applied sciences

MDPI

Article

Harnessing Generative Pre-Trained Transformers for Construction Accident Prediction with Saliency Visualization

Byunghee Yoo [1], Jinwoo Kim [2], Seongeun Park [1], Changbum R. Ahn [1,*] and Taekeun Oh [3,*]

[1] Department of Architecture and Architectural Engineering, Seoul National University, Gwanak-gu, Seoul 08826, Republic of Korea; pikaybh@snu.ac.kr (B.Y.); seongeunpark@snu.ac.kr (S.P.)
[2] Department of Architectural Engineering, Gachon University, Seongnam-si 13120, Gyeonggi-do, Republic of Korea; jinwoo@gachon.ac.kr
[3] Department of Safety Engineering, Incheon National University, Incheon 22012, Republic of Korea
* Correspondence: cbahn@snu.ac.kr (C.R.A.); tkoh@inu.ac.kr (T.O.)

Abstract: Leveraging natural language processing models using a large volume of text data in the construction safety domain offers a unique opportunity to improve understanding of safety accidents and the ability to learn from them. However, little effort has been made to date in regard to utilizing large language models for the prediction of accident types that can help to prevent and manage potential accidents. This research aims to develop a model for predicting the six types of accidents (caught-in-between, cuts, falls, struck-by, trips, and others) by employing transfer learning with a fine-tuned generative pre-trained transformer (GPT). Additionally, to enhance the interpretability of the fine-tuned GPT model, a method for saliency visualization of input text was developed to identify words that significantly impact prediction results. The models were evaluated using a comprehensive dataset comprising 15,000 actual accident records. The results indicate that the suggested model for detecting the six accident types achieves 82% accuracy. Furthermore, it was observed that the proposed saliency visualization method can identify accident precursors from unstructured free-text data of construction accident reports. These results highlight the advancement of the generalization performance of large language processing-based accident prediction models, thereby proactively preventing construction accidents.

Keywords: large language model; generative pre-trained transformer; fine-tuning; accident prediction; saliency visualization; construction safety

Citation: Yoo, B.; Kim, J.; Park, S.; Ahn, C.R.; Oh, T. Harnessing Generative Pre-Trained Transformers for Construction Accident Prediction with Saliency Visualization. *Appl. Sci.* **2024**, *14*, 664. https://doi.org/10.3390/app14020664

Academic Editor: Paulo Santos

Received: 10 December 2023
Revised: 8 January 2024
Accepted: 10 January 2024
Published: 12 January 2024

1. Introduction

An unprecedented volume of digital data, approximately 330 million terabytes daily, is now available [1]. Looking ahead to the next two years leading up to 2025, the global production of data is projected to soar beyond 181 zettabytes [2]. A significant portion of this data exists in unstructured forms, encompassing text, images, and videos, making up an estimated 80% of the total [3]. This surge in data presents the challenge of information overload, where the sheer volume surpasses the capabilities for effective processing and analysis [4]. This issue is particularly pronounced for unstructured free-text data, which traditionally relies on human intervention to extract meaningful insights [5]. Therefore, the development of automated techniques for processing natural language text is becoming increasingly vital.

The construction sector is also witnessing a data surge and an increasing emphasis on leveraging written text, particularly within the domain of construction safety through digital accident reports [6,7]. The ability to learn from past accidents, incidents, and near misses holds paramount importance in preventing future injuries [6,8–10]. Safety reports, in particular, serve as invaluable resources for safety managers, providing insights into the conditions and events that lead to accidents. This information is crucial for implementing interventions and ensuring positive safety outcomes. Traditionally, all construction-related

disasters, including accidents, incidents, and near-misses, have been documented in safety reports using unstructured or semi-structured free-text formats. These reports encompass event descriptions, timing information, and location details [11,12]. However, analyzing accident report data is often laborious and time-consuming, demanding profound understanding of safety to extract meaningful insights [11,13]. The conventional approach involves the manual classification of accident cases, typically undertaken by safety professionals [11,14]. This method requires a meticulous review of detailed textual accident reports to categorize accidents based on various accident-related attributes. Furthermore, beyond consuming substantial resources, manual classification is susceptible to human bias and errors, potentially leading to incomplete or inaccurate analyses.

The advancements in artificial intelligence (AI) now allow for the automated processing, organization, and handling of free-text data, streamlining this analytical process [15]. Specifically, text rule-based, machine, and deep learning approaches have been showing potential in predicting anticipated danger and enhancing the understanding of accident causation. For instance, Zhang et al. [16] developed models aimed at classifying 11 causes of accidents, such as instances of being caught-in-between objects, the collapse of objects, electrocution, exposure to chemical substances, among others, utilizing accident report data and machine learning algorithms. The results of their study demonstrated the average F1 score for 11 causes of accidents was 0.68, showcasing the effectiveness of their proposed ensemble model with optimized weights. Separately, Baker et al. [17] developed machine learning models (e.g., random forest, extreme gradient boosting, and linear support vector machine) to predict four safety outcomes, such as injury severity, injury type, the body part impacted, and accident type. The results showed 84.84% in F1 scores for severity prediction. These studies underscore the significance of leveraging advanced modeling techniques to gain insights into the diverse factors contributing to accidents and improve safety measures. Nevertheless, previous research in this domain has primarily concentrated on the development of traditional machine and deep learning methods. These methods involve the manual extraction of text features, which are subsequently fed into a classifier [11]. These approaches contain several inherent limitations such as limited scalability/generalization, difficulty in analyzing free-from text, high dimensionality, and limited adaptability to new tasks. Although traditional machine/deep learning-based approaches have made significant strides in the natural language processing (NLP) of the construction domain, such limitations must be mitigated by incorporating more generative and powerful AI models [18].

Recently, large language models (LLMs), particularly those built on the transformer architecture, such as the generative pre-trained transformer (GPT), present notable advantages for classification tasks compared to traditional machine learning and deep learning models [19,20]. The transformer's ability to capture contextual information, employ end-to-end learning without manual feature engineering, utilize attention mechanisms for distinct understanding, and leverage transfer learning for improved performance make it well-suited for tasks that require a comprehension of accident-related text data [21]. Additionally, its adaptability to varied input lengths, multimodal capabilities, and efficient parallelization contribute to its efficacy in handling the complexities of accident classification, marking a significant advancement over more traditional approaches in the field [22]. Despite these potential benefits of LLM, leveraging GPT models is still an explanatory stage in the construction accident domain. Further research and validation are needed to assess the model's performance, generalizability across diverse accident scenarios, and its interpretability in the context of safety-critical applications. Additionally, addressing domain-specific challenges and tailoring the model to the unique characteristics of construction-related text data via the fine-tuning method is crucial for maximizing the effectiveness of GPT-based approaches in accident classification within the construction industry.

In this regard, the authors developed a model to predict construction accidents and visualize accident precursors using a fine-tuned GPT model from raw construction accident reports. The contributions of this paper are as follows: (1) The authors developed an

approach to automatically classify construction accidents from a dataset of raw construction accident reports using a fine-tuned GPT with various hyperparameter optimizations (e.g., max token and temperature parameters). Such information holds unique value, as it can enhance the ability to understand, predict, and prevent accidents; (2) This paper proposes a method to identify word patterns that, following the fine-tuning process, consistently demonstrate high predictiveness for each safety outcome; (3) The suggested methods can also serve to visualize and comprehend the models' prediction sensitivity; and (4) The authors experimented with four state-of-the-art machine learning [Term Frequency-Inverse Document Frequency (TF-IDF)], deep learning [Convolutional Neural Network (CNN)], and two LLMs [fine-tuned GPT 2.0/3.0 and Korean Bidirectional Encoder Representations from Transformers (BERT)] using structured and unstructured text from construction accident reports.

