

Automating Pedestrian Crash Typology Using Transformer Models

Transportation Research Record
1–13

© The Author(s) 2024

Article reuse guidelines:

sagepub.com/journals-permissions

DOI: 10.1177/03611981241260691

journals.sagepub.com/home/trr

Amir Hossein Oliaee¹ , Subasish Das² , and Minh Le³

Abstract

To accurately analyze and understand the causes of traffic crashes involving pedestrians and bicyclists, the Pedestrian and Bicycle Crash Analysis Tool (PBCAT) was developed. However, manual data entry in the tool is labor intensive. Thus, a more automated method is needed for large data sets. This study developed deep-learning models to automate the classification of crash types. Additionally, the PBCAT's classification typology can lead to imbalanced data sets, underscoring the need to actively tackle the issue of imbalanced native classification. By addressing this issue, researchers can significantly enhance their ability to harness the potential of emerging large language models. This endeavor becomes even more crucial as large language models like transformer models become increasingly accessible, offering promising opportunities in transportation safety research. This study focused on police reports' text narratives concerning pedestrian crashes in three major cities in Texas from 2018 to 2020 as a case study. It evaluated the effectiveness of classification loss functions, classification typology adjustments, and model pre-training in addressing the adverse effects of data set imbalance. Our tests indicate that better classification results can be achieved by using the balanced categorical cross entropy (BCE) loss function and using a model with a more robust pre-training. This effect was noticeable when a large enough sample size was present for each class. In the case of smaller data sets, a tiered classification system was recommended, with fewer classes and more distinct text sentiment.

Keywords

artificial intelligence, pedestrian crash, safety, pedestrian crash typing, typology, transformer models

Pedestrian safety is a crucial concern within the field of traffic safety. Because of pedestrian vulnerability, understanding pedestrian-related crashes and finding solutions becomes imperative. Recent years have seen an increase in non-motorized trips, resulting in a rise in injuries among non-motorists, particularly pedestrians. According to the National Highway Traffic Safety Administration (NHTSA), there was a noteworthy increase in pedestrian fatalities in traffic crashes in 2021, claiming 7,388 lives and representing a 12.5% rise compared with the 6,565 fatalities recorded in 2020 (1, 2). This was the highest number of pedestrian deaths since 1981 when 7,837 pedestrians lost their lives in traffic crashes. Most pedestrian deaths occurred in urban areas (84%) rather than rural areas (16%). This data necessitates further investigation into pedestrian crashes.

There has been a growing focus on pedestrian-related crashes, leading to using tools like the Pedestrian and Bicycle Crash Analysis Tool (PBCAT) to gather data related to crashes involving pedestrians or bicyclists with

motor vehicles. However, crash typing, a significant step in this process, is time-consuming. To address this issue, this study developed a crash-typing framework using `ktrain` and `Transformers` Python libraries for natural language processing (NLP) to streamline the process, leveraging the compatibility of police report narration style and data collection practices to enhance the assessment and deployment of the framework. This study utilized two open-source language models for classifying pedestrian maneuver types. However, the technical insight and conclusions apply to models with large language model (LLM) backbones, which we expect to enhance the analysis of unstructured crash narratives. By classifying

¹College of Architecture, Texas A&M University, College Station, TX²Ingram School of Engineering, Texas State University, San Marcos, TX³Texas A&M Transportation Institute, Dallas, TX

Corresponding Author:

Subasish Das, subasish@txstate.edu

pedestrian maneuver types, these models will help researchers better understand the circumstances leading to crashes, enabling the development of more effective strategies to improve pedestrian safety.

The rest of the paper is organized as follows: review of related work; data preparation and descriptive statistics; explanation of the PBCAT Version 3.0 crash-typing approach and concepts of NLP-related machine-learning algorithms; report of results and findings; and study conclusion with remarks on future research.

Literature Review

Pedestrian safety has been extensively studied, leading to a deeper comprehension of the factors influencing accidents. This study reviews pertinent literature to address existing gaps in research and practices that require attention, particularly considering the evolving landscape of pedestrian safety with the introduction of e-scooters.

In 2000, the Federal Highway Administration (FHWA) and the NHTSA launched the PBCAT software, designed to aid in the analysis of non-motorist crashes and facilitate the identification of solutions and countermeasures (3). PBCAT was developed to describe the actions leading to traffic crashes between vehicles and pedestrians or bicyclists, allowing for a better understanding of event sequences and contributing efforts. The most recent version, PBCAT Version 3.0, has been updated to provide traffic safety professionals with enhanced knowledge of causation patterns, enabling the design of more effective interventions. This tool offers pre-drawn diagrams and drop-down menus, facilitating the collection of critical information from police reports about pedestrian/bicycle crashes (4). PBCAT outcomes can create imbalanced data by crash severity counts (3, 4). Despite the progress made with PBCAT, crash-narrative insufficiency remains a concern, prompting exploration into the effectiveness of current narrations and the use of language models to improve understanding and comprehension. The relevance of PBCAT-related analysis has grown in the transportation safety community, with various studies using this methodology to examine pedestrian and bicyclist crash data. Shah et al. (5) investigated e-scooter and bicycle crashes, uncovering significant differences in multiple factors. Lopez et al. (4) evaluated the data quality of text narratives in police reports on bicycle crashes, revealing a high degree of missing information. Schneider and Stefanich (6) introduced the location-movement classification method (LMCM) and demonstrated its usefulness in providing insights not captured by the PBCAT. Chavis et al. (7) analyzed pedestrian and bicycle crashes in Washington D.C. and identified NHTSA crash groups, examining relevant countermeasures.

Text mining has been applied in many transportation-safety-related studies (8–19). The introduction of the transformer architecture by Vaswani et al. (20) revolutionized the field of NLP and led to significant improvements over previous state-of-the-art networks. However, before the advent of transformer-based models, achieving human-level language processing capabilities seemed beyond the reach of computers. In this study, we focus on two transformer-based models sharing the same architecture to assess their impact on the classification quality of imbalanced data sets, a common challenge encountered in transportation safety research (21–23). Because of the limited details in state or national crash databases, crash narratives play a crucial role in providing deeper insights into factors contributing to crash occurrences (24). However, manually identifying PBCAT-coded crash types from these extensive textual data sets is labor intensive. Thus, text mining analytics offer a solution to extract insights from crash-narrative textual data efficiently (25). Various applications, such as thematic analysis, content analysis, supervised and unsupervised modeling, and NLP, have been utilized. Machine-learning models like XGBoost and BERT have successfully classified crash types and severity, demonstrating their promise in handling large language models for more accurate and efficient analysis (21–23, 26, 27). Researchers have also combined crash narratives with metadata to discern prevalent themes contributing to crash incidents (28). Overall, large language models and text mining analytics play a vital role in improving pedestrian safety research, enabling a more comprehensive understanding of crash factors and aiding in the development of effective interventions to reduce pedestrian-related crashes.

BERT (29) is one of the most impactful natural language models today, accelerating NLP research and popularizing transfer learning in this field. It stands for Bidirectional Encoder Representations from Transformers. Unlike its preceding models, such as Word2Vec (30) and GloVe (31), which used word-level embeddings, BERT uses word pieces, making it more capable of handling out-of-vocabulary words. The RoBERTa model (32) is a more robustly optimized version of BERT, demonstrating a performance that is close to or better than that model. Most of RoBERTa model's improvement is derived from the modifications that Liu et al. (32) made to the BERT's training process. These modifications included training for a longer duration, using larger batch sizes, and utilizing a larger amount of training data (CC-NEWS data set in addition to other English-language corpora).

