

Original Article

# Innovative Toxicity Detection in Metal Oxide Nanoparticles Using 1-D CNN with Transformer Architecture

V.S. Nisha<sup>1</sup>, R.S. Rimal Isaac<sup>2</sup>

<sup>1,2</sup>Department of Nanotechnology, Noorul Islam Centre for Higher Education, Tamilnadu, India.

<sup>1</sup>Corresponding Author : [nishavsakhil@gmail.com](mailto:nishavsakhil@gmail.com)

Received: 25 November 2024

Revised: 30 December 2024

Accepted: 17 January 2025

Published: 30 January 2025

**Abstract** - Nanotechnology entails the manipulation of matter at the nanoscale to generate materials with distinctive features and is employed in several fields, such as medicine, electronics, and environmental science. Engineered nanoparticles at this scale provide improved functioning, although they also raise concerns about their potential toxicity to biological systems and ecosystems. Existing toxicity detection methods frequently depend on conventional experimental techniques or machine learning models that emphasize physicochemical properties, resulting in constraints such as limited sample sizes, overfitting, and insufficient interpretability. These approaches often inadequately represent the intricate interactions between nanoparticles and biological entities, leading to incorrect predictions. To tackle these problems, we propose an innovative method employing a 1-D Convolutional Neural Network (CNN) combined with transformers aimed at enhancing the predicted accuracy of Nanoparticle (NP) toxicity evaluations. The evaluation of the proposed design demonstrates highly excellent classification outcomes, with the model attaining an impressive accuracy of 96.61%. Every detected toxic case was, in fact toxic, as evidenced by the precision of 100% and the recall of 92.85% shows how sensitive the model is to real toxic events. The balanced performance is further demonstrated by an F1 score of 96.29%, highlighting the model's efficacy in handling imbalanced toxicity data. Comparisons with existing approaches highlight the superiority of the proposed method, providing essential insights for NP toxicity prediction and facilitating the production of safer nanomaterials for various applications.

**Keywords** - Nanoparticle, Toxicity, Transformer, 1-D CNN, Metal oxide, Nanotechnology, Optics.

## 1. Introduction

The 21<sup>st</sup> century has witnessed a significant advancement in nanotechnology, a pioneering field that is transforming modern science [1]. The growing interest in nanotechnology and nanoscience is primarily due to the extensive applications that NPs provide across various industries. Nanoscale particles characterized by their substantial surface area, elevated surface energy and reactivity are recognized as exceptional catalysts [2]. They are employed in domains such as optics and photonics, electronics, magnetism and catalysis within material sciences and chemistry. Furthermore, NPs are progressively utilized in biomedicine [3].

The unique characteristics of NPs, unlike their bulk equivalents, have facilitated their incorporation into commercial products [4]. During their lifecycle nano, enabled items may emit NPs into the environment, potentially resulting in detrimental impacts on numerous organisms via many exposure pathways including inhalation, injection, skin contact and the food chain. Upon release NPs may experience transformations such as biocorona formation, agglomeration,

dissolution and alterations in shape and surface charge, which can affect their physicochemical properties, randomness and behavior [5].

Metallic oxides represent an essential domain of research among the diverse categories of NPs owing to their distinctive electrical, optical and physicochemical characteristics [6]. Multiple investigations have examined the physicochemical characteristics of Metal Oxide Nanoparticles (MeONPs), highlighting their chemical stability, antibacterial properties, and thermal, electrical and mechanical attributes. The increasing manufacturing and use of MeONPs have generated significant concerns about their potential negative impact on human health and the environment [7]. Its toxicity is affected by surface qualities such as electrical characteristics, functional groups and solubility.

NP toxicity is categorized into environmental and biological [8]. Environmental toxicity arises when NPs disperse into air, water or soil, affecting ecosystems by disrupting metabolic functions in organisms, causing



oxidative stress and impacting reproduction. Their accumulation can impair soil nutrient cycles and plant growth and pose health risks to animals and humans through inhalation.

In biological contexts, NPs can traverse multiple barriers within the human body, potentially accessing various organs and stimulating responses such as inflammation, allergic reactions, neurotoxicity, fibrosis, detrimental effects on cardiac function, prethrombotic states, pulmonary toxicity, carcinogenesis, genetic mutations and mitochondrial impairment ultimately resulting in cellular apoptosis and death, thereby reducing cell viability.

In accordance with the principles established in the European Green Deal, nanomaterials introduced to the EU market, in addition to other products and chemical substances, must adhere to the Safety and Sustainability by Design (SSbD) approach [9]. This strategy seeks to alleviate the adverse effects of these products on human health and the environment by promptly identifying and eliminating potentially toxic elements. The SSbD framework promotes the creation of innovative approach methodologies to efficiently produce data and evaluate nanoparticles in the field of nanosafety.

Nano informatics involves diverse computational methods and data science techniques designed to evaluate the risks and concerns linked to NPs in both short- and long-term exposure contexts [10]. Performing experimental toxicity analyses can be labour-intensive and expensive due to the diverse properties displayed by various NPs. Furthermore, forecasting NP toxicity poses challenges owing to the intricacies of their biological systems.

The integration of automation and optimization techniques alongside ensemble learning methods into the computational evaluation of NP characteristics, in addition to recent progress in Artificial Intelligence (AI) and Machine Learning (ML), is propelling the creation of models with enhanced predictive accuracy. A major problem in advancing ML based nano informatics is the lack of NP data and metadata coupled with dataset imbalances where an excessive number of NPs may be non-toxic. Deep Learning approaches (DL) have been proposed for identifying the impacts of nanoparticles.

The proposed research aims to tackle the significant challenges associated with NP toxicity detection through the development of a novel model that integrates a 1D CNN with transformer architecture. By utilizing the strengths of DL and attention mechanisms, this approach seeks to enhance predictive accuracy and feature representation, addressing the limitations posed by the current lack of comprehensive NP data and the prevalence of dataset imbalances. The main contributions of the proposed research are as follows:

- Introduces a novel hybrid model that combines a 1-D Convolutional Neural Network (CNN) with a transformer architecture.
- Utilizes the attention mechanisms of the transformer to capture complex correlations within nanotoxicity data.
- Offers a comprehensive comparison analysis of the suggested model relative to existing toxicity detection methods.

The subsequent sections of the paper are organized as outlined below: Section 2 presents a literature review highlighting existing works and identifying research gaps, Section 3 elaborates on the proposed model, Section 4 provides the results obtained from the study, and Section 5 provides the conclusion of the research.

## 2. Related Works

Iqra Yousaf [11] tackled the rising concerns over NP toxicity by utilizing ML models, such as Random Forests (RF), Decision Trees, and XGBoost, to predict toxicity based on physicochemical characteristics. The authors found essential parameters like the presence of oxygen atoms, particle size, surface area, dosage and exposure duration as pivotal predictors of toxicity.