This paper is structured as follows to demonstrate the aforementioned contributions. First, the authors offer a background regarding the utilization of machine learning, deep learning, and LLMs in NLP in Section 2. Subsequently, in Section 3, the authors outline the structure of the proposed model, introducing the saliency visualization of accident attributes in unstructured free-text data. In Section 4, the authors test the developed method with the actual construction accident dataset and the preprocessing procedures applied and elaborate on the experimental configuration alongside comparative benchmark models, and report the results. Then, Section 5 includes a comprehensive analysis and interpretation of the findings. Lastly, Section 6 provides a summary and conclusion of the study's key contributions. The findings of this paper underscore the improved generalization performance of construction accident prediction models based on large language processing, leading to a proactive approach in preventing accidents at construction sites.

2. Research Background

The adoption of NLP analysis presents a transformative shift in the realm of construction site safety [23]. NLP, equipped with its ability to sift through extensive textual data, provides a robust framework for uncovering subtle patterns and correlations within accident records [24,25]. This enables stakeholders to access distinctive insights into the fundamental reasons behind accidents, enabling the creation of proactive and specific safety measures to avert the recurrence of similar incidents [26,27]. By leveraging the capabilities of NLP, construction site managers and safety experts can optimize their decision-making processes, bolstering the efficacy of accident analysis and fortifying the overall safety standards within the construction industry [28]. This transition to NLP-driven analysis signifies a progressive stride toward a more data-driven and proactive approach to accident prevention, ultimately fostering a safer working environment for construction personnel and mitigating potential risks associated with complex construction operations.

Specifically, recent developments in NLP have created novel opportunities for the automated examination of textual records related to accidents [17,23,29,30]. By applying NLP techniques, pertinent information can be extracted from unstructured free-text data, enabling an effective categorization of accidents based on various parameters [31]. Numerous studies have emphasized the efficacy of NLP in automating accident classification, resulting in improved efficiency and reduced prejudice. Table 1 presents a summary of the NLP models used in the literature to predict construction accidents. Specifically, Tixier et al. [32] demonstrated analyzing unstructured incident reports utilizing the NLP model, yielding significant results with F1 score values of 0.96, respectively. Similarly, Zhang et al. [16] employed text mining and NLP methods to investigate construction accident reports, utilizing various machine learning models, with the optimized ensemble model showcasing the highest F1 score of 0.68. Cheng et al. [33] introduced the Symbiotic Gated Recurrent Unit for the classification of construction site accidents, achieving an average weighted F1 score of 0.69, outperforming other AI techniques. Additionally, Kim & Chi [34] developed a system for managing construction accident knowledge, demonstrating notable recall, precision, and F1 score values of 0.71, 0.93, and 0.80, respectively.

Table 1. Natural Language Processing and Construction Injury Classification Literature.

Authors (Year)	Task	Source Data	Text Fields	Outperformed Method	Accuracy
Tixierc, A.J.P., et al. (2020) [6]	Prediction of 6 incident type, 4 injury type, 6 body part, 2 severity from injury reports	A dataset of 90,000 incident reports from global oil refineries	Title, accident details, detail, root cause	TF-IDF + SVM	71.55%
Kim, H., Jang, Y., Kang, H. & Yi, J.S. (2022) [35]	Classification of 5 accident case from accident reports	Korea Occupational Safety and Health Agency	Accident case	CNN	52%
Zhang, Jinyue, et al. (2020) [14]	Classification of 11 accident categories from accident reports	Occupational Safety and Health Administration	Accident narratives	BERT	80%
Goh, Y.M. & Ubeynarayana, C.U. (2017) [36]	Classification of 11 labels of accident causes or types from accident reports	Occupational Safety and Health Administration	Accident narratives	SVM	62%
Zhang, Fan, et al. (2019) [16]	Classification of 11 causes of accidents from accident reports	Occupational Safety and Health Administration	Fatality and catastrophe investigation summary reports	Ensemble	68%
Cheng, M.Y., Kusoemo, D. & Gosno, R.A. (2020) [33]	Classification of 11 labels of accident causes or types from accident reports	Occupational Safety and Health Administration	Accident narratives	Hybrid model	69%

These research efforts emphasize the importance of employing advanced NLP-based modeling techniques to gain a comprehensive understanding of the various factors influencing accidents and to improve safety protocols. However, as mentioned in the Introduction, current studies have predominantly focused on developing conventional machine and deep learning methods, which entail manually extracting text features and inputting them into a classifier [15,29,36]. These approaches come with inherent limitations, including limited scalability and generalization, difficulties in analyzing free-form text, high dimensionality, and a lack of adaptability to new tasks. While traditional machine and deep learning methods have made strides in NLP within the construction domain, overcoming these limitations requires the incorporation of more innovative, generative, and robust AI models.

Recently, LLMs, particularly those built on the transformer architecture such as GPT, developed by OpenAI, present notable advantages for various NLP tasks, including classification, compared to traditional machine learning and deep learning models [20,37]. These models are trained on vast datasets, enabling them to generate human-like text based on the input they receive. The adaptability and proficiency of GPT models in processing and generating natural language have the potential to significantly impact and facilitate various aspects in diverse fields, ranging from education and customer service to research and industries.

The capacity of GPT to generate text resembling human writing can be attributed to its deployment of the transformer model. The transformer model employed in GPT is depicted in Figure 1. The model employs a decoder structure with 96 repeated layers. These layers allow the model to progressively refine its understanding of the input text, enabling it to generate coherent and contextually relevant outputs [38]. The repeated stacking of decoder layers, coupled with attention mechanisms and residual connections, ensures the model's proficiency in tasks such as language translation and text generation, particularly classification [39,40]. The transformer demonstrates proficiency in capturing contextual information, implementing end-to-end learning without manual feature engineering, employing atten-

tion mechanisms for nuanced comprehension, and leveraging transfer learning to improve performance [41]. These qualities make it well-suited for tasks that require an in-depth understanding of unstructured free-text data pertaining to accidents [21]. Additionally, its adaptability to diverse input lengths, multimodal capabilities, and efficient parallelization collectively contribute to its efficacy in tackling the complexities inherent in accident classification [42]. This marks a significant advancement over traditional approaches in the field, reflecting a substantial progress in the capabilities of NLP models. Despite these potential benefits of LLM, there still exists a knowledge gap on how to better utilize pre-trained GPT models through fine- and hyperparameter-tunings with enhancing explainability for construction accident classification. This research is anticipated to contribute to the expanding knowledge base on NLP applications in safety management, providing practical insights for safety professionals, researchers, and policymakers dedicated to enhancing safety practices within the industry.

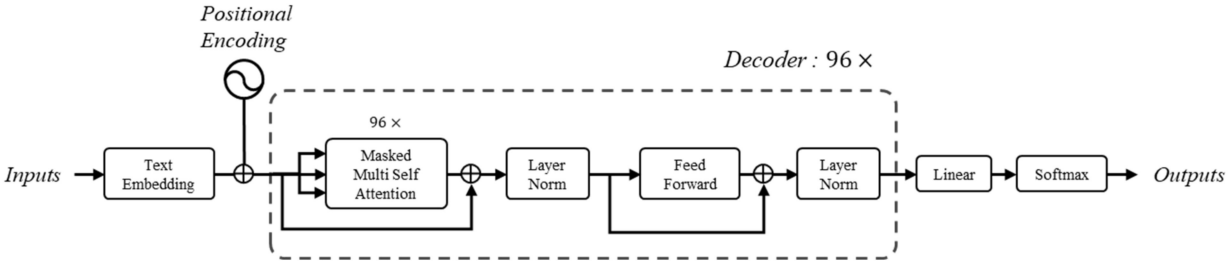

Figure 1. Transformer architecture used in a generative pre-trained transformer (Modified from Radford et al. [20] and Brown et al. [39]).