Study Objective and Contribution

This study contributes to the field by addressing several gaps in existing approaches to pedestrian crash typing.

First, it explores the limitations and gaps in current methods for classifying pedestrian crashes, shedding light on areas requiring further improvement. By identifying these gaps, the study lays the groundwork for future research and the development of pedestrian safety. Second, the study focuses on the significant issue of imbalanced data sets for crash typing. By testing several loss functions that aim to increase classification performance, we can assess their effectiveness in classifying crash narratives. We elaborate on how smaller data sets can be more effectively used for developing NLP solutions. By addressing the issue of data set imbalance, the study demonstrates the potential for improved classification accuracy and balance. This refinement of LLM algorithms opens new avenues for effectively categorizing pedestrian maneuvers and understanding the sequence of events leading to pedestrian crashes. Furthermore, the study goes beyond simply using LLMs for crash typing and delves into the automation of the PBCAT classification methodology. By investigating alternatives for automating PBCAT, the study offers insights into streamlining the analysis process, making it more efficient and reliable. These insights contribute to the development of automated tools that can assist safety engineers and policymakers in identifying factors contributing to pedestrian crashes and formulating effective countermeasures.

Methodology

Data Preparation

In this research, pedestrian crash data from 2018 to 2020 was collected from the Texas Department of Transportation's crash record information system (CRIS) for five major cities in Texas: Austin, Dallas, Fort Worth, Houston, and San Antonio. The study specifically focused on the three major cities with the most KAB (K = fatal, A = serious injury, B = moderate injury) crashes per 100,000 population, namely Austin, Dallas, and San Antonio, as shown in Table 1. In 3-year average data, Dallas shows the most crashes per 100,000 population in both KABCO (K = fatal, A = serious injury, B = moderate injury, C = minor injury, and O = no injury) and KAB crashes.

PBCAT Crash Typing

A multiclass classification task focusing on pedestrian maneuvers/actions was undertaken. The transfer learning approach was employed, utilizing the pre-trained BERT (29) and RoBERTa (32) models. The Transformers open-source library (33) and the ktrain low code Python library (34) for deep learning facilitated the development of the Pedestrian Maneuver Classification Framework. After data preparation and initial preprocessing, the

narrative data was converted into embeddings suitable for each model using the encode function of the Transformer library's tokenizer. Each pre-trained model was then expanded by incorporating a linear transformation layer representing the one-hot-encoded representation of Non-Motorist Maneuver classes (CL: crossing path from motorist's left; CR: crossing path from motorist's right; CU: crossing path, unknown direction; PO: parallel path, opposite direction; ST: stationary; PU: parallel path, unknown direction; PS: parallel path, same direction; and OT: other). Hyperparameters were adjusted based on learning-rate range tests (35). Subsequently, the models underwent training, validation, and testing, with the final evaluation of performance conducted based on the test and validation subsets.

Narrative Data Preprocessing

A point of significance for the automation of crash-narrative classification is the amount of time and effort required to perform this task. This issue persists in collecting and producing data sets for developing such automation tools. While we could not identify an optimal sample size in the existing literature for the analysis of such a tool, knowing a larger sample size would be more representative of the overall population, we aimed to develop the largest data set our constraints permitted. We maintained a high standard throughout the manual classification process which was necessary for reliable measurement of model performance. As a result, this study opted to randomly select 1,000 crash reports from a collection of 4,442 crash reports we could access to undergo manual classification using the PBCAT 3.0 crash typology. Subsequently, an early analysis of the distribution of classes revealed that some of the Non-Motorist Maneuver classes represented a small portion of the 1,000-training data set, with "Unknown" (UN) representing 37 samples, "Moving in Unknown Path/Direction" (MU) representing 12 samples, "Non-motorist Fall or Crash" (NA) representing six samples, and "Other/Unusual" (OU) representing 20 samples. In the context of this research, we refer to the resulting classification as the "Detailed" classification. As a result, we combined these classes under the umbrella term "Other" (OT) in tests. While the PBCAT 3.0 has established a standardized protocol for crash type classification, enhancing data integration and streamlining the process, the imbalance in the resulting data can make it difficult to develop automated solutions, especially with smaller sample sizes. By merging smaller yet conceptually adjacent classes, the effects of data set imbalance and size can be mitigated. So, we also tested a classification referred to in this paper as the "Abstract" classification containing, "Crossing" (C), "Parallel" (P), "Stationary"

Table 1. Pedestrian Injuries by City

City	2018–2020					3-year average	
	KABCO	KAB	2020 pop.	KABCO/100k pop.	KAB/100k pop.	KABCO/100k pop.	KAB/100k pop.
Austin	1052	759	961,855	109.37	78.91	36.46	26.30
Dallas	1644	1112	1,304,379	126.04	85.25	42.01	28.42
Houston	2712	1553	2,304,580	117.68	67.39	39.23	22.46
San Antonio	1744	1132	1,434,625	121.56	78.91	40.52	26.30
Fort Worth	671	424	918,915	73.02	46.14	24.34	15.38
Total/Mean*	7,823	4,980	6,924,354	112.98	71.92	37.66	23.97

Note: KAB = fatal, serious, or moderate injury; KABCO = fatal, serious, moderate, minor, or no injury; Pop. = population; * Total values are provided for columns 2–4 and mean values are provided for columns 5–8.

Table 2. Training Data Set by Crash Type

Vehicle/pedestrian	CR	CL	CU	PS	PO	PU	ST	MU	OU	UN	FC	Total
Detailed	CR	CL	CU	PS	PO	PU	ST					NA
Abstract		C			P		S			O		NA
Straight	209	215	99	46	11	20	81	9	11	30	na	731
Right	29	7	4	15	5	5	3	na	na	na	na	68
Left	5	6	6	29	45	18	4	na	1	2	na	116
Parked	na	na	1	na	na	na	na	na	2	na	na	3
Entering	3	3	3	na	na	na	4	na	na	na	na	13
Backing	1	1	1	na	na	na	10	1	4	1	na	19
Other	1	2	2	1	1	4	17	na	2	2	na	32
Unknown	na	na	2	na	na	na	2	2	na	2	na	8
Non-collision	na	na	na	na	na	na	na	na	na	na	8	8
Total	248	234	118	91	62	47	121	12	20	37	8	998

Note: C = crossing; P = parallel; S = stationary; O = other; CR = crossing path from motorist's right; CL = crossing path from motorist's left; CU = crossing path, unknown direction; na = not available; NA = not applicable; PS = parallel path, same direction; PO = parallel path, opposite direction; PU = parallel path, unknown direction; ST = stationary; MU = moving in unknown path/direction; OU = other/unusual; UN = unknown; FC = non-motorist fall or crash.

(S), and “Other” (O) classes. Table 2 provides a detailed breakdown of our pedestrian maneuver data using PBCAT 3.0. Note that the crash types could not be determined for four crashes because two did not involve a pedestrian. For the purposes of this study, they were classified as “OT” and “O” as Detailed and Abstract classifications, respectively.

