Enhancing Traffic Accident Classifications: Application of NLP Methods for City Safety

Enes Özeren[1]*, Alexander Ulbrich[1]*, Sascha Filimon[2], David Rügamer[1,3], and Andreas Bender[1,3] (✉)

[1] Department of Statistics, LMU Munich, Munich, Germany
{enes.oezeren,a.ulbrich}@campus.lmu.de
[2] City of Munich, Munich, Germany
[3] Munich Center for Machine Learning (MCML), Munich, Germany
andreas.bender@stat.uni-muenchen.de

Abstract. A comprehensive understanding of traffic accidents is essential for improving city safety and informing policy decisions. In this study, we analyze traffic incidents in Munich to identify patterns and characteristics that distinguish different types of accidents. The dataset consists of both structured tabular features, such as location, time, and weather conditions, as well as unstructured free-text descriptions detailing the circumstances of each accident. Each incident is categorized into one of seven predefined classes. To assess the reliability of these labels, we apply NLP methods, including topic modeling and few-shot learning, which reveal inconsistencies in the labeling process. These findings highlight potential ambiguities in accident classification and motivate a refined predictive approach. Building on these insights, we develop a classification model that achieves high accuracy in assigning accidents to their respective categories. Our results demonstrate that textual descriptions contain the most informative features for classification, while the inclusion of tabular data provides only marginal improvements. These findings emphasize the critical role of free-text data in accident analysis and highlight the potential of transformer-based models in improving classification reliability.

Keywords: Few-Shot Learning · Topic Modeling · City Safety.

1 Introduction

Traffic accidents pose a substantial risk to human life and incur high economic costs. Understanding their underlying causes and patterns is essential - not only to mitigate their consequences but also to develop effective preemptive strategies in the context of city safety and city planning. Numerous studies have investigated various aspects of traffic accidents, from identifying their root causes to assessing their severity using data analysis techniques [6, 14, 19, 27]. The foundation of such studies is the availability of high-quality accident data. However,

[*] These authors contributed equally to this work.

real-world accident records are often affected by inconsistencies and human error during data collection. These issues can lead to inaccuracies in how accidents are categorized, potentially obscuring important insights and limiting the effectiveness of data-driven safety measures.

Table 1. Classification of accidents and their explanations.

Code	Classification	Explanation
A1	Driving Accident	Loss of control of vehicle.
A2	Turning / Crossing Accid.	Conflict between turning vehicle and one moving in parallel direction.
A3	Turning Accident	Conflict betw. (turning) vehicle and another moving perpendicularly.
A4	Crossing Accident	Conflict between vehicle and crossing pedestrian.
A5	Stationary Accident	Conflict where at least one party must be stationary/parking.
A6	Longitudinal Accident	Conflict between parties moving in parallel, none of above applicable.
A7	Other Accident	None of the above applicable.

In the city of Munich, Germany, policemen record accident information at the location of the incident, which includes general information like date, time and location, person-specific characteristics (age, drug involvement, injury severity) as well as free-text description of the events leading up to the incident. The specific dataset used in this study is comprised of 105,217 unique traffic accidents recorded between 01.01.2017 and 31.12.2022, of which 102,569 contain free-text. Additionally, on-site, the policemen classify the accidents into one of seven distinct accident types (A1 – A7), listed in Table 1. This is done by (mentally) matching the course of events to the definition of the respective accident types. In order to avoid confusion later on, we define the following terms:

- *label definition*: A textual definition for each of the seven accident types (A1 - A7).
- *example text* or *accident description*: The free-text description of the accident recorded by the policemen on-site.
- *human label*: The label (A1 - A7) assigned to an accident on-site by the policemen.
- *ground truth*: 236 additional labels created by expert labelers for a small subset of accidents.

In our data set, almost 50% of the human labels fall within the fallback category A7. Given the high proportion of accidents in this category, a high misclassification rate is suspected. This is further supported by comparison to the city of Berlin, where only 25% of accidents fall into this fallback category. While there may be inherent differences between the two cities, the large proportion of accidents in the fallback category (A7) indicates a potentially high amount of mislabeling.

In this work, we aim to gain insights into the reasons for mislabeling and to improve the current classification system. This could lead to the implementation of better and more accurate safety measures. To this end, we utilize multimodal data, comprising free-text incident descriptions as well as tabular data.