3. Materials and Methods

Despite the existence of traditional machine and deep learning-based algorithms to predict causes of accidents from structured and unstructured data, their effective use for accident classification poses technical challenges (e.g., manual feature extraction, limited to scalability/generalization/explainability, difficulty in free-from text, high dimensionality, and limited adaptability to new tasks). In this section, the authors demonstrate an alternative approach that uses GPT models, on an existing construction unstructured and structured text data to classify 6 construction accident types (caught-in-between, cuts, falls, struck-by, trips, and others) and visualize word saliency to better understand the cause of accidents.

3.1. Fine-Tuning in Generative Model Using Construction Report Data

In developing a model to predict construction accident types, this paper used a pre-trained GPT 2.0 and 3.0 model with fine-tuning using training data (unstructured free-text data from construction accident reports). The foundation of the GPT model lies in the transformer architecture introduced by Vaswani et al. [21]. The transformer architecture revolutionized NLP by replacing traditional recurrent neural networks with a self-attention mechanism. This mechanism enables GPT to effectively capture long-range dependencies in text data [20].

Figure 2 presents the fine-tuning and generating prediction outcome process. The initial step involves fine-tuning a pre-trained model on the data intended for training. Unlike prompt engineering, this involves exposing the model to the new data and allows it to adjust its internal weights and biases to better generate responses similar to the specific training data. Fine-tuning a GPT model allows for customization to specific tasks or domains, enhancing its performance on specialized inputs. This process enables the model to learn task-specific information and improve its accuracy in generating contextually relevant outputs. This paper used 5400 cases of unstructured free-text data from the construction injury report for the fine-tuning process. The data is not overlapped with the test dataset. After the fine-tuning process, the developers can interact with the fine-tuned model using the application programming interface (API) gateway of OpenAI as shown in Figure 3. For the purpose of training, the preprocessing step involves formatting

the training data into JSON files, where the input data and output data are paired as 'prompt' and 'completion', respectively. These files are then uploaded through the API gateway. Once the fine-tuning process is complete the developers can iteratively utilize the completion generated by the fine-tuned model. For example, the fine-tuned model generates a prediction outcome if one case of text data from a construction accident report is submitted in the playground page. The outcome can be validated with the actual accident outcome. To effectively utilize the API gateway of OpenAI, the authors developed an automated tuning process and generating prediction outcome process using Python.

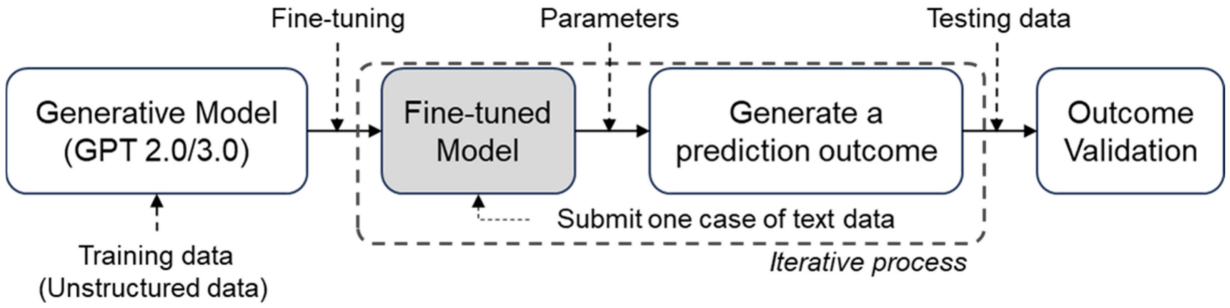

Figure 2. Overarching fine-tuning process and generating prediction outcome process.

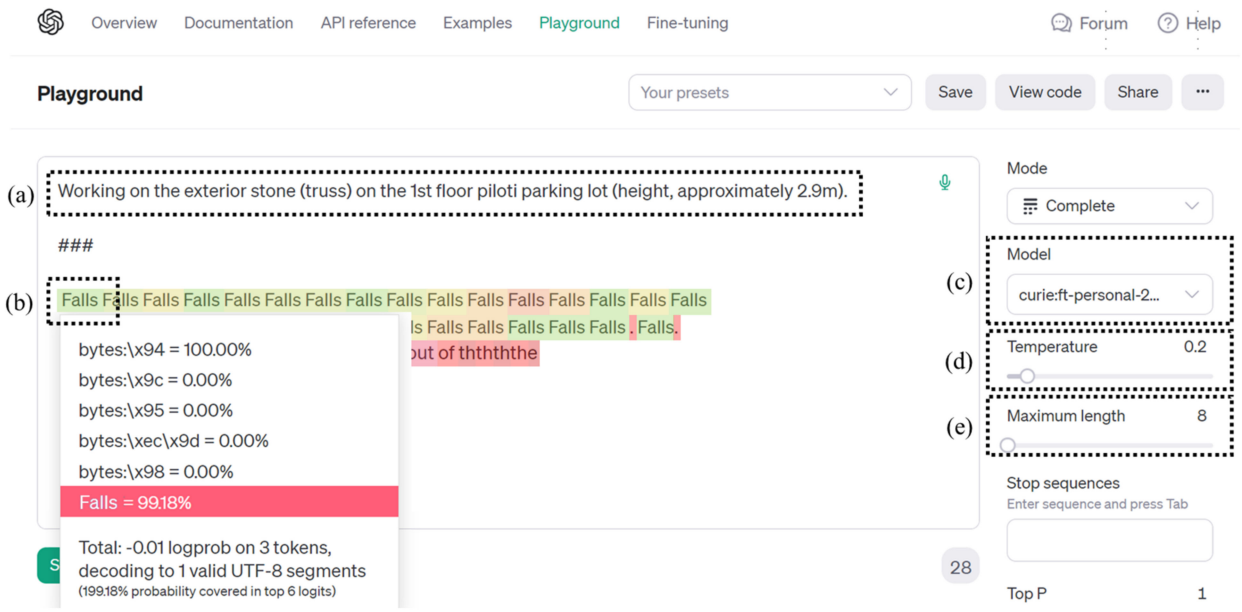

Figure 3. An example of a visualization platform to interact with a fine-tuned model: (a) a prompt with one case of text data from a construction accident report; (b) a prediction outcome, also known as a completion; (c) a model name; (d) temperature parameter; and (e) maximum length of the prediction outcome (max_tokens parameter). The completion is highlighted in various colors to visually demonstrate how confidently the fine-tuned model responded to each word. The closer to green, the closer the log probability is to 0, and the closer to red, the closer it is to negative infinity. In a small window, the model's confidence in each token is displayed as a percentage.

