such as "How many people were injured?" and "What is the crash event type?" also

score high with ROUGE-I values of 0.85 and 0.81, respectively, demonstrating the model's robustness in extracting relevant information. The query "How many people died?" has the lowest WER of 0.13, indicating high accuracy in retrieving fatality counts. Conversely, the query "What was the case scenario?" has the highest WER of 0.75, suggesting challenges in accurately retrieving complex scenario details. Nonetheless, for most other tasks, such as "What car was involved?" and "What is the crash event type?" with WERs of 0.22 and 0.28, respectively, the model performs reasonably well.

Table 11. Model evaluation metrics for IR tasks.

Feature	BLEU-4	ROUGE-I	WER
How many people were injured?	0.15	0.85	0.15
How many people died?	0.15	0.87	0.13
What was the location?	0.26	0.80	0.27
What are the contributing factors?	0.16	0.78	0.26
What car was involved?	0.15	0.80	0.22
What is the crash event type?	0.47	0.81	0.28
What was the case scenario?	0.15	0.58	0.75
What was the driver error?	0.23	0.75	0.37
Was the culprit driver identified?	0.21	0.80	0.28
Average metric performance across all features	0.22	0.78	0.30

Across all IR tasks, the model showed a balanced performance with an average BLEU-4 score of 0.22, ROUGE-I score of 0.78, and WER of 0.30. These results indicate that the model is reliable and effective in handling a wide range of queries related to road traffic accidents. The high ROUGE-I scores and relatively low WER values underscore the model's capability to retrieve accurate and relevant information, making it a valuable tool for information retrieval in traffic crash analysis.

To assess the model's performance on IR tasks, we evaluated its ability to answer various questions related to RTCs mentioned in tweets. The examples provided in Table 12 illustrate the model's strong performance in several specific IR tasks, including identifying collision types, determining the number of injuries, identifying contributing factors, pinpointing locations, describing case scenarios, and recognizing driver errors. The model demonstrates high accuracy in retrieving relevant information for these tasks. While some retrievals were not entirely accurate, the overall results indicate the model's reliability and applicability in processing and understanding complex real-world scenarios as described in tweets.

Table 12. Examples of IR tasks labeling and prediction.

Tweets for Each Feature	GPT-3.5 Labels	Model Prediction
What is the collision type?		
Six people were hurt, including four pedestrians, in a serious hit-and-run crash in River North, Chicago police said.	hit-and-run	pedestrian
R.T. @DerrickRBickley: Like a "tough, all-action thriller" and would love a "book you can't put down" THE HIT-AND-RUN MAN at or your favourite digital store and PAPERBACK HARDBACK AUDIOBOOK plus other formats	hit-and-run	not applicable
Woman dies from injuries two weeks after hit and run in Darwin's CBD	pedestrian	hit-and-run
@yourallon Hit and run mission, we were there for about 5 min.	hit-and-run	not applicable

Table 12. *Cont.*

Tweets for Each Feature	GPT-3.5 Labels	Model Prediction
How many people were injured?		
Harrison Grey in an induced coma after Bees Creek Road, Darwin hit and run	1	1
Three people and a dog have avoided serious injury after a multi-car crash in Oxenford. It happened at the intersection of Hope Island Road and the Pacific Highway exit.	3	3
R.T. @DerrickRBickley: Fancy a reader-acclaimed FIVE/FOUR star but no E-reader? THE HIT AND RUN MAN is available in three print formats: PAPERBACK HARDBACK (Amzn) (B&N) LARGE PRINT	unknown	0
R.T. @9NewsSyd: Three teenage boys on their way home from school have been hit by a car at a Sydney pedestrian crossing. The driver leaving the scene without stopping to help. @LizzyBryan1	3	3
What were the contributing factors?		
There are reports a motorcycle was seen leaving the scene in Gnangara after the shooting.	shooting	shooting
R.T. @mlmartens: @MissionRCMP It was a hit and run. Why are excusing the behaviour of the driver?	driver error	hit-and-run
FEED: BUS @TransLinkSEQ: Buses travelling along Waterworks Rd, The Gap are delayed up to 15 min due to a traffic accident.	not applicable	not applicable
2 Alberta students dead, 3 in serious condition after single-vehicle car crash <prompt> what is the contributing factor to the accident?:	unknown	unknown
What is the location?		
A witness to a fatal hit-and-run crash in Melbourne in which a woman was dragged alongside a moving car, has told how neighbours desperately tried to save the woman.	melbourne	melbourne
R.T. @MaggieRaworth: A man has been killed after a hit and run incident in Hampton Park this morning. Police have created a crime scene in front of Kilberry Valley Primary School. @9NewsMelb	hampton park in front of kilberry valley primary school	hampton park
R.T. @manny745: A Walk to Revenge A drizzly Manchester night, a hit and run. A chance meeting decades later leads ultimately to the exposure of a transatlantic crime family.	manchester	manchester
@sultan_sydney hit and run sultsssssss	sydney	unknown
What is the case scenario?		
A car tied to two separate hit-and-run incidents in Lexington, including one involving a child, has been found and the driver questioned, according to Lexington Police.	child involved in one incident	hit-and-run incident involving a child
An elderly man who suffered critical injuries in an alleged hit-and-run in Sydney has undergone a number of surgeries, as police share footage of the vehicle they believe is involved.	elderly man suffered critical injuries	elderly man undergoing surgery
Bicyclist killed in Bronx hit and run; driver fled in another car with baby.	hit-and-run resulting in fatality	hit-and-run resulting in fatality
The family of a woman allegedly run down in a hit and run on a suburban street while walking her dog has made a heartbreaking plea for her recovery from critical injuries.	woman walking her dog hit by a car	hit-and-run while walking her dog