Addressing the challenges specific to this task was crucial. Crash reports present a unique narration style that poses difficulties both for laypersons to comprehend and for an NLP model without prior exposure to such narration to be fine-tuned. Certain sections of text were often replaced with placeholders like “[retracted]”, or sequences of symbols such as “*” or “#” frequently appeared, causing disruptions in the sentence flow. To improve text coherence, it was necessary to remove numerical information related to blood alcohol level, street addresses, dates and times, and case numbers. Additionally, the lack of specific references to pedestrians using terms like “pedestrian” or “non-motorist” contributed to text fragmentation, as parties involved in crashes were often referred to as “unit 1” or “unit #1”, regardless of whether they were

motorists or non-motorists. Furthermore, Non-ASCII (American Standard Code for Information Interchange) characters were also observed in the texts, necessitating appropriate handling to ensure accurate analysis.

As mentioned earlier, we took a transfer learning approach to benefit from the model's existing knowledge which was gained by being trained on a large corpus of natural text. We adopted a minimalist approach to maintain the natural sentence structure as much as possible and keep words within their context. Our approach involved removing Non-ASCII characters, redacted sections, and sequences of repeating characters (e.g., “#”). Additionally, we filtered out numbers longer than one character and repeated space characters. Empty parentheses and brackets that no longer contained text after cleaning were also filtered out. As a result, the narratives retained their original sentence structure. This approach ensured the optimal performance of BERT and RoBERTa by maintaining the contextual integrity of the text.

To fine-tune the three NLP models for motorist maneuver classification, we partitioned our data set into three sections: train, validation, and test. These sections were

randomly sampled to ensure an equal distribution of classes. The train set, consisting of 70% of the samples, was utilized for fine-tuning the models. The validation data, accounting for 20% of the data set, served to assess the models for overfitting, a phenomenon where models excel on familiar data but may suffer in general performance. Last, we employed the test set, comprising the remaining 10% of the data set, to comprehensively evaluate various aspects of the model's overall performance.

Natural Language Processing Models

Transformer-based large language models such as BERT and RoBERTa process input text as tokens, which can be complete words, word segments, or characters, and convert them into embeddings through their initial layer. Each token is mapped to the model's internal vocabulary, represented as a sequence of numbers. The specific models we used in this research come in two variants, BASE and LARGE, with input sequence length limits of 512 and 768 embeddings, respectively. As our data set's embedding sequence length fell below the 512 limits of the BASE version, we utilized this variant for our tests.

- **BERT:** BERT (29) model takes a bidirectional approach to language representation, meaning its comprehension of words, or word embeddings, is informed by the context of the text following and preceding it. This model was originally trained for two tasks. The mask language modeling task (36) involves masking a portion of text (represented as word embeddings) and training the model to predict the masked word. The next-sentence prediction task involves training the model to comprehend the relationship between two sentences. It achieves this by learning to identify whether a given sentence is followed by another specific sentence.
- **RoBERTa:** The RoBERTa model (32) has been solely trained for the masked language modeling objective by applying a dynamically changing masking pattern to the training data in contrast to BERT, which has also been trained for next-sentence prediction. It is worth mentioning that RoBERTa's extensive vocabulary, which it has gained because of its larger amount of training data, can help it better capture some linguistic nuances.

Experimental Settings

In our initial data analysis, we identified that the data set was noticeably imbalanced. In a previous study on using BERT to classify crash severity types, researchers

identified the data set imbalance as an issue. They recommended further exploration of augmentation methods and hyperparameter adjustments as possible avenues for improving the model's performance (21). We tested two batch sizes of 32 and 64 to perform a fine-tuning simulation for five iterations (epochs) in which we linearly increased the learning rate. This allowed us to identify a suitable starting learning rate for fine-tuning each model. To reduce the effects of imbalance on our model's performance, we tested the categorical cross entropy (CCE) (37), balanced categorical cross entropy (BCE), and focal categorical cross entropy (FCE) (38) loss functions in fine-tuning our models. We used the Adam optimizer (39) and a cyclical learning-rate policy (35) to train the models. The loss values are calculated as follows:

$$\text{CCE} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C 1_{y_i \in C_j} \log_{p_{\text{model}}[y_i \in C_j]} \quad (1)$$

$$\text{BCE} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C 1_{y_i \in C_j} w_j \log_{p_{\text{model}}[y_i \in C_j]} \quad (2)$$

$$\text{FCE} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C 1_{y_i \in C_j} \alpha (1 - p_{\text{model}}[y_i \in C_j])^\gamma \log_{p_{\text{model}}[y_i \in C_j]} \quad (3)$$

where N and C are the number of samples and categories and $p_{\text{model}}[y_i \in C_j]$ represents the model's predicted probability of the sample of index i belonging to the class of index j . The balancing weight represented by w_j is the fraction of the data set that does not belong to the j th class. The focusing parameter is represented as γ (α and γ were set to 0.25 and 2 based on Lin et al.'s (38) experimental results).

Results and Findings

Performance Metrics

In our assessment of models, we considered several performance metrics. The accuracy represents the portion of the models' predictions that are correct. However, this metric does not discriminate between the models' predictions of the minority or the majority classes. In data sets with noticeable imbalances, the prediction of the smaller classes affects this metric the least, even though models struggle the most to predict the underrepresented classes correctly. We consider Precision as a measure of the accuracy of positive predictions and Recall as the measure of the models' ability to identify positive instances correctly. As a result, the macro-averaged F1, the harmonic mean of the Precision and Recall, acts as the key performance indicator in our research.

$$\text{Average Precision} = \frac{\sum_{i=1}^C \frac{TP_i}{TP_i + FP_i}}{C} \quad (4)$$

$$\text{Average Recall} = \frac{\sum_{i=1}^C \frac{TP_i}{TP_i + TN_i}}{C} \quad (5)$$

$$\text{Average F1} = \sum_{i=1}^C \frac{2 TP_i}{2 TP_i + FP_i + FN_i} \quad (6)$$

where C represents the number of classes. TP_i and TN_i are the correct predictions of sample belonging and not belonging to the i th class. FP_i and FN_i represent the false predictions of sample belonging and not belonging to the i th class.

Our assessments also consider the area under receiver operating characteristics curve (AUC). The receiver operating characteristics (ROC) curve displays the relationship between a model's true positive rate (TPR) and false positive rate (FPR). Considering that the ROC curve is independent of classification thresholds, it might not be the primary performance metric in multiclass classification problems (40, 41). However, by comparing one class versus the rest of the classes, it can provide insights into a model's performance. To better evaluate the model's performance in the context of an imbalanced classification problem, we also consider the geometric mean (G-mean) in our evaluations. G-mean is a measure of central tendency. It uses the product of a set of numbers to represent the typical value of that set. This metric gives more weight to the performance of minority classes, making it a great auxiliary to F1 as a metric for the model's overall performance.

$$TPR_i = \frac{TP_i}{TP_i + FN_i} \quad (7)$$

$$FPR_i = \frac{FP_i}{FP_i + TN_i} \quad (8)$$

$$AUC_i = \frac{Sp_i - Np_i (Nn_i + 1) / 2}{Np_i Nn_i} \quad (9)$$

$$G - \text{mean} = \sqrt{\frac{\sum_{i=1}^C TPR_i \times FPR_i}{C}} \quad (10)$$

Here, Sp_i is the ranked sum of all positive samples for the class i , while Np_i and Nn_i denote the number of positive and negative samples for the class, respectively.