Additionally, there are several parameters that need to be considered, and the authors confirmed that two hyperparameters ('temperature' and 'max_tokens') can influence overall performance on the prediction outcome of GPT models. Specifically, the 'temperature' parameter controls randomness in the generated output. Lower values make the completions less random. As the temperature approaches zero, the model becomes more deterministic and tends to produce repetitive output. The authors set the 'temperature' to 0.2 based on the results of sensitivity analysis. Separately, the 'max_tokens' parameter limits the number of tokens (words or subwords) in the generated text during language

generation tasks. Setting a specific value for 'max_tokens' limits the length of the generated output, impacting the completeness and context of the response. If set too low, it might result in truncated or incomplete responses, while a high value could lead to overly lengthy and potentially less coherent outputs. Choosing an appropriate 'max_tokens' value is crucial to obtaining desired results in terms of length and relevance. Thus, the authors tested the length of 'max_tokens' from 4 to 50 and observed that the 10 consistently produces reliable results.

3.2. Saliency Visualization of Accident Attributes in Unstructured Free-Text Data

Motivated by the desire to unravel the inner workings of GPT models and address their inherent black-box nature, the authors further developed a model to shed light on the transparent process of output generation. By systematically assessing the impact of individual words within an input sentence on the final output, this paper strives to enhance interpretability and provide users with a clearer understanding of the model's prediction accuracy. This is accomplished by strategically manipulating the location of words within a sentence, enabling the discernment of their unique contributions to the overall output. Each word is removed from its respective position, and the sentence is reassembled to obtain prediction accuracy. The Importance Score for a specific word in the sentence, denoted as w_i, is computed as follows:

$$\begin{aligned} Importance\ Score_i &= 1 - \tfrac{P(S-\{w_i\})}{P(S)} \\ &= 1 - P(w_i{}^c|S) \\ where, \quad S &= \{w_1,\ w_2,\ w_3,\ \cdots,\ w_n\} \end{aligned} \tag{1}$$

- S represents the set of all word elements contained in the sentence.
- w represents an individual word in the set S, with the subscript indicating the positional information of that word.

In this equation, $P(S - \{w_i\})$ represents the probability of an accident that can occur within the sentence S occurring when the researchers remove or exclude the element, a word, w_i from it. $P(S)$ represents the probability of an accident that can occur within the sentence S happening in its entirety. The $Importance\ Score_i$ essentially quantifies the significance or importance of the element w_i within the context of event S. It is calculated as the difference between 1 and the product of two probabilities: one representing the occurrence of S with the element w_i removed and the other representing the overall probability of S. Ultimately, it can be summarized as $1 - P(w_i{}^c|S)$ by the properties of the set difference and conditional probability. If log probability is used, then the values should be normalized to a range between 0 and 1 using exponential transformation:

$$Importance\ Score_i = 1 - e^{P(S-\{w_i\})-P(S)} \tag{2}$$

Algorithm 1 illustrates the pseudocode to operate saliency visualization. This methodology allows for a quantitative comparison of the impact of each word on the final output, providing insights into the black-box behavior of the GPT model. Using this methodology, the researchers can obtain a set of Importance Scores for all the words that make up a sentence. Below is the detailed description of Algorithm 1:

Algorithm 1 Computing Importance Scores for words in a sentence

Input: A sentence of text data *Sent*;
Output: List *Scores*;
1 Split *Sent* into a List of words S; $S = \{w_1, w_2, w_3, \cdots, w_n\}$
2 Declare *Score* variables on the disk; $Score_1, Score_2, Score_3, \cdots, Score_n$
3 **for** i from 1 to n **do**
4 $S_i = S - \{w_i\}$
5 $Sent_i = Concat(S_i)$
6 $Score_i = 1 - \frac{P(Sent_i)}{P(Sent)}$
7 **end for**
8 Assign $Scores = \{Score_1, Score_2, Score_3, \cdots, Score_n\}$

The algorithm commences by performing tokenization on the input sentence, *Sent*, resulting in the generation of a list of words, denoted as S. It subsequently initializes score variables $Score_1$ through $Score_n$ for each word in the sentence. The algorithm then proceeds to iterate through each word, eliminating the i-th word w_i from S, thereby creating modified sentences S_i. The scores $Score_i$ for each modified sentence S_i are calculated using the formula:

$$Score_i = 1 - \frac{P(S_i)}{P(Sent_i)},$$

where $P(S_i)$ represents the probability of the modified sentence, and $P(Sent)$ represents the probability of the original sentence. Finally, the resulting scores $Score_1$ through $Score_n$ are stored in the *Scores* list for subsequent analysis and interpretation.

This algorithm provides a systematic approach to assess the impact of individual words within a given sentence. It calculates a score for each word by considering the change in sentence probability when that word is removed. These scores can be valuable for various NLP tasks, such as keyword extraction, text summarization, and sentiment analysis.

4. Results

4.1. Data

The authors obtained a total of approximately 15,000 construction accident report data in South Korea from 1990s to 2022. The data were categorized into a total of six types: caught-in-between, cut, falls, struck-by, trips, and others. The "others" includes the accident types (e.g., overexertion, occupational diseases, and fire) that cannot be classified in the remaining five accident types and takes small proportions, which can decrease the reliability of the classification results. The selection of the six types of accidents in this study is based on a comprehensive analysis of accident reports, existing literature, industry standards, and common occurrences in construction sites [13,29,43]. These six accident types encompass a broad spectrum of incidents that are frequently reported and have significant implications for construction site safety.

Addressing the issue of data imbalance is indeed a critical task in order to improve the performance of a classification model. Thus, the authors downsampled by randomly selecting 1000 records from each category to resolve imbalanced data issues. Fine-tuning involves transferring learning onto a pre-trained model that has been trained on a large amount of data, and it often performs well even with a limited amount of data. Moreover, transfer learning is adopted as a viable solution to overcome data scarcity issues.

Table 2 shows variables and features from construction accident reports. The report contains two input data: unstructured and structured text data. The structured text data contain contextual information (e.g., date, time, weather, temperature, type of construction) during accident events. The unstructured, free-text, data include narrative details of the accident. Figure 4 presents an example of the unstructured data. It describes the details of accident circumstances. Additionally, the portions in the accident circumstances that directly provided information about the occurrence of accidents or allowed for the direct identification of accident types was removed. For example, the sentence highlighted in red

in Figure 4, "As the backhoe was reversing to return, the signalman, who was attempting to guide the backhoe, got entangled in its caterpillar", directly mentions the word "got entangled" which directly reveals the injury type.

Table 2. Variables and features from construction accident reports.

Variable		Type	Feature
Output	Event type	Categorical (6 events)	Caught-in-between, Cut, Falls, Struck-by, Trips, Others.
Input (Unstructured)	Narrative details of accidents	Text (Accident details)	Unstructured text data
Input (Structured)	Date	Categorical (4 seasons)	Spring, Summer, Fall, Winter
	Time	Categorical (5 windows)	Dawn, Morning, Daytime, Afternoon, Night
	Weather	Categorical	Sunny, Snowy, Rainy, Windy, Foggy, Cloudy
	Temperature	Numerical (Integer)	. . ., −3 °C, −2 °C, −1 °C, 0 °C, 1 °C, 2 °C, 3 °C, . . .
	Humidity	Numerical (Percentage in natural number)	0%, 1%, 2%, 3%, . . .
	Type of construction	Categorical	File drive, Building blocks, Formwork installation, . . .
	Method of construction	Categorical	Firewall installation, Doka form installation, Gang form dismantling, . . .
	Nationality	Categorical	Republic of Korea, Malaysia, USA, Vietnam, . . .
	Age	Numerical (Natural number)	20, 21, 22, . . ., 79
	Work progress	Categorical (Ranges of percentage)	0~9%, 10~19%, 20~29%, . . ., 90~100%

While the backhoe in the tank unit 2 area was being repositioned to respond to a support request in the tank unit 1 area, the request was subsequently canceled. As the backhoe was reversing to return, the signalman, who was attempting to guide the backhoe, got entangled in its caterpillar.