Table 12. *Cont.*

Tweets for Each Feature	GPT-3.5 Labels	Model Prediction
What is the driver error?		
@etsysofttotouch @BandB_CBS And covered up her drunken hit and run of Darla and slept with her daughter's love Rick!	driving under the influence	driving under the influence
Child in serious condition after being struck by hit-and-run driver evading Queensland police	fleeing from law enforcement	evading police
@laurarichards99 @jessradio I hope you are sitting—you may start to feel the rage after reading this. It is actually based around a hit and run case in Oz but somehow it has become this, a petition about changing the law against women?	reckless driving	not applicable
Judge Raoul Neave strikes again. A decade after going easy on hit-and-run investment banker Guy Hallwright, he's done it again to a recidivist drink driver who ran a red light, drunk, killed an innocent driver then fled the scene.	running red light	driving under the influence
What is the crash event type?		
Police have seized a car used in a suspected hit and run that killed a father of four in Fairfield yesterday morning.	hit-and-run	hit-and-run
@BLUEfingers2021 Only if you're in a new Jag. Also, it's obligatory to drive into the side of a house and repeatedly attempt to leave the scene.	single-vehicle	not applicable
1 driver taken to hospital following 3-vehicle collision in downtown Cedar City	chain reaction car accidents	chain reaction car accidents
TRAFFIC LIGHTS WENT DOWN, IMPAIRED CHARGES LAID after single vehicle collision 3:20 am Sept 18 at Main & Dundas St, Cambridge. Vehicle struck pole, traffic lights fell. Cambridge man 23 charged w dangerous driving, impaired, impaired 80+. Intersection reopened after repairs.	single-vehicle crashes	single-vehicle crashes

6.2. Baseline Model Performance: GPT-4o Mini and XGBoost

In this section, we evaluate the performance of the baseline models, GPT-4o mini and XGBoost, across the same classification and information retrieval (IR) tasks and the same test set used to assess the fine-tuned GPT-2 model. The performance metrics include accuracy, precision, recall, and F1-score for classification tasks and BLEU-4, ROUGE-I, and Word Error Rate (WER) for IR tasks.

6.2.1. Classification Task Performance

Table 13 provides a comparative summary of the performance of GPT-4o mini and XGBoost on the six classification tasks, alongside the results from GPT-2. Overall, GPT-2 consistently outperforms both baseline models across most classification tasks, reflecting the benefits of task-specific fine-tuning. However, GPT-4o mini and XGBoost each demonstrate particular strengths, with XGBoost showing robust performance in tasks where simpler, more structured data can be leveraged and GPT-4o mini exhibiting high recall in certain tasks despite lower precision.

Table 13. Model performance for classification tasks.

Classification Task	Model	Accuracy	Precision	Recall	F1-Score
Is there a road traffic accident?	XGBoost	0.950	0.920	0.850	0.880
	GPT-2	0.954	0.900	0.907	0.903
	GPT-4	0.856	0.741	0.912	0.780
Was the driver identified?	XGBoost	0.930	0.870	0.700	0.750
	GPT-2	0.924	0.798	0.727	0.755
	GPT-4	0.610	0.315	0.303	0.307
What is the severity of the accident?	XGBoost	0.820	0.780	0.730	0.740
	GPT-2	0.831	0.716	0.691	0.699
	GPT-4	0.476	0.469	0.551	0.380
What is the emotion in this tweet?	XGBoost	0.680	0.420	0.340	0.360
	GPT-2	0.743	0.489	0.446	0.456
	GPT-4	0.496	0.372	0.389	0.308
What is the collision type?	XGBoost	0.770	0.530	0.380	0.430
	GPT-2	0.777	0.370	0.340	0.352
	GPT-4	0.533	0.243	0.263	0.203
What is the sentiment in the tweet?	XGBoost	0.860	0.830	0.720	0.770
	GPT-2	0.879	0.849	0.780	0.810
	GPT-4	0.861	0.786	0.811	0.797

6.2.2. Information Retrieval Task Performance

The IR tasks presented unique challenges for the baseline models, particularly for GPT-4o mini, which was evaluated in a zero-shot setting without task-specific fine-tuning. Table 14 compares the performance of GPT-4o mini and GPT-2 across the nine IR tasks. The results show that GPT-2 significantly outperforms GPT-4o mini in both BLEU-4 and ROUGE-I metrics, highlighting the importance of fine-tuning in generating more accurate and contextually appropriate responses. GPT-4o mini, while capable of retrieving information, shows a substantial drop in performance, particularly evident in the higher WER values across most tasks.

Table 14. Model evaluation metrics for information retrieval tasks.

Feature	GPT-2 BLEU-4	GPT-4 BLEU	GPT-2 ROUGE-I	GPT-4 ROUGE-L	GPT-2 WER	GPT-4 WER
How many people were injured?	0.15	0.0203	0.85	0.1143	0.15	0.8857
How many people died?	0.15	0.1331	0.87	0.7486	0.13	0.2514
What was the location?	0.26	0.2105	0.80	0.6644	0.27	0.3506
What are the contributing factors?	0.16	0.0041	0.78	0.0162	0.26	1.7179
What car was involved?	0.15	0.1164	0.80	0.6324	0.22	0.3689
What is the crash event type?	0.47	0.0530	0.81	0.1146	0.28	0.9752
What was the case scenario?	0.15	0.0144	0.58	0.0976	0.75	11.9861
What was the driver error?	0.23	0.0017	0.75	0.0081	0.37	1.2303
Who Was the culprit?	0.21	0.0529	0.80	0.2964	0.28	0.8771
Average Metric Performance	0.22	0.0674	0.78	0.2992	0.30	2.0715

6.2.3. Comparative Analysis of Baseline Models

In comparing the baseline models—GPT-4o mini and XGBoost—with the fine-tuned GPT-2 model, it is evident that GPT-2 consistently outperforms the baseline models across most classification and information retrieval tasks. This superiority is particularly noticeable in the metrics of accuracy, F1-score, precision, and recall, where GPT-2 demonstrates a clear advantage due to the benefits of task-specific fine-tuning.