Using advanced Natural Language Processing (NLP) techniques that have been successfully applied in various domains, including medical records, legal documents, and police reports [8, 20, 28], we gain insights about missclassification by applying transformer-based methods. Furthermore, we develop a classification model that achieves high accuracy in correctly assigning accidents. We make the code publicly available.[4]

2 Related Work

2.1 Traffic Accident Analysis

With the growing popularity of NLP methods, recent research has increasingly explored their application in accident data analysis [14, 19, 27]. While structured data has been extensively studied (see, e.g., [6, 12]), the use of unstructured free-text features has gained traction only in recent years. Free-text descriptions can encode nuanced information that structured numerical data cannot fully capture, offering deeper contextual insights [27]. Early approaches to leveraging free-text descriptions often relied on simple word-count-based methods, such as [14], where text features were extracted based on keyword frequencies. Beyond traffic accidents, similar methods have been applied in legal text analysis. For instance, [13] achieved strong results using LSTMs for petition analysis and suggested exploring transformer-based methods as a next step. Recent research has increasingly focused on using word embeddings to capture richer semantic information. In this context, [19] employ BERT to incorporate free-text accident descriptions, demonstrating promising performance in a classification setting. Their findings suggest that further integrating free-text features could unlock significant potential, as such descriptions are widely used across different countries [19]. Their work focuses on extracting specific information from the accident descriptions and to compare classical and modern NLP approaches for classification, assuming the human labels to represent ground truth. In contrast, we investigate potential mislabeling of accident types and use multiple data modalities for classification compared to text-based inputs only.

2.2 Large Language Models

The Transformer architecture, proposed in 2017 for machine translation tasks [24], quickly became the dominant paradigm in the NLP domain [15]. The original design consists of an encoder-decoder structure, but both components are also independently used. While encoder-only models are utilized primarily for learning text representations [3, 4], encoder-decoder and decoder-only models are employed for text generation tasks [16, 17].

These models, often referred to as Large Language Models (LLMs), have a large number of parameters and are trained on massive text corpora [2, 3, 4, 16, 22]. They can be fine-tuned for specific tasks, allowing for domain adaptation

[4] https://github.com/enesozeren/enhancing-traffic-accident-classifications

and improved performance [4, 16]. Alternatively, techniques such as few-shot and chain-of-thought prompting have enabled the application of these models with good performance without requiring any additional parameter updating [2, 25]. In this project, we utilized both approaches, few-shot classification for creating a second opinion about accident categories, and also fine-tuning for predictive modeling.

2.3 Topic Modeling

Topic models are designed to extract semantic themes from large volumes of unstructured text [1]. Traditional approaches such as latent dirichlet allocation (LDA) and non-negative matrix factorization (NMF) have been widely used for topic modeling. However, their performance is limited by a lack of semantic understanding, as they rely solely on bag-of-words representations and fail to capture contextual information [5]. Therefore, novel methods incorporating text-embeddings have been increasingly applied to topic modeling. In [5], BERTopic is proposed, a framework to perform topic modeling by creating dense vector representations of each document, which are then used for clustering. The framework makes use of Sentence-BERT [18], a time-efficient alternative of BERT [4] enabling to compare embeddings with cosine similarity. The resulting embeddings are dimensionally reduced and clustered. Finally, text representations for each cluster are chosen by modifying the TF-IDF approach proposed by [7] in a way that all documents within one cluster are treated as a single document [5]. This allows to extract class-related representative keywords. In our study, topic modeling is used to identify relevant topics within misclassified accidents.

3 Semantic Clustering

To investigate potential mislabeling, we first evaluate what semantic characteristics the free-text descriptions of accidents in a certain category show. This enables us to discover patterns within each accident type, notably which topics show up frequently within the fallback category A7.

3.1 Methods

In order to perform semantic clustering, we apply BERTopic [5] to extract topics present in the text corpus. Clustering is performed in an unsupervised way using the free-text accident descriptions only and without taking into account their human labels (A1-A7).

The first step is to convert all texts into dense vector representations. While BERTopic allows for direct usage without specifying a specific model, it is also possible to encode the text independently and pass the resulting vectors as an additional argument. One requirement for a suitable Sentence-Transformer is a context window of at least 2000 tokens, as this is the maximal text length in the dataset. Furthermore, the model is required to have German capabilities. For the study and given the two requirements, jiina-embeddings-v3 [21] is chosen from the MTEB benchmark ranking [11]. The model performs mean-pooling by default for combining all token-vectors into a single vector for each text.

Since the resulting embedding-vectors are 1024-dimensional, their dimensionality can be reduced. UMAP has shown to be able to reduce the amount of dimensions while maintaining more of the global structure than competing methods like t-SNE or PCA [10]. Four hyper-parameters have to be specified when using UMAP [10]. *Number of dimensions* controls the number of dimensions the reduced vector should have. *Number of neighbors* influences the locality of approximation patterns. If the parameter is increased, more global structures will be captured. In the context of this study, if one would be interested in many fine-grained topics, *number of neighbors* can be decreased. *Minimal distance* is mainly important for plotting since it controls how densely points can be packed together. It can be increased to avoid overplotting.