Figure 4. A sample text to illustrate the unstructured data (stop words highlighted in red color).

4.2. Baseline Models

To evaluate the performance of the developed 2.0 and 3.0 fine-tuned GPT-based method, this study selected three base classifiers, namely TF-IDF, CNN, and BERT, for comparison based on the literature [6,11,14]. The baseline models play an important role in classification research as they provide essential benchmarks for evaluating the performance of more complex or new models. By implementing existing algorithms or traditional AI approaches, baseline models set a standard against which the effectiveness of novel methods can be measured. The baseline models serve as reference points for assessing improvements and offer insights into the challenges of a specific task. The model's performance underwent assessment through a designated test set. To ensure robust evaluation, distinct training and testing splits were formulated for outcome categories. 90% of the reports were allocated for training purposes, while the remaining 10% were reserved

for testing the model's predictive capabilities [44,45]. Additionally, for the optimization of parameters in the GPT model, a validation set was meticulously generated for each outcome. This involved randomly isolating 10% of the training set, providing a dedicated dataset for fine-tuning and enhancing the model's precision in predicting specific accident types. This allocation, termed a validation set, plays a crucial role in refining model performance by facilitating hyperparameter tuning and mitigating overfitting risks [46,47].

1. TF-IDF, is a statistical measure that indicates how important a word is within a specific document in a collection of documents [48]. Typically used in information retrieval and text mining [49], TF-IDF provides weightings but does not involve learning on its own. However, it can be integrated with machine learning techniques and has surprisingly demonstrated strong performance in prior research, earning its selection as a benchmark model. Both the stochastic gradient descent (SGD) classifier and support vector machine (SVM) classifier were trained using the weights obtained from the TF-IDF vectorizer, and their accuracy was measured. In the training of the SGD classifier, the performance of four different kernels (radial basis function, linear, poly, sigmoid) was compared. Among these, the Linear Kernel yielded the highest accuracy. In the training of the SVM classifier, nine different loss parameters (logistic, hinge, modified huber, squared hinge, perceptron, squared error, huber, insensitive, squared epsilon insensitive) were utilized. Among these, the logistic loss parameter resulted in the highest accuracy.
2. CNN specializes in deep learning models for image and grid data processing [50,51]. They use convolutional layers to detect features, pooling layers to downsample, and fully connected layers for classification. CNNs excel in tasks such as image recognition and have wide applications in computer vision and beyond. In the dataset, the following parameters yielded the most optimal results. The number of epochs was set to 8, following experimentation in the range of 6 to 100, while the batch size was configured to 64, tested across a range from 32 to 128. An embedding dimension of 300 was used, accompanied by 100 filters and filter sizes of 2, 3, and 4. A dropout rate of 0.5 was applied during training. The optimizer employed was Adam, and the criterion was defined as CrossEntropyLoss, since the task is multiclass classification. The text was tokenized using the spacy.load ("ko_core_news_sm") tokenizer supported by spacy, which is the Python library. The pre-trained model used for the tokenizer is "ko_core_news_sm". These parameters were crucial in achieving the desired outcomes, as highlighted in the provided data.
3. BERT is indeed a type of LLM, similar to GPT, but it's a smaller model with only 0.3 billion parameters compared to GPT-3.0's 175 billion parameters [39,52]. The number of parameters in LLMs is proportional to the size of the training dataset. To investigate whether there is a performance difference in the dataset based on the number of parameters, experiments were conducted using BERT. The experiments were conducted using Python and the Keras TensorFlow package [52,53]. The training of the BERT model was based on the BERT-Base model available from Google on GitHub. The BERT-Base model supports 104 languages and consists of 12 layers, 768 hidden units per layer, 12 attention heads, and 110 million parameters. For optimization, the RAdam optimizer was chosen, incorporating a weight decay of 0.0025 [54,55]. Since the task involves multi-class classification, the sparse categorical cross-entropy loss function was employed. Furthermore, the following parameters that produced the best performance were used in this paper: sequence length (128), batch size (16), epochs (8), learning rate (0.00001), optimizer (Adam).

4.3. Experiment Results

The results of the baseline models (TF-IDF, CNN, BERT) and fine-tuned GPT 2.0/3.0 models can be seen in Table 3. Each column presents the results for a specific outcome category, and the last column represents the average value across all categories. Identical performance metrics were applied to all models, encompassing precision, recall, F1 score,

and accuracy. Precision measures the ratio of correctly identified positive instances to all instances classified as positive by the classifier, while Recall quantifies the ratio of correctly identified positive instances to the total number of relevant samples (i.e., all instances that should have been correctly identified as positive) [56,57]. The F1 score can be considered as a comprehensive assessment metric that amalgamates Precision and Recall [58].

Table 3. Model performances.

Classifier (Data Type)	Performance Metrics	Caught-in-between	Cut	Falls	Struck-by	Trips	Others	Total	Accuracy (%)
TF-IDF + SGD (Unstructured data only)	Precision	0.52	0.80	0.56	0.36	0.59	0.35	0.53	51.34
	Recall	0.59	0.49	0.60	0.48	0.58	0.32	0.51	
	F1	0.55	0.61	0.58	0.41	0.59	0.33	0.51	
TF-IDF + SVM (Unstructured data only)	Precision	0.54	0.74	0.55	0.38	0.59	0.43	0.54	53.34
	Recall	0.62	0.55	0.66	0.49	0.54	0.32	0.53	
	F1	0.58	0.63	0.60	0.43	0.57	0.36	0.53	
CNN (Unstructured data only)	Precision	0.51	0.74	0.56	0.40	0.47	0.43	0.52	52.10
	Recall	0.58	0.67	0.57	0.42	0.54	0.33	0.52	
	F1	0.54	0.71	0.57	0.41	0.51	0.38	0.52	
BERT (Unstructured data only)	Precision	0.51	0.78	0.67	0.42	0.63	0.32	0.56	54.33
	Recall	0.61	0.60	0.56	0.59	0.59	0.30	0.54	
	F1	0.56	0.67	0.61	0.49	0.61	0.31	0.54	
GPT-2.0 (Unstructured data only)	Precision	0.53	0.72	0.62	0.22	0.60	0.52	0.54	56.40
	Recall	0.48	0.63	0.57	0.10	0.71	0.58	0.51	
	F1	0.50	0.67	0.59	0.14	0.65	0.55	0.52	
GPT-3.0 (Unstructured + structured data)	Precision	0.71	0.93	0.74	0.78	0.87	0.40	0.74	67.22
	Recall	0.71	0.83	0.64	0.62	0.45	0.80	0.68	
	F1	0.71	0.88	0.68	0.69	0.60	0.54	0.68	
GPT-3.0 (Unstructured data only)	Precision	0.80	0.91	0.88	0.74	0.84	0.76	0.82	82.33
	Recall	0.79	0.87	0.82	0.83	0.88	0.64	0.81	
	F1	0.80	0.89	0.85	0.79	0.86	0.75	0.82	

Overall, the results show that pretrained GPT-3 with unstructured free-text data is outperformed almost everywhere. With 82.33% accuracy the best performer is the "Cut" class (F1 score 0.88). The "Others" class presents low F1 scores in all classifiers. Since the "Others" class includes different injury scenarios (e.g., overexertion, occupational diseases, and fire) that cannot be included in the other major five accident outcomes, it may be hard to find certain attributes in injury scenarios.