Classification Task Performance

Figure 6 presents a comparative analysis of the accuracy for the three models across the six classification tasks. The fine-tuned GPT-2 model shows superior accuracy across all tasks, with the most significant performance gap observed in the tasks of "Identifying Road Traffic Accidents" and "Severity of the Accident". For instance, while XGBoost achieves an accuracy of 0.950 in detecting road traffic accidents, GPT-2 slightly outperforms it with an accuracy of 0.954. GPT-4o mini (zero shot), in contrast, lags with an accuracy of 0.856, highlighting the impact of fine-tuning in improving model performance.

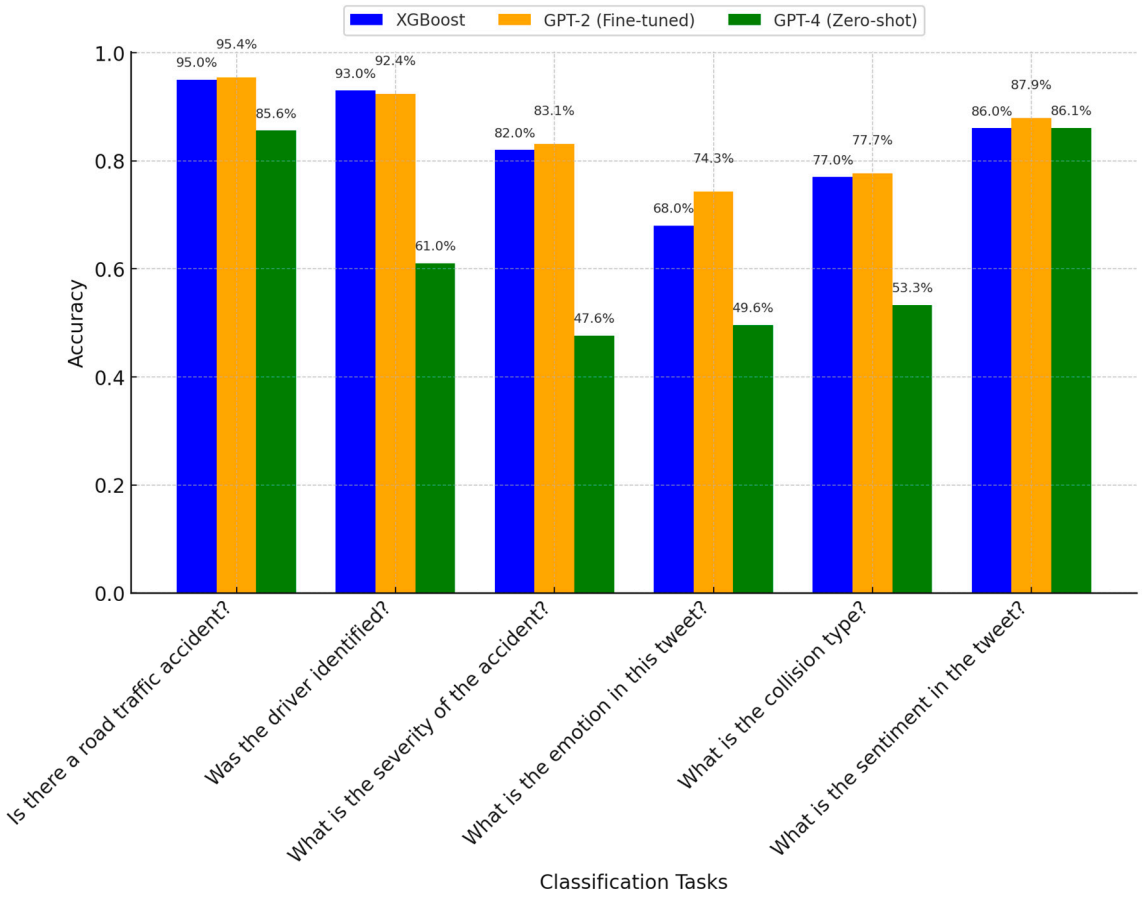

Figure 6. Accuracy for the three models across the six classification tasks.

Similarly, the F1-Score comparison in Figure 7 emphasizes GPT-2's dominance, particularly in the "Sentiment Detection" and "Driver Identification" tasks, where it outperforms both XGBoost and GPT-4o mini. The GPT-2 model's F1-scores reflect its balanced precision and recall, essential for reliable classification in nuanced tasks like emotion and sentiment detection.