Next, the reduced embedding vectors can be clustered. For the study HDBSCAN is used, an extension of DBSCAN which allows for capturing clusters with varying densities [9]. As a hyper-parameter, a minimal cluster size can be fixed. Finally, for each cluster, representations have to be generated. The goal is to find words which are relevant for certain clusters. In [5] it is proposed to employ a variation of TF-IDF, such that documents within each cluster are considered as single documents, giving rise to the c-TF-IDF approach.

$$\mathrm{W}_{t,c} = \mathrm{tf}_{t,c} \cdot \log\left(1 + \frac{\mu}{\mathrm{tf}_t}\right) \tag{1}$$

Here the term frequency $\mathrm{tf}_{t,c}$ indicates the frequency of word t in cluster c. μ is the average number of words per class while tf_t is the frequency of term t across all classes. $\mathrm{W}_{t,c}$ can therefore be interpreted as an estimated importance score for word t within class c. It needs to be stressed that this formula does not take into account word embeddings, but is only based on word frequencies. Therefore it might fail to accurately capture the true semantic meaning of each extracted topic [5].

3.2 Results

Due to computational constraints, we limited our analysis to a random subset of 50,000 accident descriptions. The topic model extracts 18 different topics, listed in Table 2. It can be seen that multiple topics about parking accidents have been extracted. Despite looking similar according to representative documents and the selected c-TF-IDF representations, different nuances are captured within some of those topics. To give one example, “Parking 3” has a relatively high cosine similarity to the topic “Intox.” which is about accidents related to drug influence. Considerations like this can give an idea of what different subtleties the seemingly identical topics show.

As one objective of the study is to understand what kind of accidents tend to get the fallback label A7 (other accident), the resulting topics can now be

Table 2. Topics extracted by BERTopic (outliers excluded). Topic (column 1) shows subjectively labeled topic names, based on representative documents and the extracted c-TF-IDF terms for better readability. Topics are ordered in descending order with regards to their counts (column 2), i.e., the number of observations per topic. The third column includes the content of each topic.

Topic	Count	Content
Parking 1	13,846	Parking accidents
Bicycle	8,293	Bike accidents, falling from bikes
Crossroad/Crash	4,376	Accidents mostly in crossroads, many with crashes
Parking 2	2,374	Parking accidents
Parking 3	2,212	Parking accidents
Parking 4	1,457	Parking accidents
Truck	1,242	Truck accidents, mostly damaging parked vehicles
Parking 5	999	Parking accidents, damaged side mirrors
Bus	828	Bus accidents
Damaged city obj.	645	Damaged objects like traffic lights, fences or traffic signs
Scooter	540	Scooter- and motorcycle accidents
Landsbergerstr.	422	Accidents in spacial proximity to Landsbergerstreet
Schleißheimerstr.	418	Accidents in spacial proximity to Schleißheimerstreet
Intox.	399	Accidents connected to drug influence
Fürstenriederstr.	394	Accidents in spacial proximity to Fürstenriederstreet
Dachauerstr.	366	Accidents in spacial proximity to Dachauerstreet
Parking 6	305	Parking accidents
Tram	301	Tram accidents

compared to the human labels. To do this, the text corpus is clustered in two different ways:

1. The texts are divided as suggested by the topic model, giving rise to 18 clusters.
2. The texts are divided as suggested by the human labels, i.e., the class assignment to one of the 7 categories is used as cluster indicator. This yields 7 clusters (A1 – A7).

Both clustering schemes are used in the following way. After encoding all texts with the same model used for clustering, representative embedding vectors are generated by applying mean-pooling within each cluster defined above. After this, their cosine similarity can be calculated and summarized as depicted in Figure 1.

First, we note that the lowest cosine similarity value is around 0.7, indicating generally high similarity, as this measure ranges up to a maximum of 1. This might be due to the fact that all texts are similar in the way that they all deal with traffic accidents. What can also be seen, for instance, is a high similarity between accident type A1 (driving accident), and the bicycle topic from the topic model (column 1). In fact, this accident type represents the class among all 7 with the highest proportion of bikes involved. Looking at column five, which

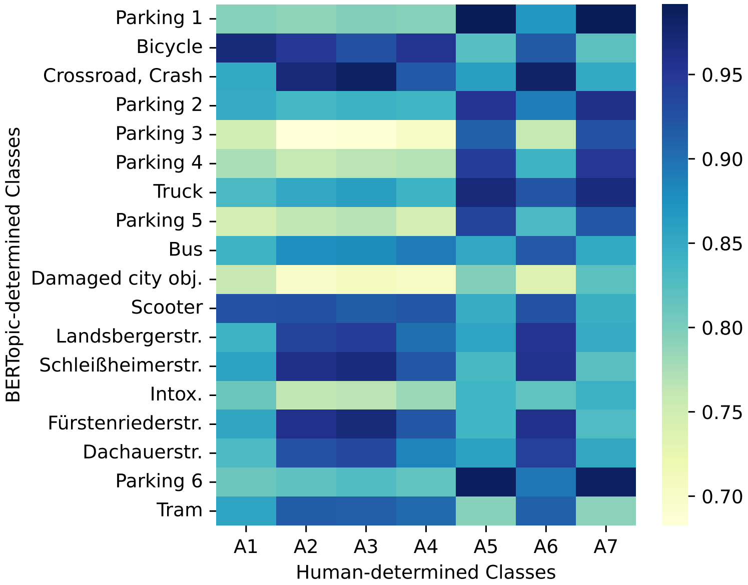


Fig. 1. Cosine similarity between BERTopic-generated and human-determined clusters.