Zhou et al. [12] investigated the ecotoxicity of Metallic Nanomaterials (MNMs) in aquatic ecosystems, acknowledging the challenges involved in experimental analyses due to the complex nature of material characteristics and ambient variables. They created an in-silico model termed ML-PEMST to predict MNM toxicity in several aquatic species by combining ML methodologies with physicochemical attributes, exposure parameters and species traits.

The findings revealed that illumination, exposure length, hydrodynamic diameter and primary size substantially affected MNM toxicity. Illumination significantly influenced other variables, impacting the mechanisms of toxicity. The RF model surpassed other ML methods. Yang Huang et al. (2022) [13] examined the immunological responses to MeONPs employing ML models in conjunction with high-throughput in vitro bioassays. They effectively resolved the class imbalance issue using an ensemble approach. The research revealed critical parameters, including the release of hazardous ions,  $\zeta$ -potential, electronegativity, and size, as significant predictors of toxicity.

Nilesh Anantha Subramanian and Ashok Palaniappan (2021) [14] developed machine-learning methods to predict the cytotoxicity of metal-oxide nanoparticles, emphasizing inherent and extrinsic physicochemical characteristics. By omitting in vitro traits and employing a feature space, they utilized several ML approaches, including random forest and neural networks. The models had a balanced accuracy of 96%,

with dosage and exposure duration identified as primary predictors. NanoTox, the initial open-source nanotoxicology pipeline, has been launched as a significant resource for evaluating nanoparticle toxicity. Nonetheless, the limitation was the comparatively small dataset and the issue of dataset imbalances, which they addressed by SMOTE resampling.

Farooq Ahmad et al. (2021) [15] examined the transformative impact of nanoparticle breakthroughs across many sectors, including electronics, healthcare, and medicine. They emphasized the necessity for rigorous oversight due to the possible adverse impacts of nanoproducts. The authors highlighted that conventional safety evaluation methodologies were inadequate for assessing the fast incorporation of nanomaterials. They investigated the capabilities of AI and ML to improve nano-bio-interaction modeling and post-market surveillance. The integration of machine learning with omics data yielded enhanced insights into biological activities and the safety of nanomaterials. They acknowledged the constraints of machine learning, including overfitting and the lack of interpretability in neural networks, which impeded trust in drug discovery applications.

Hengjie Yu et al. (2021) [16] examined safety issues associated with Engineered Nanoparticles (ENPs) that impeded their use in several domains. The researchers utilized ML to clarify intricate interactions between ENPs, organisms, and ecosystems, thereby improving the clarity of model predictions. Seven essential characteristics affecting cytotoxicity were discovered, with quantum dot diameter being the most prominent. The study employed various interpretation approaches to uncover general patterns and specific prediction insights.

Jossana A. Damasco et al. (2020) [17] emphasized the rapid development of nanomedicine, which employs nanomaterials for the diagnosis, medication and control of diseases such as cancer. Diverse biodegradable nanoplatfroms have emerged, presenting opportunities for tumor targeting, imaging and personalized therapy. Nonetheless, their diminutive size and distinctive physicochemical characteristics have elicited safety apprehensions especially about modified pharmacokinetics and the possibility of traversing biological barriers. Inorganic NPs, particularly those containing heavy metals provide concerns due to their toxicity and accumulation within the human body. The practical translation of these NPs was impeded by their complex biological interactions and the substantial expenses linked to safety evaluations. A primary constraint of this methodology was the challenge of executing systematic high throughput testing.

Bhavna Saina et al. (2020) [18] examined the application of nanocarriers in drug delivery systems, underscoring their potential for controlled and targeted distribution while simultaneously addressing the issue of toxicity. The study

created a computational model to predict nanocarrier toxicity through the analysis of microscopic images. Owing to the limited dataset, Generative Adversarial Networks (GAN) were utilized to produce synthetic images, which, in conjunction with actual images, facilitated the training of a CNN for toxicity classification. The analysis revealed favorable results illustrating the model's proficiency in appropriately categorizing toxic and non-toxic images.

Despite considerable progress in utilizing ML approaches to forecast NP toxicity, significant research gaps remain, particularly due to limitations inherent in existing methodologies. Numerous studies, including those utilizing RF and ensemble methods, predominantly depend on physicochemical properties and limited datasets frequently leading to overfitting and inadequate generalizability over various nanoparticle categories and contexts. The class imbalance and limited sample sizes, as emphasized by numerous studies, undermine the robustness and reliability of predictive models. Methods such as SMOTE resampling can rectify dataset imbalances, yet they do not intrinsically improve the interpretability or mechanistic comprehension of the fundamental toxicity mechanisms. Moreover, traditional models sometimes encounter issues with integrating temporal dynamics and intricate interactions between nanoparticles and biological systems, which are essential for precisely evaluating toxicity in practical applications. The suggested research employs a 1-D CNN in conjunction with transformers to overcome these constraints by including high-dimensional temporal data and improving feature extraction capabilities.

### 3. Materials and Methods

The suggested methodology initiates the collection of a dataset encompassing comprehensive information on metal oxide nanoparticles and their associated toxicity profiles. The EDA employs various data visualization techniques to evaluate distributions, identify patterns, and examine potential relationships for an initial understanding of the data. The dataset undergoes careful preprocessing to ensure its suitability for model training. Subsequent to preprocessing, the dataset is partitioned into training and test sets with the training set subjected to 10-fold cross-validation to improve model generalizability and mitigate over fitting.

The core of the framework is a hybrid architecture that integrates a 1D CNN for localized feature extraction with a Transformer module that captures long-range dependencies and contextual information in the toxicity data. Hyperparameter optimization is employed to enhance model parameters for optimal performance. To evaluate the effectiveness, performance metrics are calculated for both the training and testing datasets. The robustness and ability to efficiently generalize in recognizing NP toxicity in new data are further confirmed by independent validation on the test set. Figure 1 illustrates the architecture of the proposed method.