Results

We started our tests by performing a fine-tuning simulation for five epochs using batch sizes of 32 and 64 in combination with several loss functions. These loss functions included CCE, which is one of the most widely used loss functions for multiclass tasks; BCE, which applies

weights to the loss value of each class inversely proportional to their frequency in the data set; and FCE, which has proven to be highly effective in encouraging the model to focus on the more challenging samples in the data set (38). In our learning-rate tests, some hyperparameter combinations did not prove a noticeable decrease in loss values. Figure 1 demonstrates a hyperparameter combination that did not prove a noticeable descent in loss (left) and a combination that shows a point of steepest descent marked by an orange dot and a point of minimum gradient marked by a purple dot (right). As a result, we selected a learning rate that fell in the shared range between these two points in their hyperparameter combinations that demonstrated these points.

The fine-tuning process is carried out through epochs in which model weights are adjusted to better perform classifications on the training data subset. In our fine-tuning, the process was initiated with the pre-trained model parameters. The loss value was calculated for training data in each iteration, and the Adam optimizer was used to adjust the model parameters. At the end of each iteration, the loss was calculated for the validation data, and a snapshot of model parameters was stored alongside the loss and accuracy values. To prevent overfitting, in which the model would specialize in classifying training data at the cost of its general performance, we selected the model parameters from the epoch that demonstrated the lowest validation loss value. Figure 2 compares the loss values of the two best-performing models: RoBERTa and BERT. This graph marks the point in the fine-tuning process that achieved the lowest validation loss. We used the model parameters from this stage of the training for our evaluations. The corresponding model accuracies are shown in Figure 3 and demonstrate the less performant model's stagnant accuracy compared with the better-fine-tuned models.

In our tests using the detailed classification system with eight classes (shown in Figure 4), we did not observe a significant distinction between BERT and RoBERTa. Even though RoBERTa achieved the highest accuracy in some tests and the highest F1 in other tests, both were not achieved simultaneously. However, in the test performed using the abstract classification with four classes (shown in Figure 5), all the fine-tuned RoBERTa models outperformed BERT models in accuracy and F1. This informed us that a more accurate and balanced classification performance can be achieved by increasing the data allocated to each class by combining conceptually adjacent classes. Considering that the two models share the same architecture, it is evident that with this change in the classification task, the model's pre-trained knowledge also gains a more prominent role in the fine-tuned model's performance. This effect was also observed with the classification loss. Even though the detailed classification

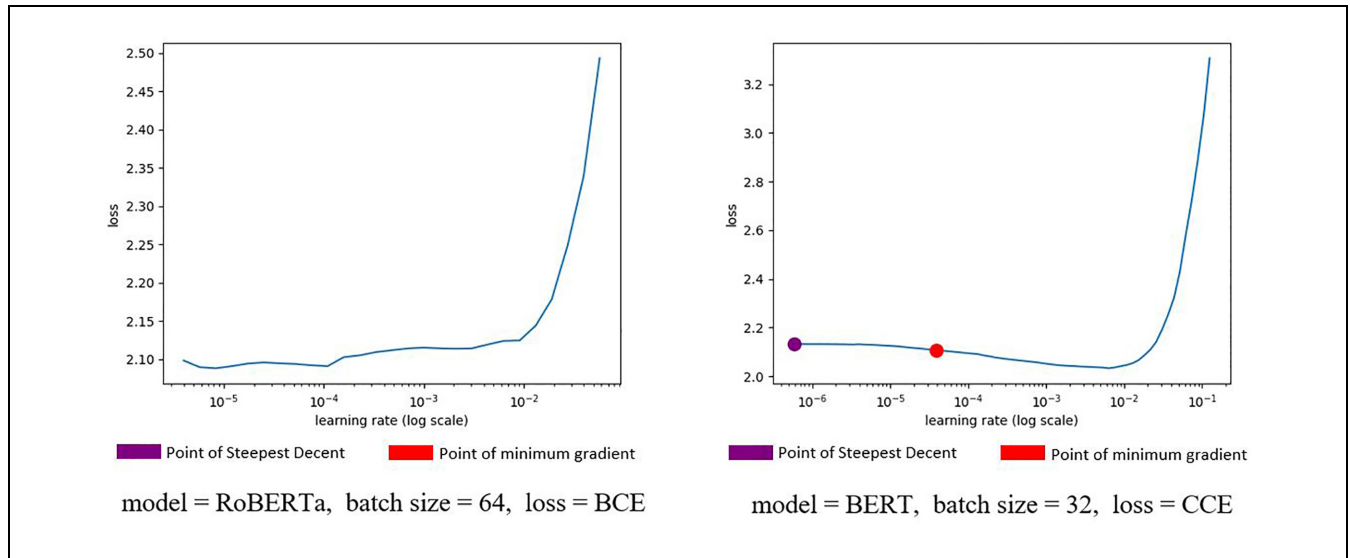


Figure 1. Examples of fine-tuning simulation for learning-rate search. (Color online only.)

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy.

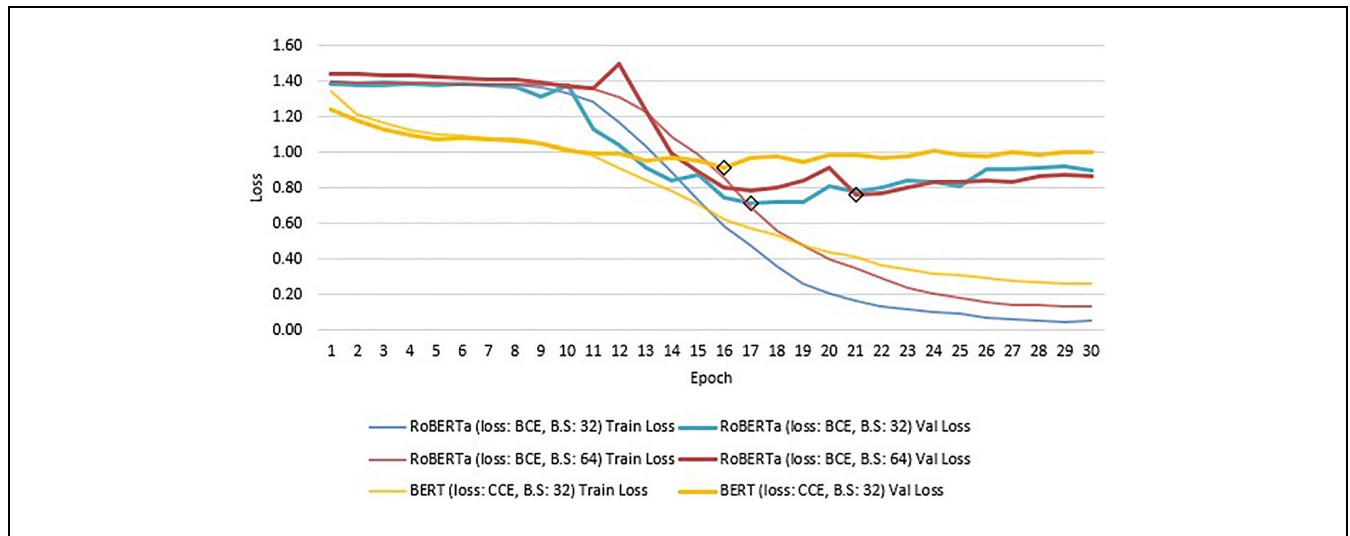


Figure 2. Comparison of loss values.

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; B.S = batch size.

of the BCE and FCE did not achieve the intended goal of a more balanced classification (as evidenced by the lower F1 scores), RoBERTa models that were fine-tuned using the BCE loss achieved the most balanced classification results in our abstract classification tests.