includes accident type A5 (stationary accidents), a relatively high similarity to all six extracted parking topics is visible. Finally, column seven, which represents the fallback category A7 (other accidents) shows almost exactly the same color pattern as column five. In other words, accidents labeled as "other accident" seem to be similar to what our topic model identifies as parking accidents.

4 Classification with Few-Shot Prompting

In this section, labels are generated based on few-shot prompting techniques, designed to mimic the human labeling process in that it matches an accident description to the label definition of an accident type. The results are compared to the human labels, revealing potential anomalies in the labeling behavior.

4.1 Methods

Few-shot prompting is performed by conditioning the LLM on a given task description and a small set of examples to solve a task [2]. This approach has been proven to work better than zero-shot prompting, which relies only on task description without examples [2]. Unlike fine-tuning, few-shot prompting does not involve updating model parameters and a small number of examples (typically 2-10) is sufficient for effective task adaptation. This makes it more efficient than fine-tuning, which generally requires hundreds or even thousands of labeled examples for language tasks.

To apply few-shot prompting for our accident classification, a suitable LLM is selected based on three requirements. First, the model should be open-source to ensure it can process confidential data locally. Second, it needs to have strong proficiency in German, as all our text data is in German. Lastly, a technical requirement is that the model should run on available hardware (two Nvidia RTX A6000 GPUs with 48GB memory in each). Based on these criteria, we choose the Gemma-2-27B-Instruct model by Google [22] as it meets our requirements and demonstrated strong performance in five widely used German language benchmarks [23].

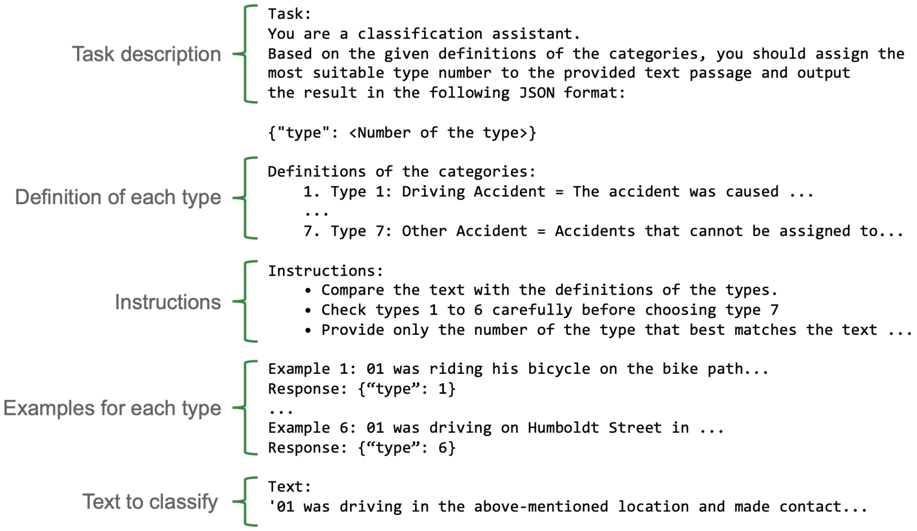
4.2 Results

For our analyses, we use the label definitions that provide a high-level, representative description of each accident type and select six exemplary accident descriptions (with verified labels) for each non-fallback accident type (categories A1 to A6) from our data set. We intentionally exclude an example for accident type A7, which is the fallback category, to prevent the model from becoming biased towards a specific instance of category A7 (e.g., an accident involving a deer). We also refrained from including multiple examples per accident type to maintain a manageable prompt length, given hardware constraints (two Nvidia RTX A6000 GPUs). This decision was necessary to keep the inference time feasible for classifying all the accidents in the dataset, which already required approximately 24 hours with our current setup. The (shortened) prompt is given in Figure 2.

We apply the few-shot prompting for each accident description individually using the Gemma model and compare the results with human labels, as shown in Figure 3. Overall, 44% of the LLM few-shot labels match the human labels. The anti-diagonal indicates a high agreement ratio for most accident types, but the large bubbles outside of the diagonal serve as an important signal for deeper analysis.