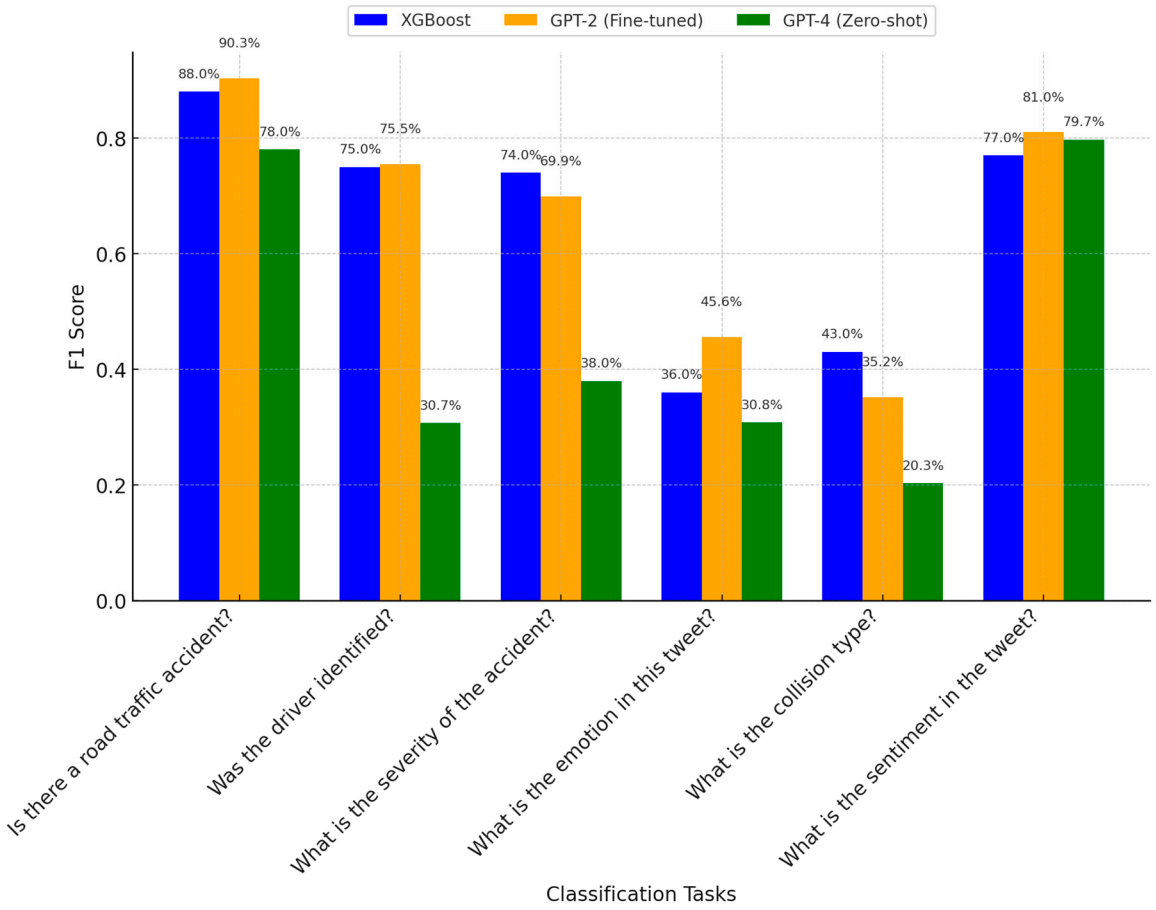

Figure 7. F1-score for the three models across the six classification tasks.

Information Retrieval Task Performance

The challenges presented by information retrieval (IR) tasks are evident when comparing GPT-2 with GPT-4o mini. Figure 8 shows the overall average performance metrics across all IR tasks for both models. GPT-2 outperforms GPT-4o mini significantly in BLEU-4 and ROUGE metrics, indicating its superior capability in generating contextually appropriate and accurate responses. For example, the average BLEU-4 score for GPT-2 across all tasks is 0.22, compared to GPT-4o mini's 0.0674, illustrating the gap in generative quality.

Similarly, the ROUGE score for GPT-2 is 0.78, significantly higher than GPT-4o mini's 0.2992. This trend highlights GPT-2's strength in retrieving and generating relevant information, which is crucial for accurate information retrieval. Additionally, GPT-4o mini's higher WER score of 2.0715, compared to GPT-2's WER of 0.30, underscores the limitations of zero-shot learning without task-specific fine-tuning.

The comparative analysis demonstrates that while GPT-4o mini and XGBoost serve as competent baseline models, particularly in structured classification tasks, the fine-tuned GPT-2 model consistently delivers superior performance across both classification and information retrieval tasks. The results underline the importance of fine-tuning in enhancing model capabilities, making GPT-2 the most reliable and effective model for handling the complexities of road traffic accident data.

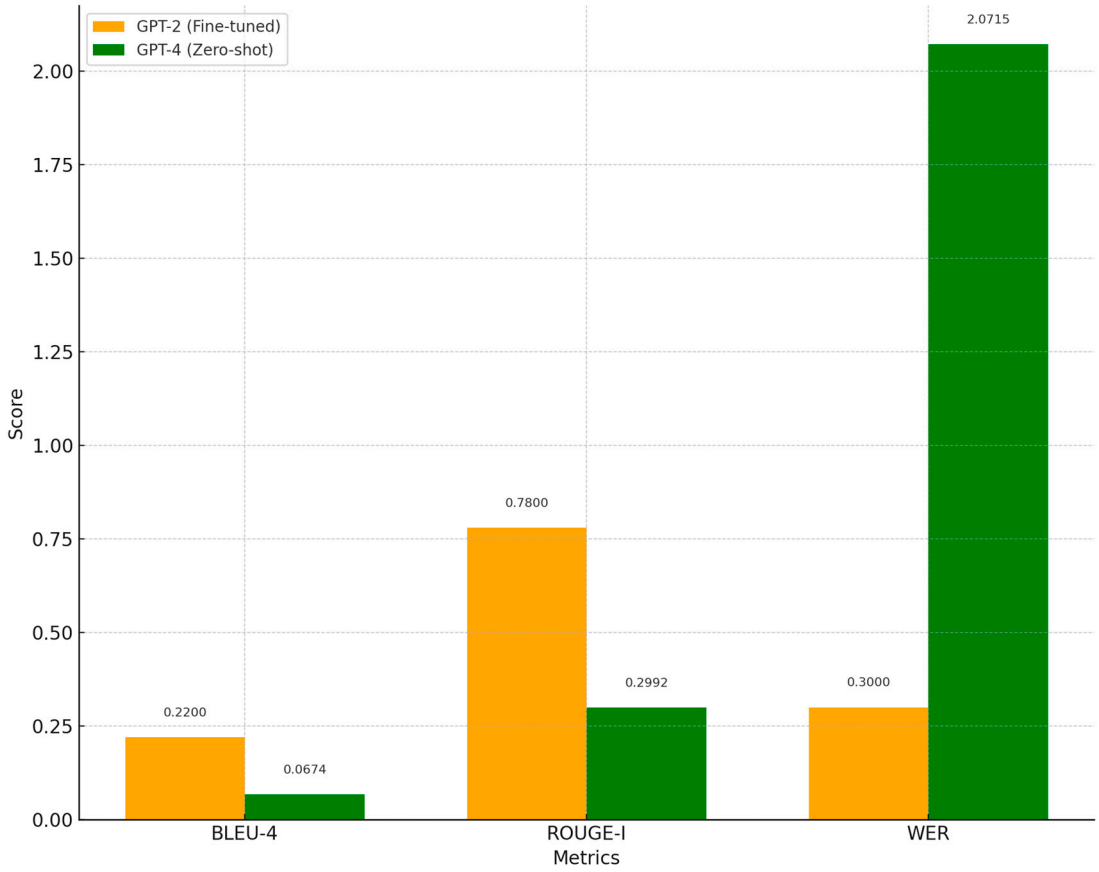

Figure 8. Average BLEU-4, ROUGE-I/ROUGE-L, and WER scores across information retrieval tasks.