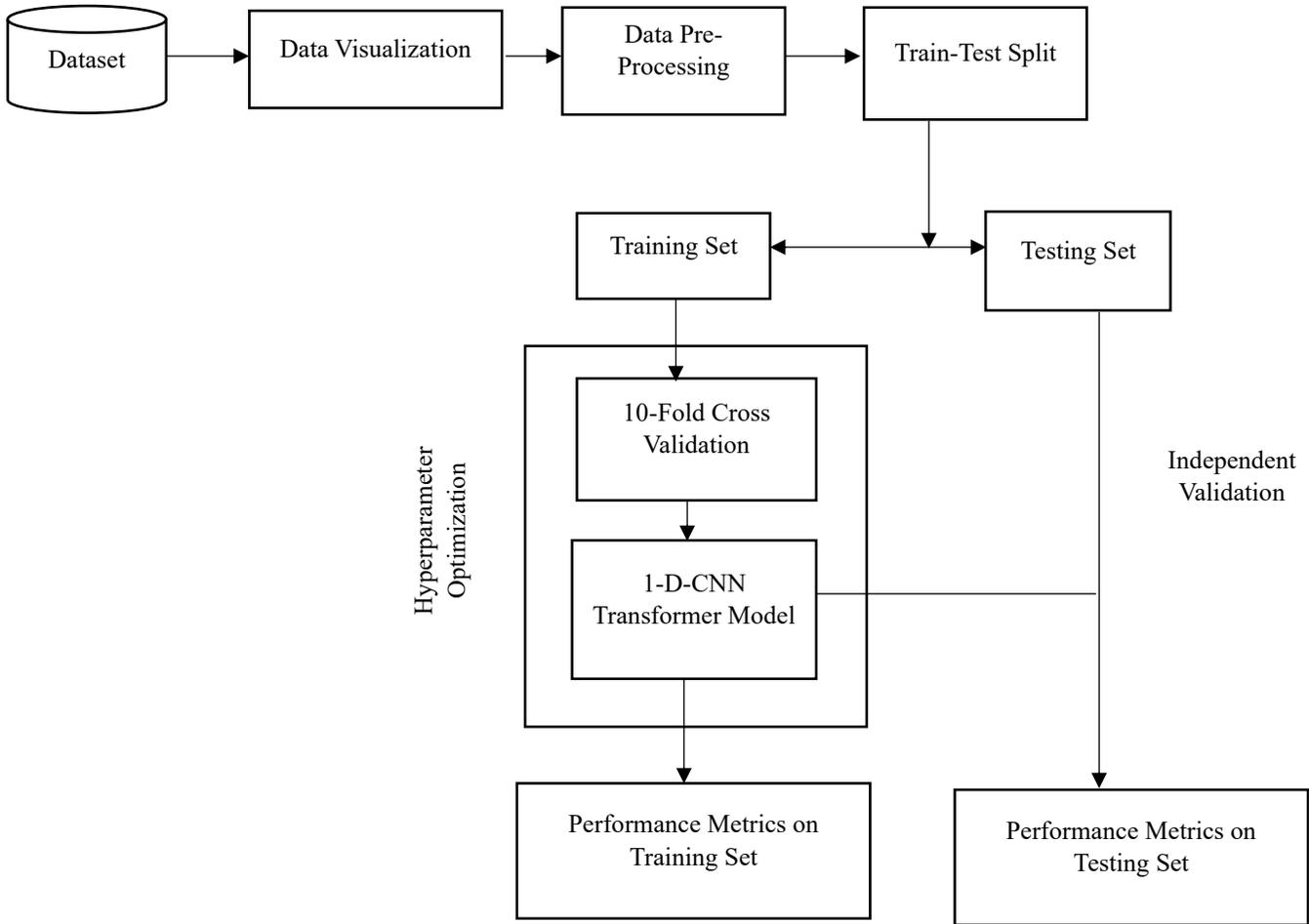


Fig. 1 Architecture of the proposed model

**3.1. Dataset**

The proposed study utilizes a toxicity dataset sourced from Kaggle to investigate the toxicological effects of various NPs. This resource can be accessed at Kaggle: Nanoparticle Toxicity Dataset. The dataset includes essential criteria for toxicity categorization, as shown in Figure 2. The dataset has rows representing distinct nanoparticle types, such as Al<sub>2</sub>O<sub>3</sub>,

together with the associated measurements for each attribute. The target variable, termed "class," indicates the toxicity status of the nanoparticles, categorizing them as either toxic or non-toxic. Figure 3 provides a visualization of the dataset distribution, showcasing the count of data points across different categories.

	NPs	coresize	hydrosized	surfcharge	surfarea	Ec	Exptime	dosage	e	NOxygen	class
0	Al <sub>2</sub> O <sub>3</sub>	39.7	267.0	36.3	64.7	-1.51	24	0.001	1.61	3	non Toxic
1	Al <sub>2</sub> O <sub>3</sub>	39.7	267.0	36.3	64.7	-1.51	24	0.010	1.61	3	non Toxic
2	Al <sub>2</sub> O <sub>3</sub>	39.7	267.0	36.3	64.7	-1.51	24	0.100	1.61	3	non Toxic
3	Al <sub>2</sub> O <sub>3</sub>	39.7	267.0	36.3	64.7	-1.51	24	1.000	1.61	3	non Toxic
4	Al <sub>2</sub> O <sub>3</sub>	39.7	267.0	36.3	64.7	-1.51	24	5.000	1.61	3	non Toxic

Fig. 2 Sample data

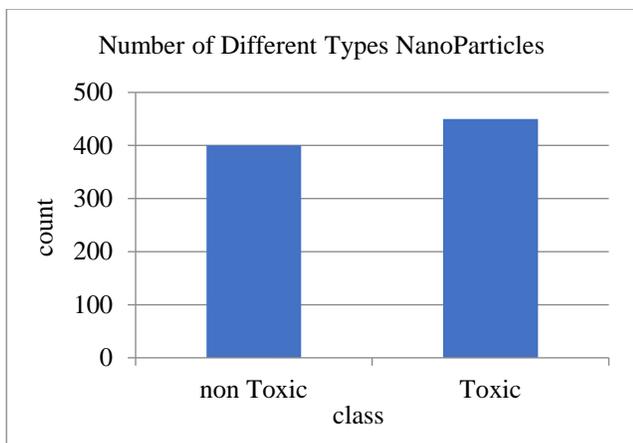


Fig. 3 Dataset distribution

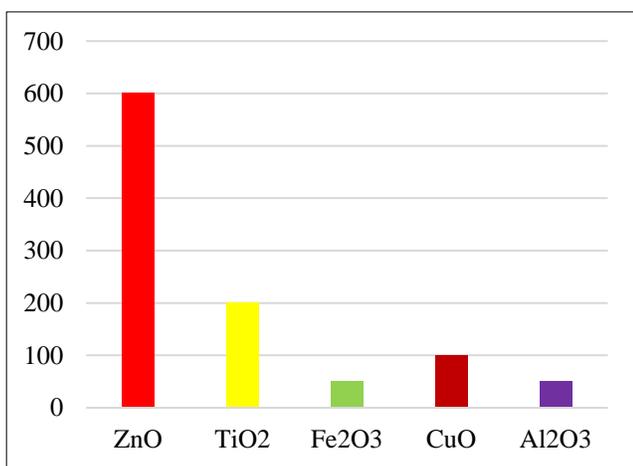


Fig. 4 Nanoparticle count

Figure 4 depicts the count of various NPs, emphasizing the dominance of ZnO nanoparticles in the dataset, which comprises around 600 entries, far surpassing the counts of other NPs. TiO<sub>2</sub> are present in a moderate quantity, with approximately 250 entries. The collection has far fewer examples of Fe<sub>2</sub>O<sub>3</sub>, CuO, and Al<sub>2</sub>O<sub>3</sub> nanoparticles, with Fe<sub>2</sub>O<sub>3</sub> exhibiting the least representation, practically absent, while CuO and Al<sub>2</sub>O<sub>3</sub> have approximately 100 and 50 cases, respectively.