There are three bubbles larger than 20% outside the anti-diagonal in Figure 3. The first case represents 49% of accidents labeled as type A2 by humans but classified as type A3 by LLM few-shot approach. Since A2 and A3 accident types are both variations of turning accidents with a small difference (the driving direction of vehicles), we observe that LLM confused them easily. Upon reviewing examples, we find that humans could distinguish these cases more accurately than the LLM few-shot approach, potentially because they have a physical view of the accident scene which is not represented accurately in the textual description. The second case represents 21% of accidents labeled as type A6 (longitudinal accident) by humans but classified as type A3 (turning/crossing accident) by the LLM few-shot approach. Similar to the first case, human judgment is more reliable as they can interpret the temporal nature of events, whereas the LLM misclassifies those longitudinal accidents occurring just after turning maneuvers.

The third and most notable case is where 69% of accidents labeled as A7 (other accidents) by humans are classified as A5 (stationary accident) by the LLM. This case accounts for 34% of all traffic accidents. We observe that these



```
Task:
You are a classification assistant.
Based on the given definitions of the categories, you should assign the
most suitable type number to the provided text passage and output
the result in the following JSON format:

{"type": <Number of the type>}

Definitions of the categories:
    1. Type 1: Driving Accident = The accident was caused ...
    ...
    7. Type 7: Other Accident = Accidents that cannot be assigned to...

Instructions:
    • Compare the text with the definitions of the types.
    • Check types 1 to 6 carefully before choosing type 7
    • Provide only the number of the type that best matches the text ...

Example 1: 01 was riding his bicycle on the bike path...
Response: {“type”: 1}
...
Example 6: 01 was driving on Humboldt Street in ...
Response: {“type”: 6}

Text:
'01 was driving in the above-mentioned location and made contact...
```

Fig. 2. Few-shot prompt for classifying accidents. Some components of the prompt shortened for illustration. The last part (text to classify) is changed for each accident text and inference is performed with the Gemma-2-27B-Instruct model. This is the translated English version; the original prompt is in German since accident texts are also in German.

accidents consistently contain parking-related keywords such as ’garage’, ’parking’, etc., as well as misspelled variations of them, indicating that they are related to damaged parked vehicles (which is in line with our findings from Section 4.2). This finding helps reduce uncertainty about accident characteristics in the fallback category A7. Before, 49% of all accidents in Munich fell into the fallback category, “Other accident (A7)”, meaning their specific nature is unknown. Our analysis reveals that most of these accidents involve damages to parked vehicles. As a result, the proportion of accidents of unknown nature can be reduced substantially. This has practical implications. Only accidents of a known nature can be counteracted by city planning. For example, if an accident of types A1-A6 occurs frequently within a specific time span and location, countermeasures (e.g., traffic signs, etc.) can be implemented. For the fallback-category, this is not the case, as accidents could be of heterogeneous nature. However, our analysis helps to identify a large proportion of those as parking related, which can be mitigated accordingly.

5 Predictive Modeling

In this section, we describe our approach to building predictive models for accident classification. Given the potential for mislabeling in the training data, we explore different strategies for constructing training sets to improve label quality.

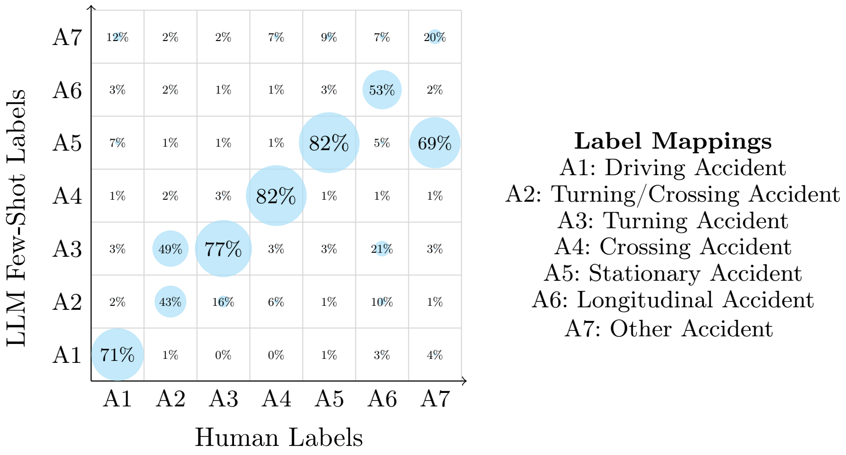


Fig. 3. Distribution of LLM few-shot labels vs. human labels. Each column sums to 100%, and bubble sizes correspond to the intersection ratio between LLM few-shot and human labels. The anti-diagonal represents the agreement ratio between the two.

Since both tabular and text data are available, we investigate these modalities individually and in combination to assess their contributions to predictive performance. Finally, we evaluate the models on an expert-labeled, ground-truth test set to provide a reliable assessment of their accuracy.