Figure 5 presents a confusion matrix of the GPT-3.0 fine-tuned model for both combined structured and unstructured data and unstructured data alone. Initially, the authors anticipated that incorporating more data would enhance the prediction performance by providing additional contextual information; the amalgamation of structured and unstructured data did not exhibit superior performance compared to utilizing unstructured data alone. The prediction performance was also affected by the version of GPT models. GPT 3.0 outperforms GPT 2.0 in almost every class. GPT 2.0 showed very low performance in the "Struck-by" class.

4.4. Saliency Visualization Results for Accident Types

As described in the Method section, a novel approach was developed to learn which parts of the input are critically influenced by the predictive outcomes. This involved computing the importance of individual words for obtaining a correct answer by deleting each word from words in a sentence. A white background indicates that redacting the word did not reduce accuracy. In other words, the greater the influence of a word on the output, the more prominently it is highlighted. The level of color highlighted in the text indicates that an amount of accuracy is reduced when the word is removed. Overall, the highlighted words appear to be the accident precursors (e.g., dust, particles, flash, chemical, oil) as mentioned in [6].

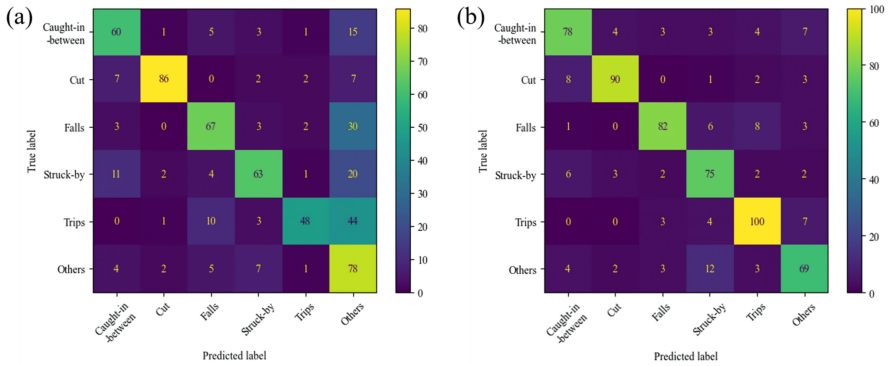

Figure 5. A confusion matrix of the GPT-3.0 fine-tuned model for (**a**) both combined structured and unstructured data and (**b**) unstructured data alone.

A visual example and a graph regarding the Importance Score of this method is shown in Figure 6. The main accident precursors (e.g., "ear", "leaving", "while", "to extract", "staircase", "the 4m", "pipe") are highlighted. Additionally, words that indicate situational status such as "On the floor" and "top" are also included. It shows that the developed method not only expresses direct factors leading to accidents but also considers contextual attributes for causing accidents. This result appears to be attributed to the transformer in the fine-tuned model, which interprets the query of the transformer in consideration of the position of words and their relationships. This indicates that the fine-tuned GPT model not only expresses direct factors leading to accidents but also considers indirect situational context for causing accidents.

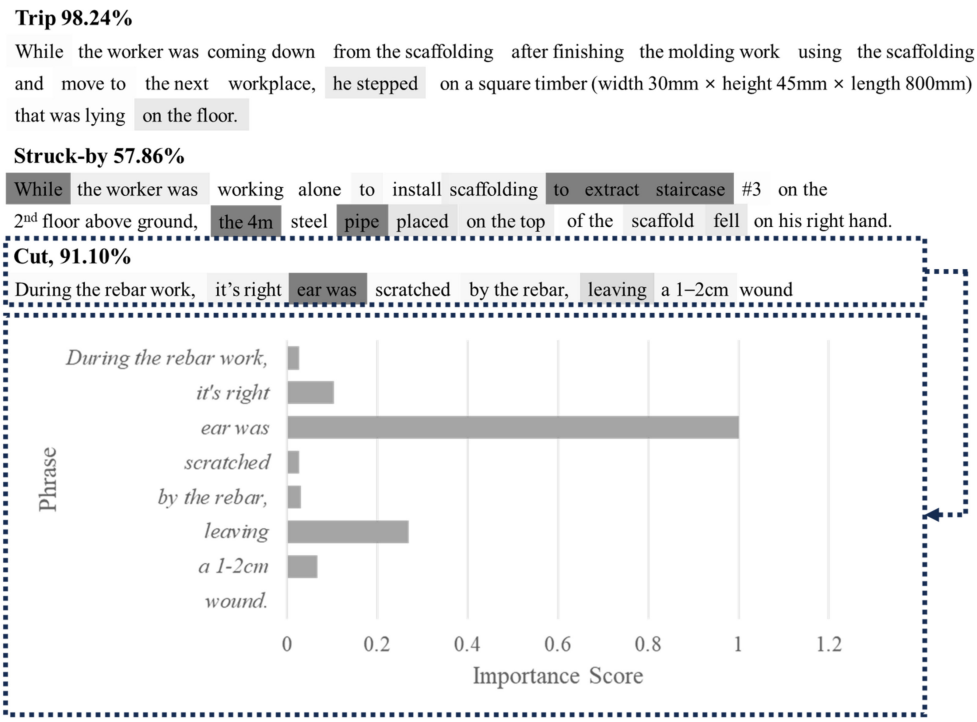

Figure 6. Examples of word saliency visualization and a bar graph of Importance Score. (The presented image demonstrates three sentences highlighted in respect of saliency visualization); (A bar graph below depicts quantitative values of importance score.)

5. Discussion

The authors developed a novel fine-tuned GPT model to predict construction accident outcomes. The GPT mode excels in tasks involving accident-related text data by capturing context, and employing end-to-end learning without the manual feature engineering of traditional machine learning and deep learning algorithms. Additionally, the introduction of a saliency visualization method enhances the interpretability of the model's decision-making process. The research makes a substantial contribution to the domain of construction safety through the application of NLP models, with a particular emphasis on utilizing generative AI models. The findings of the study underscore the superior performance of fine-tuned GPT models, specifically highlighting the effectiveness of GPT-3.0, when compared to traditional models such as TF-IDF, CNN, BERT, and GPT-2.0. The achieved overall accuracy of 82.33% signals the potential of the fine-tuned GPT model for predicting construction accident types from unstructured text data.

The proposed saliency visualization method is a noteworthy addition to the research, offering a solution to the black-box nature of GPT models. By systematically assessing the impact of individual words on prediction outcomes, the method enhances interpretability, suggesting unique insights into the decision-making process of the model. This aspect of the research contributes to building trust in the model's predictions, minimizing the labor required to extract valuable insights from free-form text, and generating dependable safety predictions from accident reports. It also has great potential to understand accident causal relationships that support existing correlational evidence in attribute-based accident analytics studies [15,23]. Furthermore, this research offers actionable insights for practical application at construction sites. Implementation involves incorporating construction documents (e.g., daily work reports and text data regarding scheduling, estimating, and specifications) to proactively predict potential accidents. By utilizing easy-to-use generative AI at construction job sites, it becomes possible to automatically assess the anticipated accident types and discern the influential attributes or textual elements. This approach facilitates a preventive strategy by leveraging the model's predictive capabilities to mitigate safety risks. Such integration of advanced LLM models into daily construction safety management holds promise for enhancing on-site safety measures.