7. Discussion

7.1. Multitask Learning Framework for Enhanced RTC Analysis

This study introduces an innovative multitask learning (MTL) framework that leverages large language models (LLMs) to analyze road traffic crashes (RTCs) using real-time social media data, particularly from Twitter (now X). By employing advanced natural language processing (NLP) techniques, we demonstrate the potential of LLMs, specifically GPT-2 fine-tuned within our framework, to enhance road safety measures through timely and detailed insights.

The MTL framework efficiently handles multiple classification and information retrieval tasks simultaneously, significantly improving the extraction and classification of information from unstructured social media data. The integration of GPT-2 for fine-tuning highlights the model's adaptability in processing large volumes of diverse text data, leading to high precision and recall in tasks such as detecting road traffic accidents and assessing accident severity.

Our results underscore the reliability and effectiveness of the MTL framework. The fine-tuned GPT-2 model consistently achieved high performance across various metrics, making it a robust tool for traffic safety analytics. The practical implications of these results are substantial, offering stakeholders actionable insights for emergency response and traffic management. Furthermore, this approach addresses challenges related to data scarcity and imbalance, ensuring that even limited datasets can produce reliable and comprehensive insights.

7.2. Comparison with GPT-4 Zero-Shot Baseline

In comparison, the GPT-4o mini model, evaluated in a zero-shot setting, and the XGBoost model, a traditional machine learning baseline, provided valuable insights into the strengths and limitations of different approaches. GPT-4o mini, while capable of handling

tasks without specific training, showed lower performance across all classification and information retrieval tasks compared to the fine-tuned GPT-2 model. This difference underscores the importance of fine-tuning LLMs for domain-specific tasks, where contextual understanding and precision are critical.

XGBoost, on the other hand, demonstrated strong performance in structured classification tasks, particularly those involving more straightforward data patterns. However, it struggled with more nuanced tasks that required deeper contextual understanding, highlighting the limitations of traditional machine learning models in handling complex, unstructured data compared to LLMs.

Overall, these comparisons reinforce the superiority of fine-tuned LLMs like GPT-2 for complex and diverse RTC-related tasks. While GPT-4o mini and XGBoost serve as competent baselines, the results emphasize the necessity of task-specific fine-tuning to achieve high accuracy and relevance in real-time traffic safety analysis.

8. Conclusions and Future Work

This study has demonstrated the effectiveness of a multitask learning framework (MTF) leveraging large language models (LLMs) for the real-time analysis of road traffic crashes (RTCs) using social media data. The robust performance observed across both classification and information retrieval tasks underscores the potential of this approach to significantly enhance road safety measures. By delivering detailed, timely, and actionable insights, our methodology represents a substantial advancement in traffic safety analytics, offering valuable tools for emergency responders, urban planners, and policymakers focused on improving public safety on the roads.

8.1. Key Contributions

1. **Development of multitask learning framework (MMF) for classification and information retrieval**: this study introduced a sophisticated MTF that utilizes LLMs to manage multiple classification and information retrieval tasks simultaneously, enabling a more comprehensive and efficient analysis of RTCs.
2. **Curated dataset**: We developed a curated dataset specifically designed to fine-tune models for more accurate RTC analysis. This dataset includes a variety of labels and classes relevant to road traffic accidents, contributing significantly to the research community and providing a valuable resource for further RTC-related studies.
3. **Automated labeling using prompt engineering**: The study employed prompt engineering techniques to automate the labeling process, enhancing both the efficiency and accuracy of data annotation. This automation is essential for scaling the analysis to larger datasets while maintaining consistency in labeling.
4. **Benchmarking fine-tuning vs. prompt engineering**: By comparing the performance of fine-tuned models against those utilizing prompt engineering in a zero-shot setting, this study sets a benchmark for future research. This comparison provides valuable insights into the strengths and limitations of different methodologies, guiding the development of more effective approaches in RTC analysis.

8.2. Limitations and Future Directions

While this study presents promising results, there are several limitations to consider. The reliance on Twitter data introduces potential biases due to the demographic and geographic distribution of its users. Additionally, the current dataset may not encompass the full spectrum of road traffic scenarios. Future research should aim to broaden the dataset by collecting more Twitter data to cover a wider range of scenarios. This would enhance the robustness and accuracy of the findings and provide a more comprehensive analysis of road traffic crashes.

As the field of LLMs evolves rapidly, future research could explore more advanced models and fine-tune other lightweight open-source models to further enhance the extraction and classification of RTC-related information. Additionally, leveraging other sources

of crash data, such as images and videos from traffic cameras, could provide a more comprehensive analysis of road traffic crashes by adding visual context and insights that are not available through text alone.

8.3. Implications for Research and Practice

The findings of this study have significant implications for both research and practical applications in traffic safety. By utilizing advanced AI techniques and social media data within a multitask learning framework, we have introduced a transformative approach to real-time RTC analysis. This methodology not only enhances proactive decision-making for emergency response teams but also supports the development of targeted interventions and policy adjustments aimed at reducing traffic accidents and improving road safety.

Additionally, the curated dataset generated through this research represents a valuable resource for the traffic safety community. Researchers can leverage this dataset to extract deeper insights, develop and validate predictive models, and enhance real-time monitoring systems. The integration of diverse data sources demonstrated in this study has the potential to greatly enrich our understanding of road safety dynamics, ultimately informing more comprehensive strategies for accident prevention and management.

Author Contributions: S.J.: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing—original draft, Writing—review and editing. M.E.: Supervision, Methodology, Conceptualization and Writing—review and editing. R.N.: Supervision, Methodology and Review. A.P.: Supervision. H.I.A.: Conceptualization, Review and Proofreading. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a PhD scholarship from Queensland University of Technology (QUT). The Article Processing Charge (APC) was funded by the Centre of Data Science at QUT.

Data Availability Statement: Data will be made available on request.