Figure 5 categorizes nanoparticles into two classifications. Fe<sub>2</sub>O<sub>3</sub>, CuO, and Al<sub>2</sub>O<sub>3</sub> nanoparticles are primarily classified as toxic, with a negligible occurrence in the non-toxic category. In contrast, TiO<sub>2</sub> and ZnO nanoparticles demonstrate a more diverse distribution, with a significant fraction classified as non-toxic.

However, it is noteworthy that ZnO nanoparticle toxicity data approach 450, indicating that a substantial portion of these particles are considered toxic. This indicates that ZnO nanoparticles can exhibit both highly toxic and non-toxic properties contingent upon specific conditions or uses.

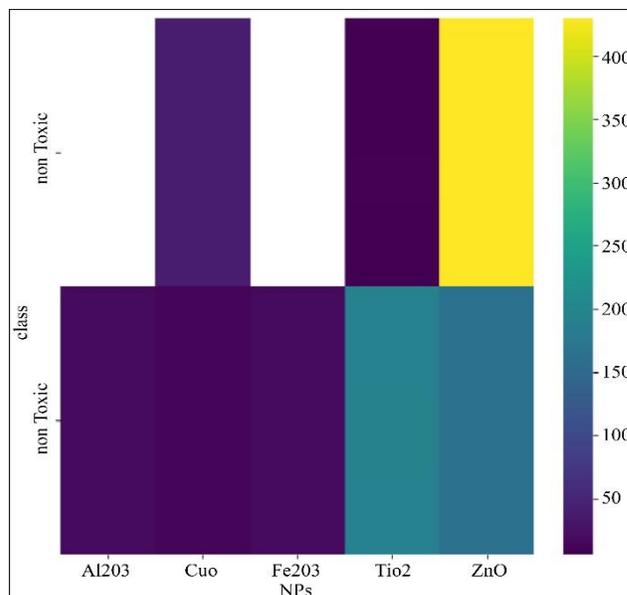


Fig. 5 NP vs Class

The boxplot, as shown in Figure 6, illustrates the distribution of core sizes across several types of NPs. ZnO demonstrates considerable variability, with a median core size of approximately 60 and a broad interquartile range reaching up to 120, in addition to several outliers. TiO<sub>2</sub> exhibit a relatively narrow size distribution, with a median near 40 and a few outliers below 20, suggesting greater uniformity in size. CuO possesses a median size of around 60, with a rather mild variation in comparison to ZnO. Fe<sub>2</sub>O<sub>3</sub> and Al<sub>2</sub>O<sub>3</sub> have minimal size distributions, with medians around 40 and negligible variance, signifying their more consistent core sizes.

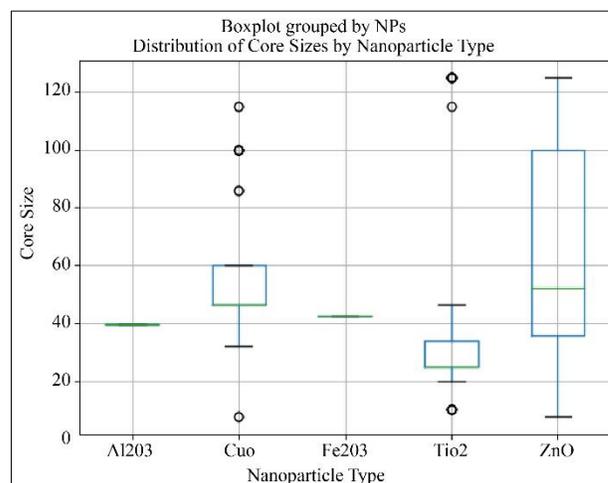


Fig. 6 Core size of nanoparticles

The statistical analysis of the dataset provides detailed insights into the properties of nanoparticles, as illustrated by numerous metrics in Figure 7, facilitating further investigations of their toxicological effects.

	coresize	hydrosize	surfcharge	surfarea	Ec	Expotime	dosage	e	NOxygen
count	881.000000	881.000000	881.000000	881.000000	881.000000	881.000000	881.000000	881.000000	881.000000
mean	56.313280	513.781385	1.642111	42.074075	-4.018127	27.459705	39.651270	1.646050	1.308740
std	33.700297	346.601373	25.635780	47.111739	0.509806	19.534667	38.163289	0.089304	0.543581
min	7.500000	74.000000	-41.600000	7.000000	-5.170000	3.000000	0.000010	1.540000	1.000000
25%	32.000000	273.400000	-11.700000	15.000000	-4.160000	12.000000	10.000000	1.650000	1.000000
50%	45.300000	327.000000	-9.300000	24.100000	-3.890000	24.000000	25.000000	1.650000	1.000000
75%	86.000000	687.000000	29.400000	42.500000	-3.890000	24.000000	50.000000	1.650000	2.000000
max	125.000000	1843.000000	42.800000	210.000000	-1.510000	72.000000	300.000000	1.900000	3.000000

Fig. 7 Statistics of dataset description

**3.2. Exploratory Data Analysis (EDA) and Visualization**

EDA is an essential phase in the data analysis process, focused on comprehending the fundamental structure and patterns inside a dataset before using machine learning models. To investigate the features of nanoparticles, this study used EDA with a number of important visualizations. Feature distribution plots reveal key trends within the data, as shown in Figure 8. Core sizes range from 20 to 120 nm, indicating diverse morphologies, while hydrodynamic sizes peak below

500 nm, suggesting the presence of larger aggregates. Surface charges are predominantly negative, centered around -20 mV, critical for membrane interactions. Surface areas fall below 50 m<sup>2</sup>/g, and Energy Concentrations (Ec) hover around -4, indicating nanoparticle stability. Exposure durations are primarily around 20 hours, and dosage values are typically under 50 µg/mL, with some exceeding 200 µg/mL, reflecting varying in vitro exposure conditions.