While this study presents significant advancements in the prediction of construction accidents through fine-tuned GPT models, several limitations warrant consideration. The fine-tuning process of the GPT models on a specific dataset raises concerns about its adaptability to unforeseen circumstances and diverse construction environments. Incorporating multimodal data, such as images and videos from construction sites, could further enrich the model's learning capabilities and provide a more comprehensive understanding of accidents. Additionally, the downsampling approach employed to address data imbalance, while effective, may inadvertently omit crucial information related to infrequent accident types, impacting the model's robustness. Furthermore, the study's focus on the Korean construction context may limit the generalizability of the findings to construction sites in other geographical regions with distinct safety reporting practices and cultural nuances. Including and testing more data from different countries and languages can mitigate the generalizability and reliability concerns of the model. Separately, the study found that GPT-3.0 consistently outperformed GPT-2.0, mainly due to its substantial increase in parameters, with 175 billion compared to GPT-2.0's 1.5 billion. This increased scale allowed GPT-3.0 to capture more intricate patterns, resulting in significantly improved performance. This highlights the importance of using the latest language models to achieve optimal results, guiding practitioners and researchers in model selection. Lastly, the interpretability of the saliency visualization method, while a valuable addition, may be limited in capturing subtle relationships within complex textual data. Incorporating additional advanced explainable AI methods (e.g., layer-wise relevance propagation and local interpretable model-agnostic explanations) is needed to enhance the interpretability of the accident prediction model.

The authors anticipate the developed fine-tuned GPT model to play a pivotal role in revolutionizing construction safety practices. As the construction industry increasingly embraces data-driven approaches, the model's predictive capabilities offer a transformative impact on accident prevention. Proactively identifying potential accidents and safety risks through the analysis of unstructured free-text data enables construction sites to implement targeted interventions, thereby enhancing overall safety outcomes. The model's adaptability to dynamic environments, potential integration with multimodal data sources, and real-time monitoring mechanisms position it as a versatile tool for safety professionals and decision-makers. Moreover, the saliency visualization method contributes to model interpretability, fostering collaboration between AI technologies and human expertise in safety management. Collaborative efforts with industry stakeholders, safety professionals, and policymakers can facilitate the real-world integration of the proposed model, ensuring its practical applicability and effectiveness in enhancing safety management practices. The study represents a significant step forward in leveraging the GPT model for construction safety. By addressing key challenges and offering innovative solutions, the research opens avenues for future investigations that can further refine and expand the applicability of these models in real-world safety management contexts.

6. Conclusions

This study presents the remarkable potential of leveraging fine-tuned GPT models for predicting construction accidents from unstructured free-text data, achieving an outstanding overall accuracy of 82.33%. The superior performance of these models compared to baseline AI model approaches is evident, underscoring their efficacy in enhancing predictive capabilities. The introduction of the saliency visualization method further elevates the model's interpretability, providing valuable insights into accident attributes. Beyond its performance metrics, the significance of this research extends to its practical application in real-world safety management. The proposed model, with its promising adaptability for dynamic updating and potential integration with multimodal data, emerges as a proactive tool for accident prevention in the construction industry. As the field of construction safety increasingly embraces advanced NLP models, this study contributes to the broader transformation toward data-driven and proactive safety practices. This research, therefore, holds pivotal importance for the global scientific community by advancing the understanding of how state-of-the-art language models can revolutionize safety prediction in a construction industry. Its implications reach beyond the construction sector, influencing the broader landscape of safety management across diverse domains.

Author Contributions: Conceptualization, C.R.A., T.O., B.Y. and S.P.; methodology, B.Y. and S.P.; writing—original draft preparation, J.K. and B.Y.; writing—review and editing, B.Y., J.K. and C.R.A.; funding acquisition, T.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Research Foundation of Korea, grant number 2021R1I1A2050912.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are not publicly available due to potential confidentiality concerns.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rydning, D.R.-J.G.-J.; Reinsel, J.; Gantz, J. The digitization of the world from edge to core. *Fram. Int. Data Corp.* **2018**, *16*, 1–28.
2. Data Growth Worldwide 2010–2025. Available online: https://www.statista.com/statistics/871513/worldwide-data-created/ (accessed on 23 October 2023).
3. EDER Unstructured Data and the 80 Percent Rule. Breakthrough Analysis. 2008. Available online: https://breakthroughanalysis.com/2008/08/01/unstructured-data-and-the-80-percent-rule/ (accessed on 1 August 2008).

4. Woods, D.D.; Patterson, E.S.; Roth, E.M. Can We Ever Escape from Data Overload? A Cognitive Systems Diagnosis. *Cogn. Tech. Work* **2002**, *4*, 22–36. [CrossRef]
5. Henke, N.; Jacques Bughin, L. *The Age of Analytics: Competing in a Data-Driven World*; McKinsey Global Institute Research: New York, NY, USA, 2016.
6. Baker, H.; Hallowell, M.R.; Tixier, A.J.-P. Automatically learning construction injury precursors from text. *Autom. Constr.* **2020**, *118*, 103145. [CrossRef]
7. Liu, Y.; Wang, J.; Tang, S.; Zhang, J.; Wan, J. Integrating information entropy and latent Dirichlet allocation models for analysis of safety accidents in the construction industry. *Buildings* **2023**, *13*, 1831. [CrossRef]
8. Lukic, D.; Littlejohn, A.; Margaryan, A. A framework for learning from incidents in the workplace. *Saf. Sci.* **2012**, *50*, 950–957. [CrossRef]
9. Sanne, J.M. Incident reporting or storytelling? Competing schemes in a safety-critical and hazardous work setting. *Saf. Sci.* **2008**, *46*, 1205–1222. [CrossRef]
10. Ganguli, R.; Miller, P.; Pothina, R. Effectiveness of natural language processing based machine learning in analyzing incident narratives at a mine. *Minerals* **2021**, *11*, 776. [CrossRef]
11. Fang, W.; Luo, H.; Xu, S.; Love, P.E.; Lu, Z.; Ye, C. Automated text classification of near-misses from safety reports: An improved deep learning approach. *Adv. Eng. Inform.* **2020**, *44*, 101060. [CrossRef]
12. Wu, H.; Zhong, B.; Medjdoub, B.; Xing, X.; Jiao, L. An ontological metro accident case retrieval using CBR and NLP. *Appl. Sci.* **2020**, *10*, 5298. [CrossRef]
13. Li, J.; Wu, C. Deep Learning and Text Mining: Classifying and Extracting Key Information from Construction Accident Narratives. *Appl. Sci.* **2023**, *13*, 10599. [CrossRef]
14. Zhang, J.; Zi, L.; Hou, Y.; Deng, D.; Jiang, W.; Wang, M. A C-BiLSTM approach to classify construction accident reports. *Appl. Sci.* **2020**, *10*, 5754. [CrossRef]
15. Tixier, A.J.-P.; Hallowell, M.R.; Rajagopalan, B.; Bowman, D. Application of machine learning to construction injury prediction. *Autom. Constr.* **2016**, *69*, 102–114. [CrossRef]
16. Zhang, F.; Fleyeh, H.; Wang, X.; Lu, M. Construction site accident analysis using text mining and natural language processing techniques. *Autom. Constr.* **2019**, *99*, 238–248. [CrossRef]
17. Baker, H.; Hallowell, M.R.; Tixier, A.J.-P. AI-based prediction of independent construction safety outcomes from universal attributes. *Autom. Constr.* **2020**, *118*, 103146. [CrossRef]
18. Locatelli, M.; Seghezzi, E.; Pellegrini, L.; Tagliabue, L.C.; Di Giuda, G.M. Exploring natural language processing in construction and integration with building information modeling: A scientometric analysis. *Buildings* **2021**, *11*, 583. [CrossRef]
19. Lee, J.-K.; Cho, K.; Choi, H.; Choi, S.; Kim, S.; Cha, S.H. High-level implementable methods for automated building code compliance checking. *Dev. Built Environ.* **2023**, *15*, 100174. [CrossRef]
20. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. *Improving Language Understanding with Unsupervised Learning*; Technical Report; OpenAI: San Francisco, CA, USA, 2018.
21. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*; NIPS: New Orleans, LA, USA, 2017; Volume 30.
22. Pal, A.; Lin, J.J.; Hsieh, S.-H.; Golparvar-Fard, M. Automated vision-based construction progress monitoring in built environment through digital twin. *Dev. Built Environ.* **2023**, *16*, 100247. [CrossRef]
23. Esmaeili, B.; Hallowell, M. Attribute-Based Risk Model for Measuring Safety Risk of Struck-By Accidents. In *Proceedings of the Construction Research Congress 2012*; American Society of Civil Engineers: West Lafayette, IN, USA, 2012; pp. 289–298.
24. Jeong, J.; Jeong, J. Quantitative Risk Evaluation of Fatal Incidents in Construction Based on Frequency and Probability Analysis. *J. Manag. Eng.* **2022**, *38*, 04021089. [CrossRef]
25. Kang, Y.; Cai, Z.; Tan, C.-W.; Huang, Q.; Liu, H. Natural language processing (NLP) in management research: A literature review. *J. Manag. Anal.* **2020**, *7*, 139–172. [CrossRef]
26. Hallowell, M.R. Safety-Knowledge Management in American Construction Organizations. *J. Manag. Eng.* **2012**, *28*, 203–211. [CrossRef]
27. Huang, X.; Hinze, J. Owner's Role in Construction Safety. *J. Constr. Eng. Manag.* **2006**, *132*, 164–173. [CrossRef]
28. Ding, Y.; Ma, J.; Luo, X. Applications of natural language processing in construction. *Autom. Constr.* **2022**, *136*, 104169. [CrossRef]
29. Chokor, A.; Naganathan, H.; Chong, W.K.; El Asmar, M. Analyzing Arizona OSHA injury reports using unsupervised machine learning. *Procedia Eng.* **2016**, *145*, 1588–1593. [CrossRef]
30. Tixier, A.J.-P.; Hallowell, M.R.; Rajagopalan, B. Construction Safety Risk Modeling and Simulation. *Risk Anal.* **2017**, *37*, 1917–1935. [CrossRef]
31. Hsieh, H.-F.; Shannon, S.E. Three Approaches to Qualitative Content Analysis. *Qual. Health Res.* **2005**, *15*, 1277–1288. [CrossRef] [PubMed]
32. Tixier, A.J.-P.; Hallowell, M.R.; Rajagopalan, B.; Bowman, D. Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports. *Autom. Constr.* **2016**, *62*, 45–56. [CrossRef]
33. Cheng, M.-Y.; Kusoemo, D.; Gosno, R.A. Text mining-based construction site accident classification using hybrid supervised machine learning. *Autom. Constr.* **2020**, *118*, 103265. [CrossRef]