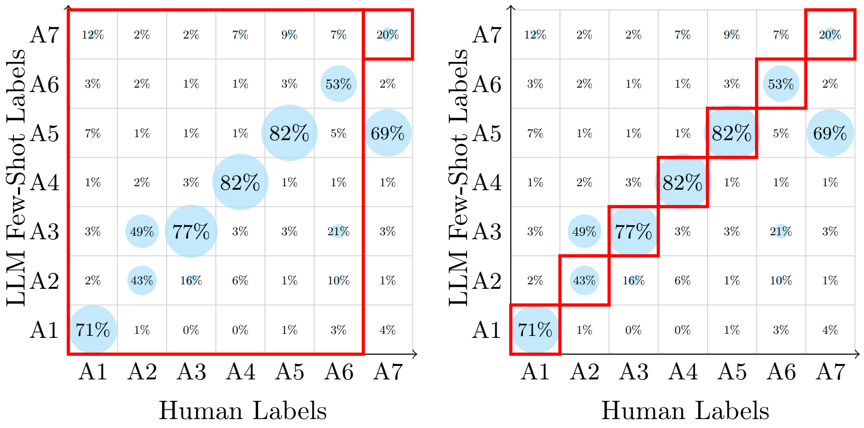
5.1 Methods

Evaluation strategy

Training Set. One of the key motivations of this study is to detect mislabeled data. Therefore, relying solely on human labels is not ideal for this purpose. Instead, we explore two approaches to construct the training set, as illustrated in Figure 4. The main objective is to select accident labels that are more likely to be correct. To achieve this, we compare human labels with those generated by an LLM using a few-shot approach (explained in Section 4).

Our first strategy assumes that all human-labeled instances for accident types A1–A6 are correct, while for A7, only labels agreed upon by both humans and the LLM are considered valid. This approach results in approximately 62,000 training labels, which we refer to as presumably low-quality labels since the assumption about human labels being entirely correct is relatively weak. The second strategy is more conservative, considering only those labels where both the human annotators and the LLM agree. This produces a smaller but more reliable set of approximately 45,000 labels, which we refer to as presumably high-quality labels.

We train supervised models using both training sets and compare their performance to assess the impact of label quality on model accuracy.



a. Low Quality Labels (63K) b. High Quality Labels (45K)

Fig. 4. Two strategies for constructing the training set. The accidents used to create the training set are highlighted in red. (a) Assuming all human-labeled instances for types A1–A6 are correct, while for type A7, only labels agreed upon by both humans and the LLM few-shot approach are considered correct. Due to the weaker assumption in this approach, we name these 62K labels as low-quality. (b) Considering only labels where both humans and the LLM few-shot approach agree as correct, resulting in 45K high-quality labels.

Test Set. To ensure reliable evaluation of model performances, we ask domain specialists from the city of Munich to carefully label 236 traffic accidents. We use these expert-labeled examples as our test set to report results.

Models & Training Details

MLP Model. Without using any text data, we employ a multi-layer perceptron (MLP) model to predict accident types based solely on tabular data. The model consists of feed-forward layers with skip connections and layer normalization. This model has 42 million parameters and is initialized randomly. The input is a fixed-size vector derived from the tabular features of the accident, and the model utilizes a softmax head to predict the accident type.

We train the MLP model on two Nvidia RTX A6000 GPUs. The optimization is performed using AdamW with an initial learning rate of 0.001, which decreases linearly by a factor of 0.9 every 5 epochs. We apply dropout with a probability of 0.1 and use a weight decay of 0.01 for regularization. The model is trained for 50 epochs, and we report the results from the best checkpoint based on the lowest validation loss.

LLM Few-shot Labeling. We use the Gemma Few-shot labels for comparison with the other supervised trained models. The details of few-shot labeling are given in Section 4.

Finetuned XLM-R. To predict accident types using text data, we employ an encoder-only LLM. XLM-RoBERTa-Large (XLM-R), a multilingual model with 550 million parameters [3], serves as our backbone. We use a pre-trained version of the model for transfer learning, fine-tuning it on our dataset. The model takes the accident text description as input and predicts the accident type through a softmax classification head with the model having 560 million parameters in total.

For fine-tuning XLM-R, we use Hugging Face [26]. The model is fine-tuned on two Nvidia RTX A6000 GPUs. We use an initial learning rate of 5×10^{-5}, a weight decay of 0.01, and train for 6 epochs. The effective batch size is set to 128. As before, we select the best checkpoint based on the lowest validation loss.

Multimodal Model. To effectively handle both tabular and text data modalities, we design a multimodal model that integrates information from both sources. The architecture follows a two-branch structure:

- **Textual Input Pathway:** The accident description is processed using the XLM-R model, which transforms the text into a text embedding via mean pooling.
- **Tabular Input Pathway:** The numerical and categorical accident-related features are fed into a multi-layer perceptron (MLP) model, which encodes them into a tabular feature embedding.
- **Fusion and Prediction:** The text embedding and tabular feature embedding are concatenated and passed through another MLP model, which acts as the final classifier with a softmax output layer to predict the accident type.

The XLM-R model is initialized with pretrained weights to leverage prior knowledge from multilingual text data, while the MLP components are randomly initialized. During training, all model parameters are updated.