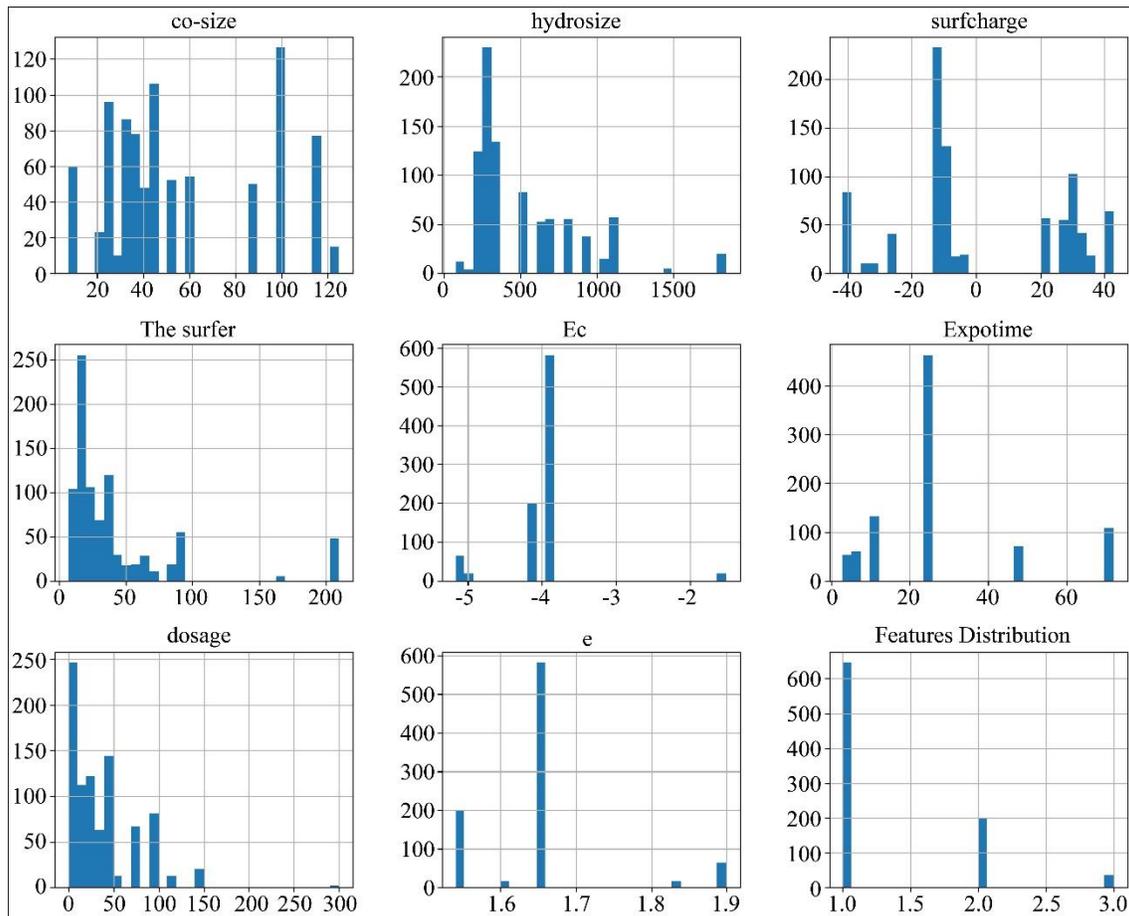


Fig. 8 Feature distribution

The correlation analysis highlights key physicochemical parameters influencing NP toxicity, as illustrated in Figure 9. Surface size shows a strong positive correlation, likely due to increased reactive surface activity. Hydrodynamic size has a moderate positive correlation, indicating that larger agglomerates may impact toxicity. In contrast, surface charge exhibits a negative correlation, with highly charged, especially negatively charged, nanoparticles being less toxic, possibly due to electrostatic repulsion. The exposure period has a slight negative correlation, suggesting that longer exposure does not necessarily increase toxicity.

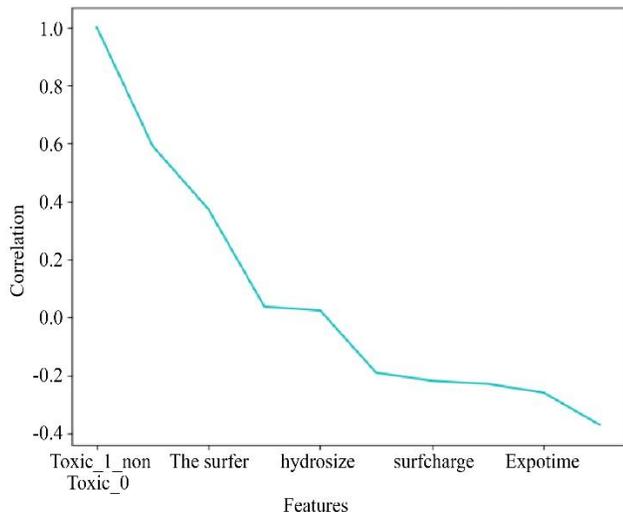


Fig. 9 Correlation of features with toxicity

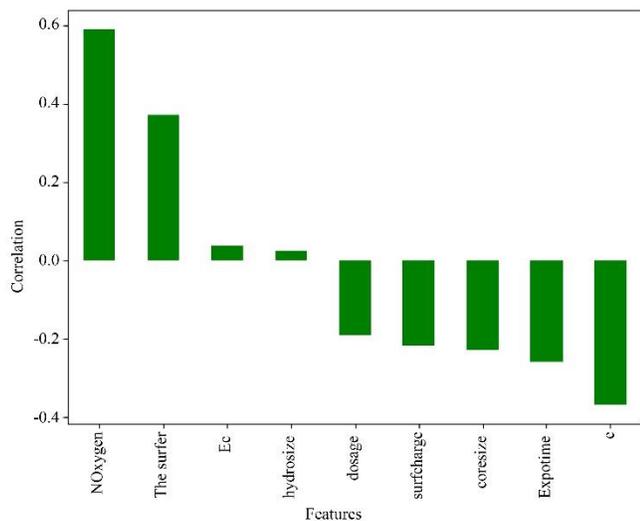


Fig. 10 Correlation with nanoparticle type

The correlation study of characteristics with nanoparticle type reveals significant correlations essential for comprehending nanoparticle behavior and toxicity, as shown in Figure 10. The presence of NOxygen and surface area show strong positive correlations, indicating that larger surface areas may enhance nanoparticle reactivity and affect toxicity.

Conversely, hydrosize, dose, surface charge, core size, and exposure time demonstrate negative correlations, suggesting that increases in these attributes reduce the likelihood of certain nanoparticle types being categorized into specific groups. These findings emphasize the complex interplay of physical and chemical properties in determining nanoparticle behavior and toxicity.

Figure 11 presents a heatmap that depicts the pairwise correlations among diverse physicochemical parameters of nanoparticles and their corresponding toxicity, elucidating the impact of each characteristic on the toxicological profile.