Even though data imbalance has a notable impact on the classifier's performance, a close look at the confusion matrices of the best-performing RoBERTa and BERT models (Figure 6 and Table 3) shows the impact of the class sentiment on the model's classification ability. For instance, even though "S" is a smaller class than "P", it is

better classified by both models. This is most likely because "P" represents the pedestrian's direction of movement, which can be conflated with the motorist's direction. However, the collaboration between a stationary vehicle and a moving pedestrian is less likely to occur, and the model may learn to classify the lack of movement in the narrative instead of specifically the pedestrian's lack of movement. The RoBERTa's confusion matrix shows that with better fine-tuning focusing on a more balanced classification, the model's performance can significantly improve for the more challenging classes.

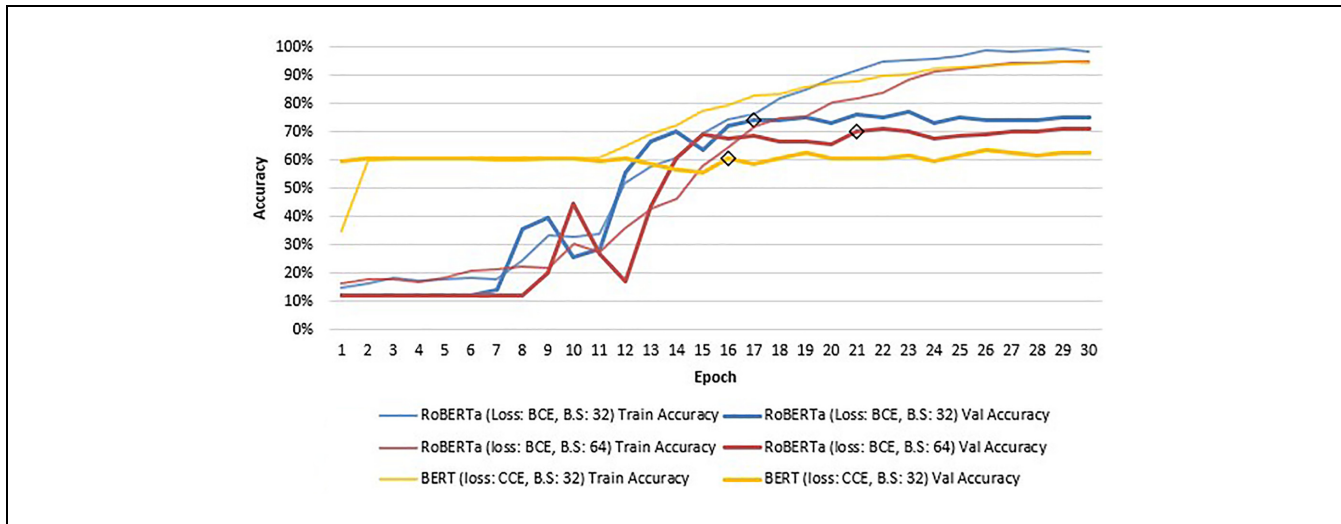


Figure 3. Comparison of accuracies.

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; B.S = batch size.

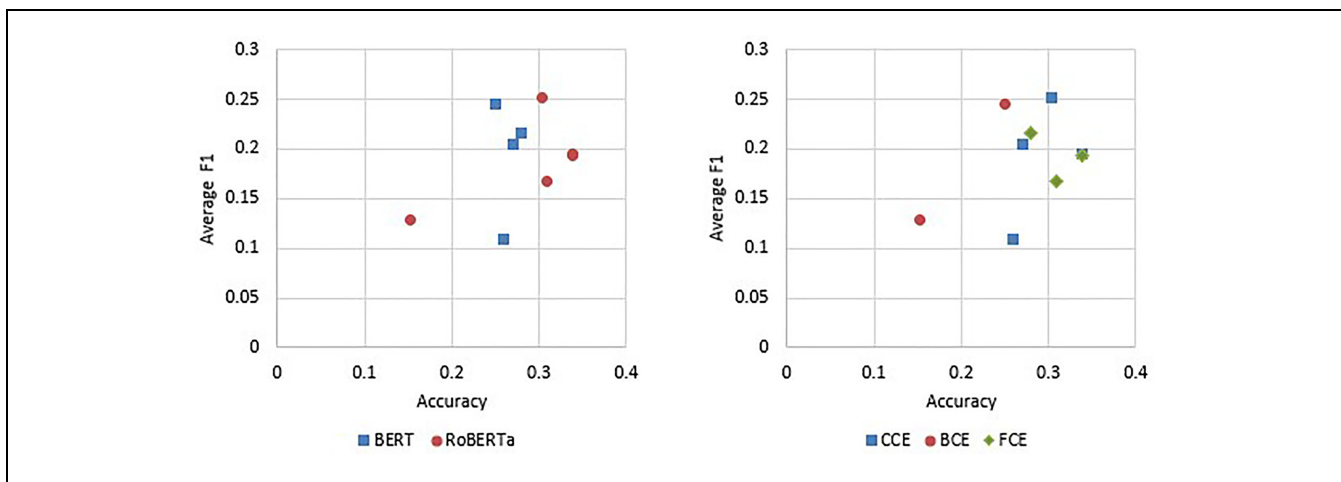


Figure 4. Detailed classification results (using eight classes).

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; FCE = focal categorical cross entropy.

In contrast to the previous performance metrics discussed earlier, which are based on the model's predictions, the ROC is produced based on the model's predicted probability for a sample belonging to each class. Figure 7 compares the two best-performing RoBERTa models and BERT. The area under ROC curves (AUC) for the RoBERTa model was higher in both the validation and test. The higher AUC indicates that this model demonstrates better discriminative power for all classes.

Based on the previous analysis, it can be concluded that the framework can only perform less complicated classification tasks with high accuracy because of the limited available data and the inherent imbalance between crash types. Beyond the overall accuracy, the macro-

averaged F-measure could be considered a stopping criterion for future research. Because of the observed importance of maintaining the narrative structure in the preprocessing stage, the exploration of a uniform narration guideline and its effects on classification performance is recommended for future research.

Our tests indicate that with a large enough sample size or fewer classes, the model's pre-training can have a noticeable impact when developing fine-tuned classification models. The more robustly pre-trained model (RoBERTa) was more accurate and balanced in its prediction. The balanced categorical cross entropy (BCE) could similarly improve the model's performance. The G-mean values (Table 3) confirm that classification was

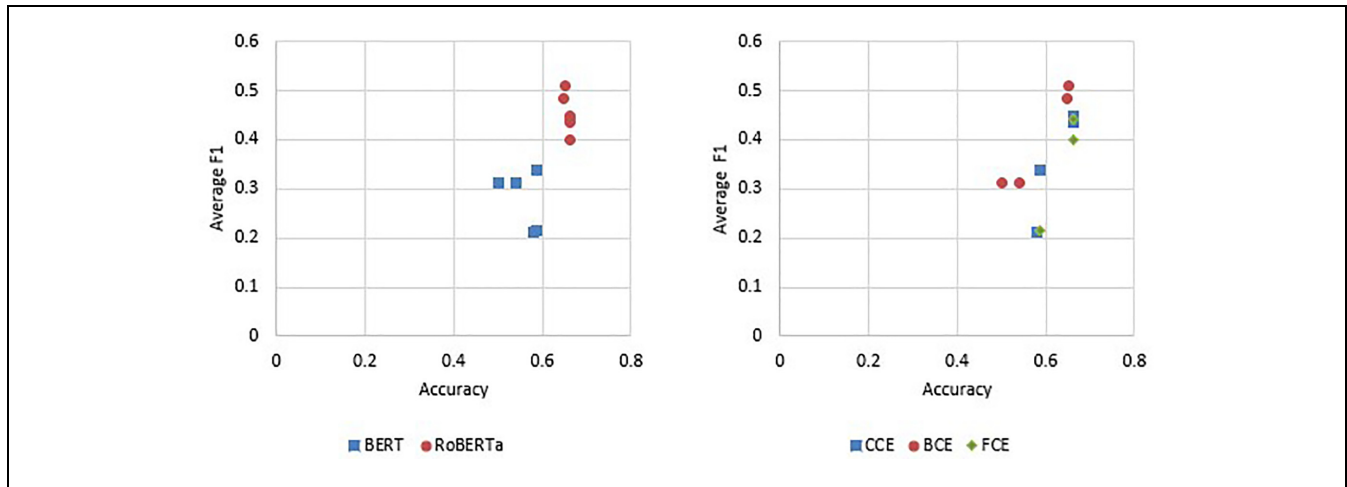


Figure 5. Abstract classification results (using four classes).