Acknowledgments: We extend our gratitude to the Digital Observatory at Queensland University of Technology for providing the data essential for this research. We also thank Queensland University of Technology for awarding a scholarship that supported my research. Additionally, we are grateful to the Centre of Data Science at Queensland University of Technology for funding the Article Processing Charge (APC) for this publication.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Example Labeled Data from Traffic Incident Tweets

This appendix provides detailed examples of the labeled data used in model development, showcasing how the raw tweets were processed and annotated for various classification tasks. Tables A1 and A2 below provide examples of the labeled data and the distribution of data labels across the six classification tasks. The careful curation of these samples was essential to ensure that the machine learning models were trained and evaluated on diverse and representative data, which is crucial for effective performance in real-world applications.

Table A1. Example tweets used for training (part 1).

Tweet	Road Accident	Severity	Culprit	Culprit Apprehended	No of Injured	No of Deaths
"23 yr old man charged over Caboolture hit and run that killed Collin Young"	Yes	Fatal	23 yr old man	Yes	None	1
"Teenage driver arrested, woman in critical condition after alleged hit-and-run in Darwin"	Yes	Critical	Teenage driver	Yes	1	None

Table A1. *Cont.*

Tweet	Road Accident	Severity	Culprit	Culprit Apprehended	No of Injured	No of Deaths
"A professional Perth golfer accused of a hit-and-run that killed an elderly man on the freeway in October 2019 claims he was suffering a medical episode at the time and did not have control of his body."	Yes	Fatal	Professional Perth golfer	No	None	1
"#Duingal—Three stable patients have been transported to Bundaberg Hospital after a traffic incident on the Bruce Highway at 12.46pm."	Yes	Mild	None	None	3	0
"UPDATE: Peel St remains closed northbound at Swan St due to a 2-car crash with a person trapped. Diversions are via Swan St, Johnson St, Tribe St, Moore Creek Rd, Browns Ln to rejoin Manilla Rd near Hallsville. Continue to allow extra travel time."	Yes	Moderate	None	None	1	None
"#Beenleigh—Two vehicle traffic incident with three patients, all with minor injuries. Paramedics transporting two patients to Logan Hospital in a stable condition."	Yes	Mild	None	None	3	0
"Monday—Mixed 4's: Slap That Ace beat Hit n Run (23–15)"	No	None	None	None	None	None
Brisbane City: several lanes are closed on Countess Street due to a traffic incident where a truck has crashed into a rail bridge. Motorists are advised to use Hale Street as an alternate route, and avoid the area or expect delays.	Yes	Moderate	Truck	None	None	None
7NEWS understands a 13-year-old boy has died at the Children's Hospital after yesterday's stolen car crash in Oakey. A 14-year-old boy remains on life support—while another teenager has been released from the Base Hospital and charged.	Yes	Fatal	Stolen car	Yes	2	1

Table A2. Example tweets used for training (part 2).

Location of Accident	Contributing Factor	Type of Car Involved	Crash Event Type	Driver Error	Collision Type	Case Scenario	Sentiment	Emotions
Caboolture	Hit-and-run	None	Hit-and-run	None	None	Hit-and-run resulting in fatality	Negative	Sadness
Darwin	Alleged hit-and-run	Unknown	Hit-and-Run	Reckless driving	Unknown	Alleged hit-and-run	Negative	Sadness
Freeway in Perth	Medical episode	Unknown	Hit-and-run	None	None	Medical episode leading to fatal hit-and-run	Negative	Sadness

Table A2. *Cont.*

Location of Accident	Contributing Factor	Type of Car Involved	Crash Event Type	Driver Error	Collision Type	Case Scenario	Sentiment	Emotions
Bruce Highway	None	None	Traffic incident	None	None	Stable patients transported to hospital	Neutral	Neutral
Peel St at Swan St	Driver error	None	2-car collision	Yes	None	Person trapped	Neutral	Fear
Beenleigh	None	Two vehicles	Traffic incident	None	None	None	Neutral	Neutral
None	None	None	None	None	None	None	Neutral	None
Countess Street, Brisbane City	Crash into a rail bridge	Truck	Collision with a rail bridge	None	Vehicle-structure collision	Traffic incident	Negative	Fear
Oakey	Stolen car crash	Stolen car	Collision	None	Car crash	Stolen car crash resulting in death and injuries	Negative	Sadness

Appendix B. Prompt Templates for Zero-Shot GPT-4o Mini

```
prompt = """
Given this tweet below:
<tweet>
{text}
</tweet>
{instruction}
"""
road_accident_prompt = """
I want you to classify for this field:
- Road Accident: (Yes or No)
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
severity_prompt ="""
I want you to classify for this field:
- Severity: (none, unknown, mild, moderate, severe, critical, or fatal)
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:
"""
culprit_prompt = """
I want you to extract information for this field:
- Culprit:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:
"""
culprit_appehended_prompt="""
I want you to classify for this field:
- Culprit apprehended: (unknown, Yes or No)
```

```
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:
"""
num_injured_prompt = """I want you to extract information for this field:
- No of injured:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
num_deaths_prompt = """
I want you to extract information for this field:
- No of deaths:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:
"""
location_prompt="""I want you to extract information for this field:
- Location of accident:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
contributing_factor_prompt="""I want you to extract information for this field:
- Contributing factor:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
type_of_car_involved_prompt="""I want you to extract information for this field:
- type of car involved:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
crash_event_type_prompt="""I want you to extract information for this field:
- crash event type:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
driver_error_prompt="""I want you to extract information for this field:
- driver error:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
collision_type_prompt="""I want you to classify for this field:
- collision type: (single-vehicle crashes, 'types of car' accidents, broadside collision, chain reaction car
accidents, hit and run accidents, stationary object collision,
pedestrian accidents or not applicable)
```

```
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
case_scenario_prompt="""I want you to extract information for this field:
- case scenario:
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
sentiment_prompt="""I want you to classify for this field:
- sentiment: (positive, negative or neutral)
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
emotions_prompt="""I want you to classify for this field:
- emotions: (fear, anger, sadness, happy, neutral, disgust, love, confusion, curiosity, gratitude, sympathy or
empathy)
RULES
- You must provide a value for this field above.
- if the tweet does not contain enough information to answer, you must return 'None'.
- Pick the best answer (only one value from the list provided) as the value for this field.
Use the values' spellings as they have been provided you in your response.
- Do not be unnecessarily verbose or make additional statements.
Your response:"""
model_prompt = PromptTemplate.from_template(prompt)
model = ChatOpenAI(model="gpt-4o-mini", temperature=0, streaming=False)#.bind(response_format=
{"type":"json_object"})
chain = model_prompt | model | StrOutputParser()
```