34. Kim, T.; Chi, S. Accident Case Retrieval and Analyses: Using Natural Language Processing in the Construction Industry. *J. Constr. Eng. Manag.* **2019**, *145*, 04019004. [CrossRef]
35. Kim, H.; Jang, Y.; Kang, H.; Yi, J.-S. A Study on Classifying Construction Disaster Cases in Report with CNN for Effective Management. In *Proceedings of the Construction Research Congress 2022*; American Society of Civil Engineers: Arlington, VA, USA, 2022; pp. 483–491.
36. Goh, Y.M.; Ubeynarayana, C.U. Construction accident narrative classification: An evaluation of text mining techniques. *Accid. Anal. Prev.* **2017**, *108*, 122–130. [CrossRef]
37. Liu, P.; Shi, Y.; Xiong, R.; Tang, P. Quantifying the reliability of defects located by bridge inspectors through human observation behavioral analysis. *Dev. Built Environ.* **2023**, *14*, 100167. [CrossRef]
38. Zhang, M.; Li, J. A commentary of GPT-3 in MIT Technology Review 2021. *Fundam. Res.* **2021**, *1*, 831–833. [CrossRef]
39. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
40. Balkus, S.V.; Yan, D. Improving short text classification with augmented data using GPT-3. In *Natural Language Engineering*; Cambridge University Press: Cambridge, UK, 2022; pp. 1–30.
41. Niu, Z.; Zhong, G.; Yu, H. A review on the attention mechanism of deep learning. *Neurocomputing* **2021**, *452*, 48–62. [CrossRef]
42. Lin, T.; Wang, Y.; Liu, X.; Qiu, X. A survey of transformers. *AI Open* **2022**, *3*, 111–132. [CrossRef]
43. Alkaissy, M.; Arashpour, M.; Golafshani, E.M.; Hosseini, M.R.; Khanmohammadi, S.; Bai, Y.; Feng, H. Enhancing construction safety: Machine learning-based classification of injury types. *Saf. Sci.* **2023**, *162*, 106102. [CrossRef]
44. Hastie, T.; Friedman, J.; Tibshirani, R. *The Elements of Statistical Learning*; Springer Series in Statistics; Springer: New York, NY, USA, 2001; ISBN 978-1-4899-0519-2.
45. Stone, M. Cross-Validatory Choice and Assessment of Statistical Predictions. *J. R. Stat. Soc. Ser. B Methodol.* **1974**, *36*, 111–133. [CrossRef]
46. Ryan, K.N.; Bahhur, B.N.; Jeiran, M.; Vogel, B.I. Evaluation of augmented training datasets. In *Proceedings of the Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXXII*; SPIE: Bellingham, WA, USA, 2021; Volume 11740, pp. 118–125.
47. DeVries, T.; Taylor, G.W. Improved Regularization of Convolutional Neural Networks with Cutout. *arXiv* **2017**, arXiv:1708.04552.
48. Crnic, J. Introduction to Modern Information Retrieval. *Libr. Manag.* **2011**, *32*, 373–374. [CrossRef]
49. Schütze, H.; Manning, C.D.; Raghavan, P. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2008; Volume 39.
50. Kim, Y. Convolutional Neural Networks for Sentence Classification. *arXiv* **2014**, arXiv:1408.5882.
51. Zhang, Y.; Wallace, B. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification. *arXiv* **2015**, arXiv:1510.03820.
52. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
53. Gulli, A.; Pal, S. *Deep Learning with Keras*; Packt Publishing Ltd: Birmingham, UK, 2017.
54. Sokolov, A.; Mostovoy, J.; Ding, J.; Seco, L. *Building Machine Learning Systems to Automate ESG Index Construction*; Luis Seco Publications: Toronto, ON, Canada, 2020.
55. Won, K.; Jang, Y.; Choi, H.; Shin, S. Design and implementation of information extraction system for scientific literature using fine-tuned deep learning models. *SIGAPP Appl. Comput. Rev.* **2022**, *22*, 31–38. [CrossRef]
56. Li, J.; Zhao, X.; Zhou, G.; Zhang, M. Standardized use inspection of workers' personal protective equipment based on deep learning. *Saf. Sci.* **2022**, *150*, 105689. [CrossRef]
57. Tang, S.; Golparvar-Fard, M. Machine Learning-Based Risk Analysis for Construction Worker Safety from Ubiquitous Site Photos and Videos. *J. Comput. Civ. Eng.* **2021**, *35*, 04021020. [CrossRef]
58. Sasaki, Y. The truth of the F-measure. *Teach Tutor Mater.* **2007**, *1*, 1–5.