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; FCE = focal categorical cross entropy.

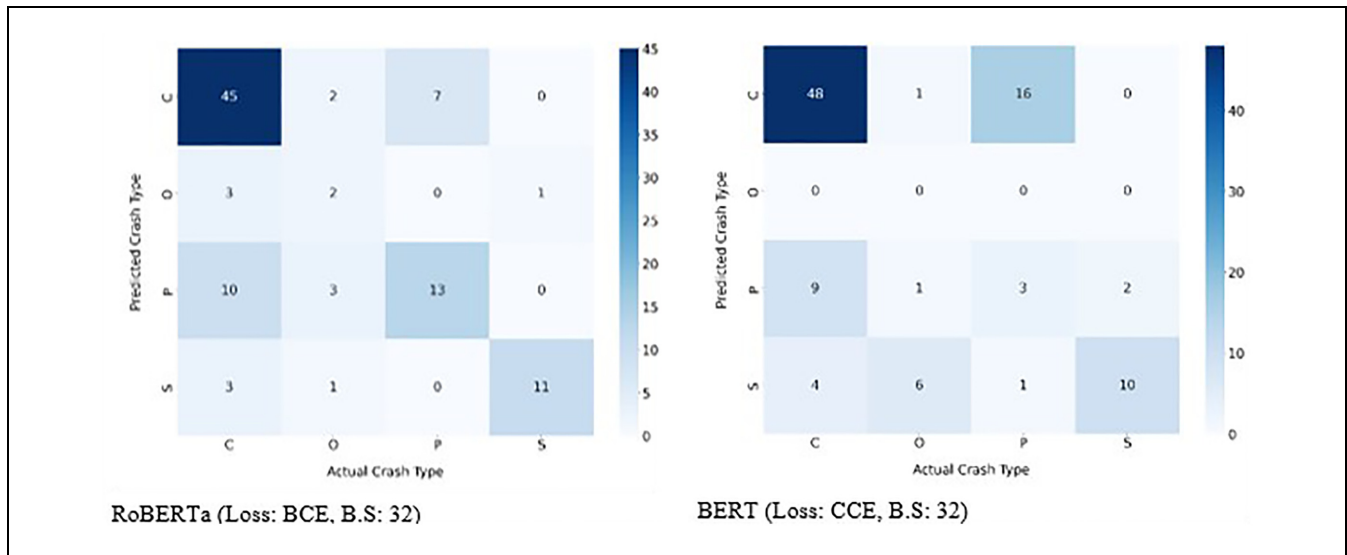


Figure 6. Test confusion matrices of the best-performing BERT and RoBERTa models.

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; B.S = batch size.

consistently more balanced when the BCE loss function was used. Even though we did not observe noticeable improvements using the FCE loss with the parameters that resulted in improved results in Lin et al.'s (38) study, we cannot say with certainty that through further experimentation such results cannot be achieved. However, the FCE loss function introduces two additional hyperparameters to the classification task, which can be computationally taxing.

The classification and class sentiments play a significant role, and model improvement and fine-tuning techniques help reduce the barrier for the model to learn the classification task. We expect an increasingly more

balanced classification (despite data imbalance) with a larger data set by using the BCE loss and a more robustly pertained model.

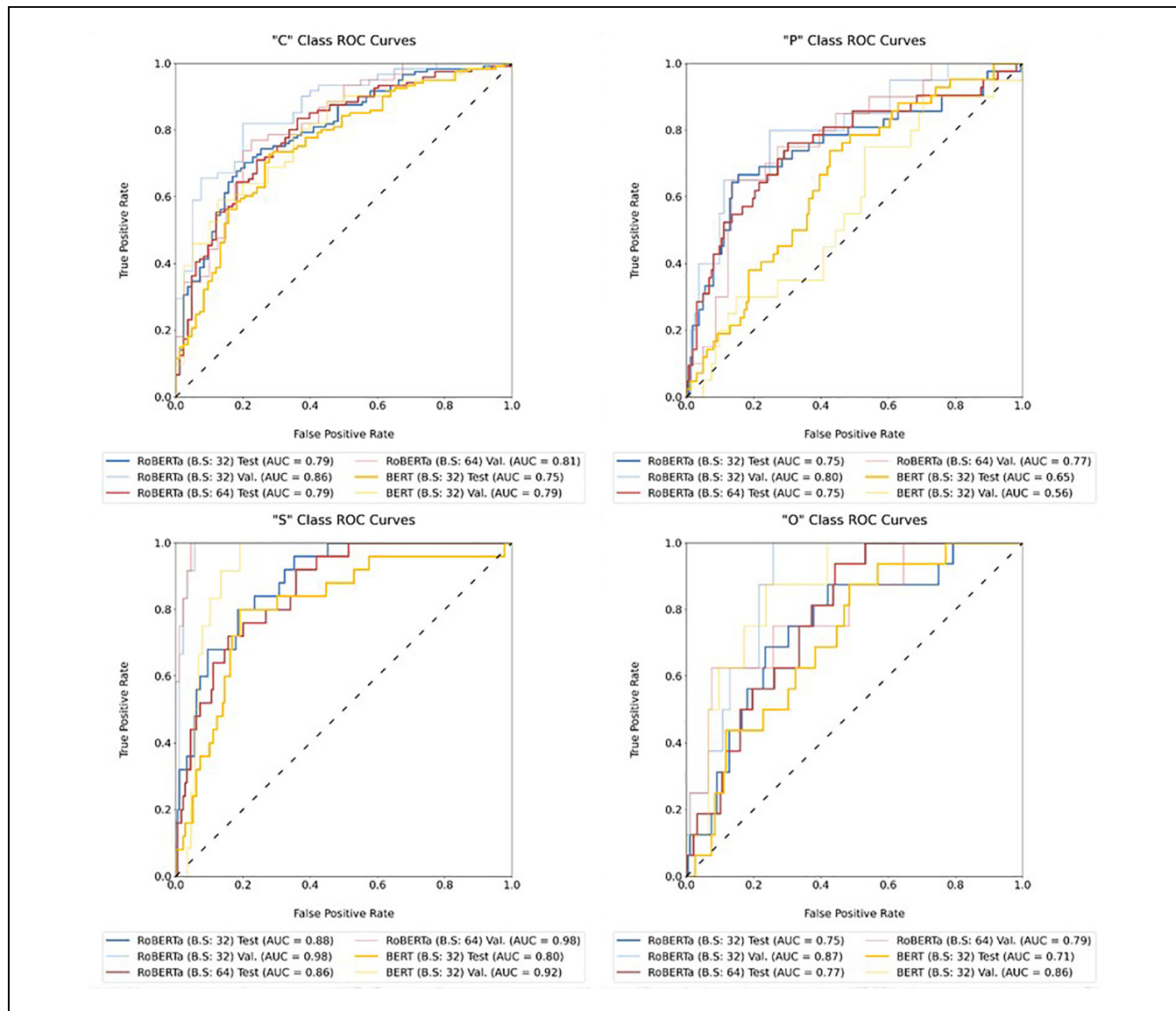
Conclusions

An abundance of crash narratives exists in various crash databases across the United States. However, like other unstructured information, these crash-narrative data provide little to no insight and cannot be readily utilized in decision-making processes. The PBCAT has offered a potential solution by incorporating pre-drawn diagrams and drop-down menus to efficiently organize and index