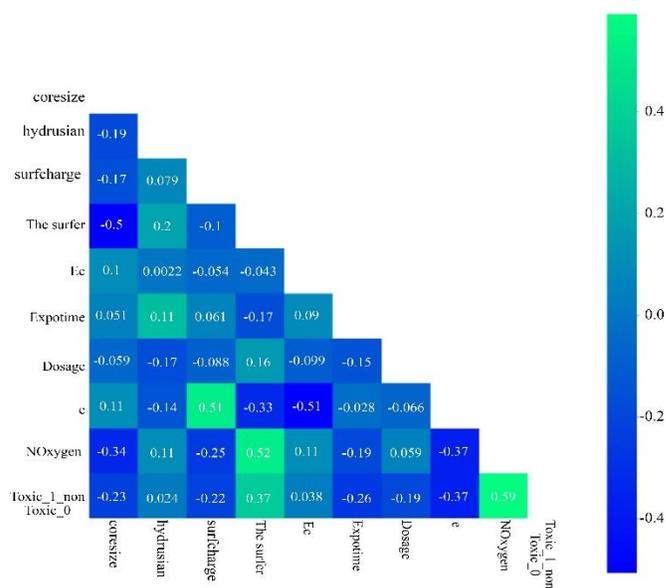


Fig. 11 Heatmap of the data

### 3.3. Preprocessing

Data preprocessing is an essential phase in the preparation of raw data for learning models, assuring that the data is cleaned, consistent and prepared for analysis. The dataset consists of 881 items and encompasses 11 columns, as shown in Figure 12, with no missing or null values. The absence of null values simplifies the preprocessing stage, as there is no need to handle missing data. Due to this completeness, data cleaning is unnecessary, facilitating a more efficient approach to further modeling and analysis.

A StandardScaler is utilized on the feature set to normalize the data, rescaling it to possess a mean of 0 and a standard deviation of 1. This step is crucial for reducing bias resulting from varying feature scales and enhancing the efficacy of model training. The data is reshaped into a format suitable for 1D CNNs by incorporating an additional dimension facilitating effective convolutional operations by the network. The dataset is finally divided into 80% allocated for training and 20% for testing sets. This implies that the model gets trained on the majority of the data while being assessed on an individual unseen segment.

```
[ ] df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 881 entries, 0 to 880
Data columns (total 11 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   NPs          881 non-null    object
1   coresize    881 non-null    float64
2   hydrosized  881 non-null    float64
3   surfcharge  881 non-null    float64
4   surfarea    881 non-null    float64
5   Ec          881 non-null    float64
6   Exptime     881 non-null    int64
7   dosage      881 non-null    float64
8   e           881 non-null    float64
9   NOxygen     881 non-null    int64
10  class       881 non-null    object
dtypes: float64(7), int64(2), object(2)
memory usage: 75.8+ KB
```

Fig. 12 Null value check

### 3.4. Model Development

This phase emphasizes the creation of a resilient architecture that combines a 1D CNN with transformer mechanisms. The 1D CNN is very adept at assessing the

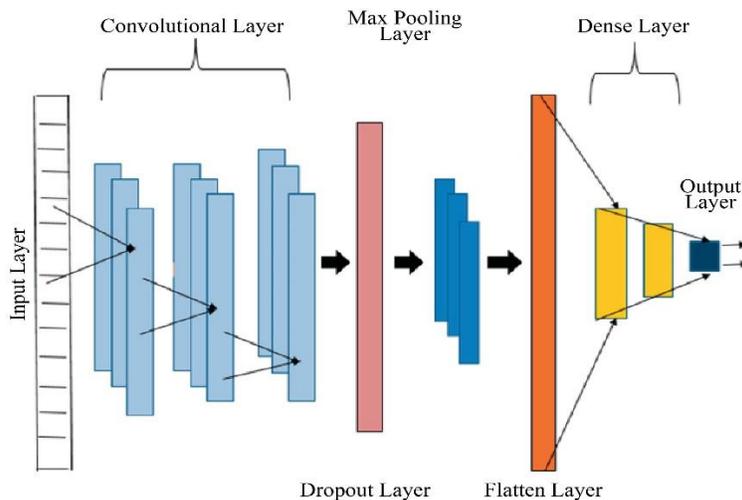


Fig. 13 Architecture of 1-D CNN

The architecture generally comprises multiple vital layers, illustrated in Figure 13, which enhance the network's capacity to discern intricate patterns and correlations within the input data [19]. The initial layer of the architecture is the input layer, which is set up to take sequential data as input. In a 1-D CNN, the input form is typically denoted as in Equation (1).

$$X = [x_1, x_2, \dots, x_T] \quad (1)$$

Where  $T$  is the sequence length,  $x_T$  is the value at time step  $T$ .

The initial processing component subsequent to the input layer is the 1-D Convolutional Layer. This layer performs convolutional operations to extract localized characteristics

sequential characteristics of physicochemical feature data, facilitating efficient feature extraction and the identification of complex patterns associated with NP behavior.

The integration of transformers augments the model's capability to concentrate on pertinent information via self-attention mechanisms, hence enhancing its ability to identify minor correlations among features that may affect toxicity consequences.

#### 3.4.1. 1-D Convolutional Neural Network

Conventional CNNs are primarily designed for 2D data, such as images utilizing 2D convolutional layers to extract spatial hierarchies and features across height and width. Conversely, 1D CNNs are tailored for 1D data, rendering them especially appropriate for tasks related to sequential or time-series data. The 1D CNN concentrates on examining the sequence of features within a singular dimension, enabling it to proficiently discern patterns and relationships essential for predicting NP toxicity.

from the input data. Mathematically, the convolution operation's output is represented by Equation (2).

$$Y[i] = \sum_{j=0}^{k-1} X[i+j] \cdot W[j] + b \quad (2)$$

Where,  $Y[i]$  signifies the output at position  $i$ ,  $X$  refers to the input sequence,  $W$  represents the filter weights,  $b$  denotes the bias, and  $k$  indicates the kernel size. The convolutional layer utilizes an activation function, commonly the Rectified Linear Unit (ReLU), as specified in Equation (3).

$$f(x) = \max(0, x) \quad (3)$$

This activation incorporates non-linearity into the model, enabling it to learn more intricate mappings from input to output. A max-pooling layer is added following the

convolutional layer, which selects the maximum value from each region of the feature map, effectively lowering the dimensionality of the feature maps while preserving the most essential information. Figure 14 shows the basics of the max pooling operation. The pooling operation is mathematically represented, as shown in Equation (4).

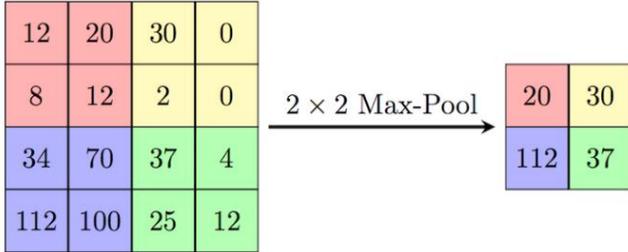


Fig. 14 Basics of max pooling

$$Y[i] = \max(X[i:i + p]) \tag{4}$$

Where  $p$  is the pool size. This method facilitates down-sampling the output, enhancing model efficiency and mitigating the risk of overfitting.