References

1. Sahana, S.; Palaniappan, D.; Bobade, S.D.; Rafi, S.M.; Kannadasan, B.; Jayapandian, N. Deep learning ensemble model for the prediction of traffic accidents using social media data. *J. Pharm. Negat. Results* **2022**, *13*, 485–495. [CrossRef]
2. Jaradat, S.; Alhadidi, T.I.; Ashqar, H.I.; Hossain, A.; Elhenawy, M. Exploring traffic crash narratives in Jordan using text mining analytics. *arXiv* **2024**, arXiv:2406.09438.
3. Gutierrez-Osorio, C.; González, F.A.; Pedraza, C.A. Deep learning ensemble model for the prediction of traffic accidents using social media data. *Computers* **2022**, *11*, 126. [CrossRef]
4. Kumar, K.P.K.; Geethakumari, G. Detecting misinformation in online social networks using cognitive psychology. *Hum. -Centric Comput. Inf. Sci.* **2014**, *4*, 14. [CrossRef]
5. Stieglitz, S.; Mirbabaie, M.; Ross, B.; Neuberger, C. Social media analytics—Challenges in topic discovery, data collection, and data preparation. *Int. J. Inf. Manag.* **2018**, *39*, 156–168. [CrossRef]
6. Atefeh, F.; Khreich, W. A survey of techniques for event detection in Twitter. *Comput. Intell.* **2015**, *31*, 132–164. [CrossRef]
7. Batrinca, B.; Treleaven, P.C. Social media analytics: A survey of techniques, tools and platforms. *AI Soc.* **2015**, *30*, 89–116. [CrossRef]
8. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *arXiv* **2020**, arXiv:2005.14165.
9. Pei, X.; Li, Y.; Xu, C. GPT self-supervision for a better data annotator. *arXiv* **2023**, arXiv:2306.04349.
10. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
11. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models Are Unsupervised Multitask Learners. 2019. Available online: https://openai.com/index/better-language-models/ (accessed on 15 July 2024).
12. Chalapathy, R.; Chawla, S. Deep learning for anomaly detection: A survey. *arXiv* **2019**, arXiv:1901.03407.
13. Kutela, B.; Mwekh'iga, R.J.; Kilaini, A.M.; Magehema, R.T.; Mbatta, G. Leveraging social media data to understand spatial and severity of roadway crashes in Tanzania. *J. Saf. Stud.* **2022**, *7*, 27–51. [CrossRef]
14. Ruder, S. An overview of multitask learning in deep neural networks. *arXiv* **2017**, arXiv:1706.05098.
15. Zhang, Y.; Yang, Q. A survey on multitask learning. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 5586–5609. [CrossRef]

16. Liu, S.; Wang, Z.; Liu, X. Jointly learning multi-task sequences and language models with shared hiddenlLayers. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 7 November 2019; pp. 5939–5948.
17. Bingel, T.; Søgaard, S. Identifying beneficial task relations for multitask learning in deep neural networks. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Spain, 3–7 April 2017; pp. 164–169.
18. Thrun, S.; Pratt, L. *Learning to Learn*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1998.
19. Liu, T.; Ma, X.; Liu, L.; Liu, X.; Zhao, Y.; Hu, N.; Ghafoor, K.Z. LAMBERT: Leveraging Attention Mechanisms to Improve the BERT Fine-Tuning Model for Encrypted Traffic Classification. *Mathematics* **2024**, *12*, 1624. [CrossRef]
20. Zhou, Y.; Li, Z.; Tian, S.; Ni, Y.; Liu, S.; Ye, G.; Chai, H. SilverSight: A multi-task Chinese financial large language model based on adaptive semantic space learning. *arXiv* **2024**, arXiv:2404.04949.
21. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A survey of large language models. *arXiv* **2023**, arXiv:2303.18223.
22. Vaswani, V.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; The MIT Press: Cambridge, MA, USA, 2017.
23. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global vectors for word representation. In Proceedings of the EMNLP, Doha, Qatar, 25–29 October 2014; pp. 1532–1543. [CrossRef]
24. Yang, J.; Jin, H.; Tang, R.; Han, X.; Feng, Q.; Jiang, H.; Yin, B.; Hu, X. Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond. *arXiv* **2023**, arXiv:2304.13712. [CrossRef]
25. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. In Proceedings of the NeurIPS, New Orleans, LA, USA, 28 November–9 December 2022.
26. Ye, X.; Durrett, G. The unreliability of explanations in few-shot prompting. In Proceedings of the NeurIPS, New Orleans, LA, USA, 28 November–9 December 2022.
27. Hasan, S.; Ukkusuri, S.V. Location contexts of user check-ins to model urban geo life-style patterns. *PLoS ONE* **2015**, *10*, e0124819. [CrossRef]
28. Radford, A.; Narasimhan, K.; Saliman, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://openai.com/index/language-unsupervised/ (accessed on 15 July 2024).
29. Ni, M.; He, Q.; Gao, J. Forecasting the subway passenger flow under event occurrences with social media. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1623–1632. [CrossRef]
30. Shirky, C. The political power of social media: Technology, the public sphere, and political change. *Foreign Aff.* **2011**, *90*, 28–41.
31. Ye, Q.; Chen, X.; Ozbay, K.; Li, T. Mining social media data for transport policy: Approaches, challenges, and recommendations. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; IEEE: Piscataway, NJ, USA; pp. 785–794. [CrossRef]
32. Demertzis, K.; Iliadis, L.; Anezakis, V.-D. MOLESTRA: A multitask learning approach for real-time big data analytics. In Proceedings of the 2018 Innovations in Intelligent Systems and Applications (INISTA), Thessaloniki, Greece, 3–5 July 2018; IEEE: Piscataway, NJ, USA. [CrossRef]
33. Wang, G.; Kim, J. The prediction of traffic congestion and incident on urban road networks using Naive Bayes classifier. In Proceedings of the ATRF, Melbourne, Australia, 16–18 November 2016.
34. Liu, X.; He, P.; Chen, W.; Gao, J. Multi-task deep neural networks for natural language understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4487–4496.
35. Zhang, Z.; He, Q.; Zhu, S. Potentials of using social media to infer the longitudinal travel behavior: A sequential model-based clustering method. *Transp. Res. Part C Emerg. Technol.* **2017**, *85*, 396–414. [CrossRef]
36. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
37. D'Andrea, E.; Ducange, P.; Bechini, A.; Renda, A.; Marcelloni, F. Real-time detection of traffic from Twitter stream analysis. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2269–2283. [CrossRef]
38. Mehri, S.; Eskenazi, M. USR: An unsupervised and reference free evaluation metric for dialog generation. *arXiv* **2020**, arXiv:2005.00456.
39. Vishwakarma, M.; Kesswani, N. A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection. *Decis. Anal. J.* **2023**, *7*, 100233. [CrossRef]
40. Liu, Z.; He, S.; Ding, F.; Tan, H.; Liu, Y. Exploring the potential of social media data in interpreting traffic congestion: A case study of Jiangsu Freeways. In Proceedings of the CICTP 2023, Beijing, China, 14–17 July 2023. [CrossRef]
41. Ding, Y.; Tao, H.; Zhang, R.; Cheng, Y.; Wang, H. Social media-based traffic situational awareness under extreme weather. In Proceedings of the CICTP 2023, Beijing, China, 14–17 July 2023. [CrossRef]
42. Yang, X.; Bekoulis, G.; Deligiannis, N. Traffic event detection as a slot filling problem. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106202. [CrossRef]
43. Zheng, O.; Abdel-Aty, M.; Wang, Z.; Ding, S.; Wang, D.; Huang, Y. Avoid: Autonomous vehicle operation incident dataset across the globe. *arXiv* **2023**, arXiv:2303.12889.
44. Jaradat, S.; Nayak, R.; Paz, A.; Elhenawy, M. Ensemble Learning with Pre-Trained Transformers for Crash Severity Classification: A Deep NLP Approach. *Algorithms* **2024**, *17*, 284. [CrossRef]