Table 3. Classification Performance Tests Using Four Classes

Model	Batch size	Learning rate	Loss function	Accuracy, %	Average F1, %	Average recall, %	Average precision, %	G-mean, %
RoBERTa	64	1.00e-05	BCE	64.7	48.5	48.4	50.3	86.4
RoBERTa	32	1.00e-05	BCE	65.2	50.9	51.8	50.7	85.8
RoBERTa	64	1.00e-05	CCE	66.2	43.5	44.8	43.3	75.6
RoBERTa	32	1.00e-05	CCE	66.2	44.8	43.4	53.7	73.8
RoBERTa	64	1.00e-05	FCE	66.2	44.1	46.4	43.4	72.3
RoBERTa	32	1.00e-05	FCE	66.2	40.0	41.7	46.8	67.0
BERT	64	1.00e-05	BCE	50.0	31.2	31.3	31.3	82.8
BERT	32	1.00e-05	BCE	53.9	31.2	30.8	34.1	82.7
BERT	64	1.00e-05	CCE	57.8	21.2	25.5	19.6	50.8
BERT	32	1.00e-05	CCE	58.8	33.9	34.6	34.0	70.6
BERT	64	1.00e-05	FCE	58.8	21.5	26.0	25.6	59.9

Note: BCE = balanced categorical cross entropy; CCE = categorical cross entropy; FCE = focal categorical cross entropy; G-mean = geometric mean.

**Figure 7.** Comparison of the best-performing BERT and RoBERTa models.

Note: ROC = receiver operating characteristics; AUC = area under ROC curve; B.S = batch size.

this unstructured crash-narrative data. By combining PBCAT crash typing with textual data mining techniques, a new avenue emerges to automatically process these crash narratives, transforming them from unstructured to structured data and extracting valuable crash attributing information.

We addressed data imbalance issues commonly encountered in transportation data classification. We tested two classification systems based on PBCAT 3.0 and three loss functions. Increasing the data set size or choosing a smaller classification system with more distinguishable sentiments improved the model's performance. However, it is a given that a smaller classification system would increase the downstream manual efforts. Also, the ability to distinguish between classes of similar sentiment is most desirable as it would preserve the richness of the information derived from the narratives. We achieved further improvements by using the BCE loss function alongside a more robustly pre-trained model. The BCE significantly improved the model's sensitivity to minority classes and increased the model's robustness as evidenced by the geometric mean values. Compared with a previous study, which achieved 19.6% accuracy and a 12.5% F1 score, our detailed classification system yielded 25% accuracy and a 24.6% F1 score. However, by utilizing a more abstract classification system, the RoBERTa model, and the BCE loss function, we achieved a 65% accuracy and a 51% F1 score.

This research revealed some challenges in developing more complex crash typing models. The limited size of the data sets, inconsistent narration styles, and extreme data imbalance are identified as significant factors hindering progress. To address these issues, future research is recommended to focus on utilizing large language models with greater exposure to crash narratives or similar reports during their pre-training stage. This exposure will enhance the model's understanding of the nuances and complexities present in the crash narratives, leading to more accurate and reliable results. Furthermore, adopting cross-validation evaluation methods in future studies is suggested to mitigate inconsistencies observed in evaluating multiclass tasks, especially when a small sample can be used for validation. Cross-validation helps ensure that the model's performance is thoroughly assessed and not overly influenced by random variations in the data-splitting process.

Through our analysis, we observed variations and inconsistencies in the quality of text narratives, which may raise concerns about the reliability and completeness of the data. Previous studies have highlighted a high degree of missing information in crash narratives, indicating a potential challenge in leveraging this data for decision-making purposes. Therefore, it is crucial for law enforcement agencies to consider implementing measures

to improve data collection practices. We recommend providing clear guidelines and training for police officers to ensure standardized reporting practices, including the inclusion of specific details and the adoption of consistent narrative styles. By enhancing data collection processes, agencies can enhance the overall quality and usefulness of crash-narrative data for transportation safety analysis. Our integration of PBCAT crash typing and textual data mining techniques offers a promising avenue for addressing these data quality concerns, enabling the extraction of valuable information from otherwise unstructured narratives. Moving forward, efforts to improve data collection practices and enhance the integration of automated analysis tools will be essential for unlocking the full potential of crash-narrative data in informing pedestrian safety decisions.

In conclusion, integrating PBCAT crash typing and textual data mining techniques offers a promising path to process and extract valuable information from vast amounts of unstructured crash-narrative data. The automation of the crash typing process can yield structured data at a scale that provides valuable insight into pedestrian safety. With the constant evolution of large language models, it is also important to address the existing challenges that the literature has identified such as data set size, inconsistent narration styles, and data imbalance. Addressing these challenges will be critical in developing more sophisticated crash typing tools. By enhancing the exposure of models to crash narratives and employing robust evaluation methods, we can unlock the full potential of crash-narrative data and make significant strides in improving transportation safety decision-making processes.

Acknowledgments

The authors would like to acknowledge the valuable contributions of Mahin Ramezani to this research. Portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: S. Das, M. Le; data collection: M. Le; analysis and interpretation of results: A. Oliaee, S. Das; draft manuscript preparation: A. Oliaee, S. Das, and M. Le. All authors reviewed the results and approved the final version of the manuscript.


Declaration of Conflicting Interests


The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This project has been funded by Center for Transportation Safety (Texas A&M Transportation Institute).