The multimodal model is trained on two Nvidia RTX A6000 GPUs for 5 epochs with a batch size of 64. We use an initial learning rate of 1×10^{-5}, which decreases linearly by a factor of 0.7 every epoch. We apply a dropout rate of 0.2 and a weight decay of 0.01.

Table 3. Comparison of predictive models for accident classification using different data sources and corresponding model sizes.

Model	Data Modality	Parameter Size
MLP	Tabular	42 M
LLM Few-shot Labeling	Text	27B
Finetuned XLM-R	Text	560 M
Multimodal Model	Tabular + Text	603 M

Model Comparisons To assess the different approaches and data modalities for accident classification, we compare the models using accuracy and weighted F1 score metrics calculated on the test set. Accuracy measures the proportion of correct predictions over all predictions. Weighted F1 score is computed as the weighted mean of F1 scores for each accident type

$$\text{Weighted } F_1 \text{ Score} = \textstyle\sum_{i=1}^{7} w_i \cdot F_{1,i}$$

where $F_{1,i}$ is the F1 score of class i, calculated as the harmonic mean of precision and recall, and $w_i = n_i / (\sum_{j=1}^{7} n_j)$ with n_i the number of samples in class i.

5.2 Results

Label Quality Effect. As discussed above, we constructed two training datasets: a larger one with lower-quality labels, containing approximately 62,000 accidents, and a smaller one with higher-quality labels, consisting of around 45,000 accidents. To examine the trade-off between dataset size and label quality, we trained three supervised models (MLP, Finetuned XLM-R, and Multimodal model) with the settings given in Section 5.1. We excluded the LLM few-shot approach from this comparison, as it does not involve parameter updates during training.

Table 4 presents the performance of models trained on either low- or high-quality labels, evaluated on a test set of 236 expert-labeled accidents. Despite the smaller size of the high-quality label dataset, models trained on it achieve comparable performance to those trained on the larger low-quality label dataset. This highlights the importance of high-quality labels in model training. However, one should take into account the additional cost associated with generating high-quality labels—specifically, the computational expense of applying LLM few-shot labeling to a larger set of accidents. In our case, this was not an additional burden, as the few-shot labeling had already been applied to the full dataset for the preceding analysis.

Model Comparisons. Comparing the different models and modalities, the results indicate that the MLP model, which relies solely on tabular data, performs the worst among all approaches (53% accuracy, 0.49 weighted F1 score). In contrast, all models incorporating text data—whether alone or in combination with tabular features—demonstrate better performance ($\geq$ 61% accuracy, $\geq$ 0.58 weighted F1 score). This suggests that textual information is the primary source of information for this classification task, whereas tabular data alone lacks the necessary detail for accurate accident classification.

When comparing the LLM Few-Shot approach (61% accuracy) to the finetuned XLM-R model (72% accuracy), we observe that finetuning leads to superior performance. Even though Gemma is a much larger model, with 27 billion parameters in a decoder-only architecture, it underperforms relative to the 560-million-parameter encoder-only XLM-R model. Two key factors likely contribute to this outcome. First, encoder-only models like XLM-R leverage a bidirectional attention mechanism, which is inherently more effective for text classification

Table 4. Test set accuracy and weighted F1 scores for different models trained with datasets containing either low-quality (Low-Q) or high-quality (High-Q) labels. LLM Few-Shot Labeling is not explicitly trained, therefore presented in the center of both columns. Best accuracy and weighted F1 score values are highlighted in bold.

Model	Data Modality	Test Set Performance			
		Accuracy		W. F1 Score	
		Low-Q	High-Q	Low-Q	High-Q
MLP	Tabular	0.53	0.53	0.49	0.48
LLM Few-Shot Labeling	Text	0.61		0.58	
Finetuned XLM-R	Text	0.72	0.72	0.68	**0.70**
Multimodal Model	Text + Tabular	**0.73**	0.70	0.68	0.68

tasks compared to the autoregressive nature of decoder-only models. Second, finetuning allows the model to better adapt to domain-specific data, whereas few-shot prompting, even with six examples, does not provide the same level of task specialization.

Finally, comparing the Multimodal Model to the Finetuned XLM-R shows no large difference in performance when both text and tabular data are used. This suggests that textual data carries the most relevant information for accident classification in our predictive models.

6 Conclusion and Future Directions

In this study, we analyzed Munich traffic accidents using multiple data modalities to uncover meaningful patterns. Through semantic clustering, we identified distinct topics across seven accident categories, providing deeper insights into their characteristics. To further investigate accident categorization, we employed an LLM with a few-shot approach and compared its results with human labels. Disagreements between the two revealed that many cases in the "other accidents" category involved damaged parked vehicles, supporting our findings from semantic clustering.