45. Luceri, L.; Boniardi, E.; Ferrara, E. Leveraging large language models to detect influence campaigns on social media. *arXiv* **2023**, arXiv:2311.07816.
46. Yang, K.; Zhang, T.; Kuang, Z.; Xie, Q.; Huang, J.; Ananiadou, S. MentaLLaMA: Interpretable mental health analysis on social media with large language models. In Proceedings of the ACM Web Conference, Singapore, 13–17 May 2024; pp. 4489–4500. [CrossRef]
47. Kim, S.; Kim, K.; Jo, C.W. Accuracy of a large language model in distinguishing anti- and pro-vaccination messages on social media: The case of human papillomavirus vaccination. *Prev. Med. Rep.* **2024**, *42*, 102723. [CrossRef]
48. Li, M.; Conrad, F. Advancing annotation of stance in social media posts: A comparative analysis of large language models and crowd sourcing. *arXiv* **2024**, arXiv:2406.07483.
49. Xue, H.; Zhang, C.; Liu, C.; Wu, F.; Jin, X. Multi-task prompt words learning for social media content generation. *arXiv* **2024**, arXiv:2407.07771.
50. Liu, J.; Siu, M. Enhancing mental health condition detection on social media through multi-task learning. *medRxiv* **2024**. [CrossRef]
51. Ilias, L.; Askounis, D. Multitask learning for recognizing stress and depression in social media. *arXiv* **2023**, arXiv:2305.18907. [CrossRef]
52. Aduragba, O.T.; Yu, J.; Cristea, A.I. Multi-task learning for personal health mention detection on social media. *arXiv* **2022**, arXiv:2212.05147.
53. Bruns, A.; Burgess, J.; Highfield, T. A 'big data' approach to mapping the Australian Twittersphere. In *Advancing Digital Humanities*; Palgrave Macmillan: London, UK, 2014. [CrossRef]
54. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. In Proceedings of the NAACL-HLT 2018, New Orleans, LA, USA, 1–6 June 2018; pp. 2227–2237. [CrossRef]
55. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
56. Gal-Tzur, A.; Grant-Muller, S.; Kuflik, T.; Minkov, E.; Nocera, S.; Shoor, I. The potential of social media in delivering transport policy objectives. *Transp. Policy* **2014**, *32*, 115–123. [CrossRef]
57. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 785–794. [CrossRef]
58. Rathje, S.; Mirea, D.-M.; Sucholutsky, I.; Marjieh, R.; Robertson, C.E.; Van Bavel, J.J. GPT is an Effective Tool for Multilingual Psychological Text Analysis. *Proc. Natl. Acad. Sci. USA* **2024**, *121*, e2308950121. [CrossRef]
59. Manning, C.D.; Raghavan, P.; Schütze, H. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2008.
60. Lin, C.-Y. ROUGE: A package for automatic evaluation of summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004; Association for Computational Linguistics: Stroudsburg, PA, USA; pp. 74–81.
61. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.-J. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Stroudsburg, PA, USA, 7–12 July 2002; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; pp. 311–318. [CrossRef]
62. Morris, A.C.; Maier, V.; Green, P. From WER and RIL to MER and WIL: Improved evaluation measures for connected speech recognition. In Proceedings of the Interspeech, Jeju Island, Republic of Korea, 4–8 October 2004; pp. 2765–2768.
63. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 16–20 November 2020; Liu, Q., Schlangen, D., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 38–45. [CrossRef]