ORCID iDs

Amir Hossein Oliaee  <https://orcid.org/0000-0002-0574-2690>

Subasish Das  <https://orcid.org/0000-0002-1671-2753>

Minh Le  <https://orcid.org/0000-0003-0129-1615>

References

1. Traffic Safety Facts 2021 Data: Pedestrians. Traffic Safety Facts, 2023. <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813509.pdf>
2. Traffic Safety Facts 2020 Data: Pedestrians. Traffic Safety Facts, 2022. <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813310.pdf>
3. Federal Highway Administration. Develop Pedestrian and Bicycle Crash Analysis Tool. *Public Roads*, Vol. 63, No. 5, 2000, p. 54.
4. Lopez, D., L. C. Malloy, and K. Arcoletto. Police Narrative Reports: Do They Provide End-Users with the Data They Need to Help Prevent Bicycle Crashes? *Accident Analysis & Prevention*, Vol. 164, 2022, p. 106475. <https://doi.org/10.1016/j.aap.2021.106475>.
5. Shah, N. R., S. Aryal, Y. Wen, and C. R. Cherry. Comparison of Motor Vehicle-Involved e-Scooter and Bicycle Crashes Using Standardized Crash Typology. *Journal of Safety Research*, Vol. 77, 2021, pp. 217–228. <https://doi.org/10.1016/j.jsr.2021.03.005>.
6. Schneider, R. J., and J. Stefanich. Application of the Location–Movement Classification Method for Pedestrian and Bicycle Crash Typing. *Transportation Research Record: Journal of the Transportation Research Board*, 2016. 2601: 72–83.
7. Chavis, C., Y.-J. Lee, and S. Dadvar. *Analysis of Bicycle and Pedestrian Crash Causes and Interventions*. District Department of Transportation, Washington, D.C., 2018, p. 368.
8. Das, S. *Artificial Intelligence in Highway Safety*. CRC Press, Boca Raton, FL, 2022.
9. Das, S., A. Dutta, G. Medina, L. Minjares-Kyle, and Z. Elgart. Extracting Patterns from Twitter to Promote Biking. *IATSS Research*, Vol. 43, No. 1, 2019, pp. 51–59.
10. Das, S., A. Dutta, T. Lindheimer, M. Jalayer, and Z. Elgart. YouTube as a Source of Information in Understanding Autonomous Vehicle Consumers: Natural Language Processing Study. *Transportation Research Record: Journal of the Transportation Research Board*, 2019. 2673: 242–253.
11. Das, S. #TRBAM: Social Media Interactions from Transportation's Largest Conference. *TR News*, No. 324, 2019, pp. 18–23.
12. Hosseini, P., S. Khoshsirat, M. Jalayer, S. Das, and H. Zhou. Application of Text Mining Techniques to Identify Actual Wrong-Way Driving (WWD) Crashes in Police Reports. *International Journal of Transportation Science and Technology*, Vol. 12, No. 4, 2023, pp. 1038–1051.
13. Das, S. TCRP Synthesis 156: How Do Transit Agencies Use Social Media? *TR News*, No. 343, 2023, pp. 10–14.
14. Kutela, B., S. Das, and B. Dadashova. Mining Patterns of Autonomous Vehicle Crashes Involving Vulnerable Road Users to Understand the Associated Factors. *Accident Analysis & Prevention*, Vol. 165, 2022, p. 106473.
15. Das, S., J. J. C. Aman, and M. A. Rahman. Content Analysis on Homelessness Issues at Airports by News Media Mining. *Transportation Research Record: Journal of the Transportation Research Board*, 2023. 2677: 635–647.
16. Das, S., and S. Sarkar. News Media Mining to Explore Speed-Crash-Traffic Association During COVID-19. *Transportation Research Record: Journal of the Transportation Research Board*, 2022: 03611981221121261.
17. Das, S., X. Sun, and A. Dutta. Text Mining and Topic Modeling of Compendiums of Papers from Transportation Research Board Annual Meetings. *Transportation Research Record: Journal of the Transportation Research Board*, 2016. 2552: 48–56.
18. Das, S., A. Dutta, and I. Tsapakis. Topic Models from Crash Narrative Reports of Motorcycle Crash Causation Study. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 449–462.
19. Das, S. Exploratory Analysis of Unmanned Aircraft Sightings Using Text Mining. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 291–300.
20. Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. *arXiv*, 2017. <https://arxiv.org/abs/1706.03762>.
21. Oliaee, A., S. Das, J. Liu, and M. A. Rahman. Using Bidirectional Encoder Representations from Transformers (BERT) to Classify Traffic Crash Severity Types. *Natural Language Processing Journal*, Vol. 3, 2023, p. 100007.
22. Weng, Y., S. Das, and S. G. Paal. Applying Few-Shot Learning in Classifying Pedestrian Crash Typing. *Transportation Research Record: Journal of the Transportation Research Board*, 2023. 2677: 563–572.
23. Das, S., A. H. Oliaee, M. Le, M. P. Pratt, and J. Wu. Classifying Pedestrian Maneuver Types Using the Advanced Language Model. *Transportation Research Record: Journal of the Transportation Research Board*, 2023. 2677: 599–611.
24. NHTSA. 2020 FARS/CRSS Pedestrian Bicyclist Crash Typing Manual: A Guide for Coders Using the FARS/CRSS Ped/Bike Typing Tool. National Highway Traffic Safety Administration, Washington, D.C., 2022, p. 86.
25. Banks, G. C., H. M. Woznyj, R. S. Wesslen, and R. L. Ross. A Review of Best Practice Recommendations for Text Analysis in R (and a User-Friendly App). *Journal of Business and Psychology*, Vol. 33, No. 4, 2018, pp. 445–459.
26. Kwayu, K. M., V. Kwigizile, J. Zhang, and J.-S. Oh. Semantic N-Gram Feature Analysis and Machine Learning-Based Classification of Drivers' Hazardous Actions at Signal-Controlled Intersections. *Journal of Computing in Civil Engineering*, Vol. 34, No. 4, 2020, p. 04020015.
27. Das, S., M. Le, and B. Dai. Application of Machine Learning Tools in Classifying Pedestrian Crash Types: A Case

- Study. *Transportation Safety and Environment*, Vol. 2, No. 2, 2020, pp. 106–119. <https://doi.org/10.1093/tse/tdaa010>.
28. Kwayu, K. M., V. Kwigizile, K. Lee, and J.-S. Oh. Discovering Latent Themes in Traffic Fatal Crash Narratives Using Text Mining Analytics and Network Topology. *Accident Analysis & Prevention*, Vol. 150, 2021, p. 105899.
 29. Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805* [cs], 2019.
 30. Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed Representations of Words and Phrases and Their Compositionality. *Advances in Neural Information Processing Systems*, Vol. 26, 2013, p. 9.
 31. Pennington, J., R. Socher, and C. Manning. Glove: Global Vectors for Word Representation. *Proc., 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1532–1543.
 32. Liu, Y., M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A Robustly Optimized Bert Pretraining Approach. *arXiv Preprint arXiv:1907.11692*, 2019.
 33. Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, and M. Funtowicz. Transformers: State-of-the-Art Natural Language Processing. 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 38–45.
 34. Maiya, A. S. Ktrain: A Low-Code Library for Augmented Machine Learning. *The Journal of Machine Learning Research*, Vol. 23, No. 1, 2022, pp. 7070–7075.
 35. Smith, L. N. Cyclical Learning Rates for Training Neural Networks. *arXiv Preprint arXiv:1506.01186*, 2017.
 36. Taylor, W. L. “Cloze Procedure”: A New Tool for Measuring Readability. *Journalism Quarterly*, Vol. 30, No. 4, 1953, pp. 415–433. <https://doi.org/10.1177/107769905303000401>.
 37. Phan, S., G. Henter, Y. Miyao, and S. Satoh. Consensus-Based Sequence Training for Video Captioning. *arXiv*, 2017. <https://arxiv.org/abs/1712.09532>.
 38. Lin, T., P. Goyal, R. Girshick, K. He, and P. Dollár. Focal Loss for Dense Object Detection. *arXiv*, 2018. <https://arxiv.org/abs/1708.02002>.
 39. Kingma, D. P., and J. Ba. Adam: A Method for Stochastic Optimization. <http://arxiv.org/abs/1412.6980>. Accessed June 9, 2022.
 40. Hossin, M., and M. N. Sulaiman. A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*, Vol. 5, No. 2, 2015, pp. 1–11. <https://doi.org/10.5121/ijdkp.2015.5201>.
 41. Hand, D. J., and R. J. Till. A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning*, Vol. 45, 2001, pp. 171–186.