We also explored predictive modeling. Our results showed that models using text data outperformed those relying solely on tabular data, demonstrating the value of textual information for accident classification. Overall, our findings highlight the importance of NLP techniques in understanding traffic accidents. By leveraging textual data and machine learning, this approach offers valuable insights that can inform safety measures and contribute to the development of safer cities.

Future developments could focus on improving and incorporating data-based classification into practice, for example by deploying such a model to make real-time suggestions to human labelers (human-in-the-loop) and use active learning approaches to improve model-based classification over time.

Acknowledgments. We thank the City of Munich for supporting this project and providing access to the dataset.

Bibliography

[1] Blei, D.M.: Probabilistic topic models. Communications of the ACM **55**(4), 77–84 (2012)

[2] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)

[3] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116 (2019)

[4] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers). pp. 4171–4186 (2019)

[5] Grootendorst, M.: Bertopic: Neural topic modeling with a class-based tf-idf procedure. arXiv preprint arXiv:2203.05794 (2022)

[6] Hossain, M.M., Zhou, H., Das, S.: Data mining approach to explore emergency vehicle crash patterns: A comparative study of crash severity in emergency and non-emergency response modes. Accident Analysis & Prevention **191**, 107217 (2023)

[7] Joachims, T., et al.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: ICML. vol. 97, pp. 143–151. Citeseer (1997)

[8] Li, L., Zhou, J., Gao, Z., Hua, W., Fan, L., Yu, H., Hagen, L., Zhang, Y., Assimes, T.L., Hemphill, L., et al.: A scoping review of using large language models (llms) to investigate electronic health records (ehrs). arXiv preprint arXiv:2405.03066 (2024)

[9] McInnes, L., Healy, J., Astels, S., et al.: hdbscan: Hierarchical density based clustering. J. Open Source Softw. **2**(11), 205 (2017)

[10] McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)

[11] Muennighoff, N., Tazi, N., Magne, L., Reimers, N.: Mteb: Massive text embedding benchmark. arXiv preprint arXiv:2210.07316 (2022). https://doi.org/10.48550/ARXIV.2210.07316, https://arxiv.org/abs/2210.07316

[12] Nassereddine, H., Santiago-Chaparro, K.R., Noyce, D.A.: Evaluating right-turn flashing yellow arrow for vehicle–pedestrian interactions using a non-probabilistic regression approach. Transportation research record **2678**(2), 212–222 (2024)

[13] Noguti, M.Y., Vellasques, E., Oliveira, L.S.: Legal document classification: An application to law area prediction of petitions to public prosecution

service. In: 2020 International joint conference on neural networks (IJCNN). pp. 1–8. IEEE (2020)

[14] Park, S., Park, S., Jeong, H., Yun, I., So, J.: Scenario-mining for level 4 automated vehicle safety assessment from real accident situations in urban areas using a natural language process. Sensors **21**(20), 6929 (2021)

[15] Patwardhan, N., Marrone, S., Sansone, C.: Transformers in the real world: A survey on nlp applications. Information **14**(4), 242 (2023)

[16] Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)

[17] 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. Journal of machine learning research **21**(140), 1–67 (2020)

[18] Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 3982–3992. Association for Computational Linguistics, Hong Kong, China (Nov 2019). https://doi.org/10.18653/v1/D19-1410, https://aclanthology.org/D19-1410/

[19] Seo, Y., Park, J., Oh, G., Kim, H., Hu, J., So, J.: Text classification modeling approach on imbalanced-unstructured traffic accident descriptions data. IEEE Open Journal of Intelligent Transportation Systems **4**, 955–965 (2023)

[20] Siino, M., Falco, M., Croce, D., Rosso, P.: Exploring llms applications in law: A literature review on current legal nlp approaches. IEEE Access (2025)

[21] Sturua, S., Mohr, I., Akram, M.K., Günther, M., Wang, B., Krimmel, M., Wang, F., Mastrapas, G., Koukounas, A., Koukounas, A., Wang, N., Xiao, H.: jina-embeddings-v3: Multilingual embeddings with task lora (2024), https://arxiv.org/abs/2409.10173

[22] Team, G., Riviere, M., Pathak, S., Sessa, P.G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., et al.: Gemma 2: Improving open language models at a practical size. arXiv preprint arXiv:2408.00118 (2024)

[23] Thellmann, K., Stadler, B., Fromm, M., Buschhoff, J.S., Jude, A., Barth, F., Leveling, J., Flores-Herr, N., Köhler, J., Jäkel, R., et al.: Towards multilingual llm evaluation for european languages. arXiv preprint arXiv:2410.08928 (2024)

[24] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[25] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems **35**, 24824–24837 (2022)

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

[27] Wu, J., Heydecker, B.: Natural language understanding in road accident data analysis. Advances in Engineering Software **29**(7-9), 599–610 (1998)

[28] Xing, X., Chen, P.: Entity extraction of key elements in 110 police reports based on large language models. Applied Sciences **14**(17), 7819 (2024)