Integrating dropout layers subsequent to the pooling layers is a prominent strategy to alleviate overfitting during training. A dropout layer randomly nullifies a portion of the input units during training, thereby mitigating the model's dependence on particular features. The output from the last convolutional or pooling layer is flattened and input into one or more fully connected (dense) layers. The dense layer operates according to Equation (5).

$$z_j = \sum_{i=1}^N v_{ij} \cdot y_i + c_j \tag{5}$$

The final dense layer usually employs a softmax function in classification tasks, generating a probability distribution among the potential classes. The softmax function is delineated in Equation (6).

$$p_j = \frac{\exp(z_j)}{\sum_{k=1}^C \exp(z_k)} \tag{6}$$

The combination of these layers forms the core of the 1D CNN architecture, enabling the model to capture spatial dependencies and assimilate relevant information from the input data for accurate predictions.

### 3.4.2. Transformer Block

The Transformer architecture is a Deep Learning model designed for sequential data, notably distinct from conventional approaches like Long Short-Term Memory (LSTM) networks and Recurrent Neural Networks (RNNs). The transformer design, introduced by Vaswani et al. (2017) [20], relies on the principle of self-attention, allowing for the

effective modeling of long-range dependencies in input data. Its architecture facilitates parallel processing, rendering it more scalable and quicker than previous sequence-based models. The initial phase of the transformer architecture involves converting input tokens (words or sequential data) into dense vector representations through an embedding layer. For a series of tokens  $X = (x_1, x_2, \dots, x_n)$ , each token  $x_i$  corresponds to an embedding vector of dimension  $d$ , producing a matrix  $E \in \mathbb{R}^{n \times d}$ , where  $n$  denotes the sequence length, and  $d$  signifies the embedding dimension.

Furthermore, due to the transformer's lack of an inherent understanding of sequential order, in contrast to RNNs, positional encodings are incorporated into the embeddings to provide the positional information of all tokens within the sequence. The positional encoding  $PE$  is calculated for each position  $pos$  and dimension  $i$ , shown in Equation (7).

$$PE(pos, 2i) = \sin\left(\frac{pos}{10000^{2i/d}}\right)$$

$$PE(pos, 2i + 1) = \cos\left(\frac{pos}{10000^{2i/d}}\right) \tag{7}$$

The fundamental component of the transformer design is the self-attention mechanism, enabling each token in the sequence to concentrate on every other token, thereby capturing relationships between them, regardless of their positional distance in the sequence. The self-attention technique employs three learnt matrices: *Query* ( $Q$ ), *Key* ( $K$ ), and *Value* ( $V$ ), derived from linear transformations of the input embeddings as specified below.

$$Q = XW^Q; K = XW^K; V = XW^V \tag{8}$$

Where  $W^Q$ ,  $W^K$ , and  $W^V$  are learned weight matrices. The attention score for each token pair is calculated as the dot product of their query and key vectors, scaled by the square root of the embedding dimension  $d_k$ , and subsequently processed through a softmax function to derive the attention weights, as illustrated in Equation (9). This provides a weighted sum of the value vectors, with tokens that are more pertinent to a certain token receiving more weights.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{9}$$

The transformer employs Multi Head Attention (MHA) to enable the model to concurrently concentrate on various segments of the sequence, utilizing multiple attention mechanisms (or "heads") in parallel, as shown in Figure 15. Each attention head acquires distinct relationships among the sequence segments, capturing diverse aspects of the data. The outputs from the various heads are concatenated and subjected to a linear transformation.

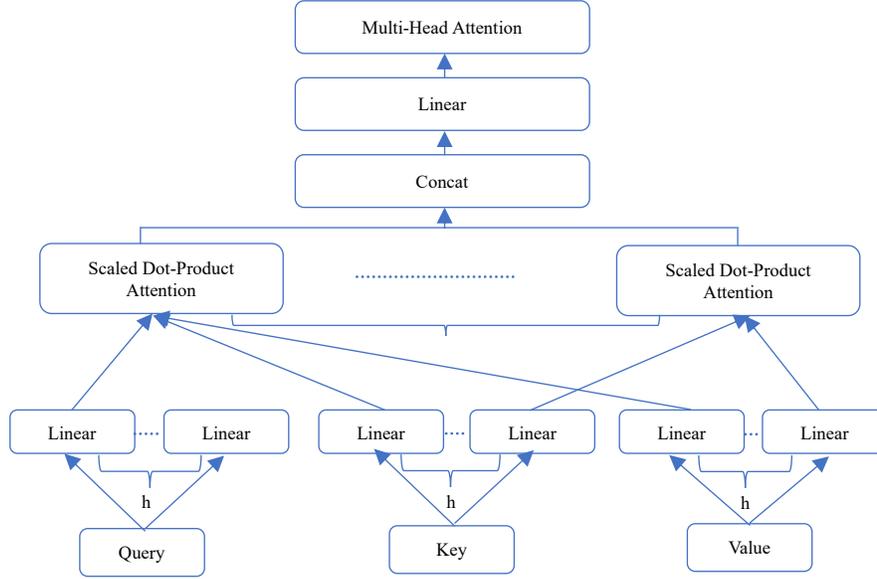


Fig.15. Structure of multi head attention layer

Mathematically, for each head  $i$ :

$$Head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (10)$$

Where  $W_i^Q$ ,  $W_i^K$ , and  $W_i^V$  are the weight matrices for the  $i^{th}$  attention head.

The outputs are subsequently projected through a final dense layer to integrate the information from each head. The output of MHA is computed as described in Equation (11).

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{Head}_1, \text{Head}_2, \dots, \text{Head}_h)W^o \quad (11)$$

Where  $\text{Concat}(\cdot)$  denotes the concatenation of outputs from all attention heads,  $h$  is the number of heads,  $W^o$  is the weight matrix for the final linear transformation, projecting the concatenated output into the desired dimension.

Subsequently, each point in the sequence undergoes processing via a Feed Forward Neural network (FFN) comprising two linear transformations separated with a ReLU activation function. The FFN is mathematically outlined as presented in Equation (12).

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (12)$$

This framework allows the model to incorporate non-linearity and enhance the sequence representations following attention. Subsequent to the FFN, each sub-layer, encompassing both self-attention and feed-forward layers, is enhanced by layer normalization to improve training stability. Furthermore, residual connections are utilized to alleviate vanishing gradient problems in deeper networks. The output

for each sub-layer is denoted as shown in Equation (13). This ensures that the model can learn more efficiently as it progresses.

$$\text{LayerNorm}(x + \text{sublayer}(x)) \quad (13)$$

The Transformer model comprises two primary components: the encoder and the decoder. The encoder has several identical layers, each featuring two essential components: a multi-head self-attention mechanism and a feed-forward neural network. This framework allows the encoder to analyze the input sequence and generate a series of encoded representations that are then transmitted to the decoder. Conversely, the decoder is rather more intricate, consisting of several identical layers while integrating three sub-layers. This encompasses masked multi-head self-attention, which inhibits the model from accessing future positions in the sequence during training, an MHA layer that focuses on the encoder's output, and a feed-forward network. This architecture enables the decoder to construct output predictions incrementally, leveraging both the encoded data from the encoder and the previously generated outputs. The final output from the Transformer decoder is passed through a softmax layer to produce probability distributions over the output classes.

### 3.4.3. Proposed 1-D CNN with Transformers

The proposed design combines a 1-D CNN with a Transformer block to utilize the benefits of both models. The input layer takes data formatted as (num\_features, 1), which is processed by the initial 1-D convolutional layer that employs 64 filters with a kernel size of 3 and leverages the ReLU activation function to identify local patterns in the data. A max pooling layer with a pool size of 2 subsequently decreases dimensionality and highlights the most salient features. A

dropout layer with a rate of 0.3 is subsequently employed to reduce overfitting. The architecture proceeds with a subsequent 1-D CNN that incorporates 128 filters, maintaining the same kernel size and activation function. Max pooling and dropout layers are utilized to enhance the feature extraction process prior to the data entering the Transformer

block. The model utilizes MHA, enabling it to concentrate on multiple representation subspaces concurrently. This capacity is crucial for comprehending intricate connections in nanoparticle toxicity. The Transformer block consists of several essential components, as illustrated in Figure 16.

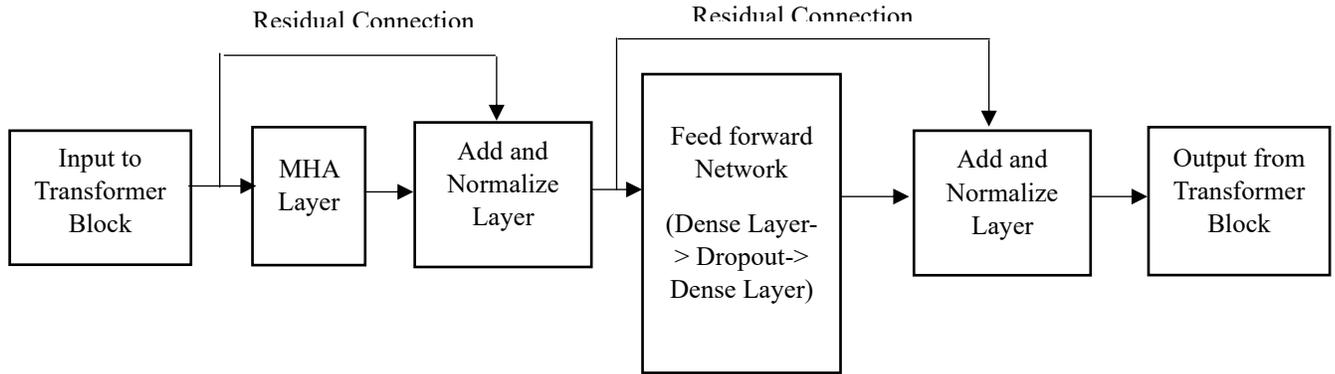


Fig. 16 Structure of transformer block

The multi-head attention layer's output and the initial input are connected via a residual link that is created by the add and normalize layer. This connection is essential for maintaining information across layers and enabling stable training via layer normalization. Subsequently, a feed-forward network employs an extensive layer with a ReLU activation, followed by an additional dropout layer to mitigate overfitting. A further dense layer devoid of activation modifies the output data prior to its incorporation into the residual link, succeeded by an additional normalization phase.

The final normalized output from the transformer block is efficiently transmitted to the successive layers in the neural network, ensuring an uninterrupted flow of processed information. This architecture allows the model to successfully capture both local and global contextual variables, rendering it highly appropriate for the intricate task of nanoparticle toxicity identification. The suggested approach improves its capacity to identify intricate patterns by merging CNNs and Transformers, resulting in more precise predictions of nanoparticle toxicity. Figure 17 illustrates the architecture of the suggested framework.

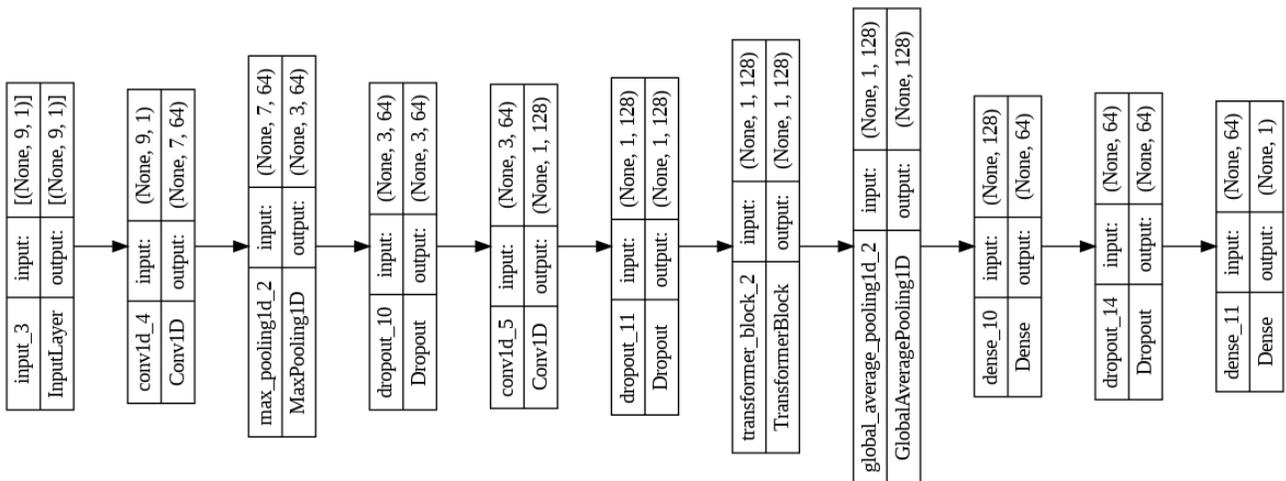


Fig. 17 Model architecture of the proposed model

### 3.5. Hardware and Software Setup

The study utilized a high-performance computational configuration comprising an Intel Core i7 CPU, 32GB of RAM, and an NVIDIA GeForce GTX 1080Ti GPU,

facilitating the effective execution of complex computational tasks. The modeling framework was constructed utilizing the Keras library, a high-level neural network API based on TensorFlow, recognized for its intuitive interface and robust

functionalities. The training was conducted using Google Colab, a cloud-based Python notebook platform that provides easy access to substantial computational resources, thus facilitating the model training process.

A crucial element of this research was the precise selection of hyperparameters, which significantly influence the model's performance during training. In contrast to model parameters derived from data, hyperparameters are established by the user and are crucial for optimizing the training process of the nanoparticle toxicity detection model. The precise selections for hyperparameters and model setups are thoroughly detailed in Table 1.

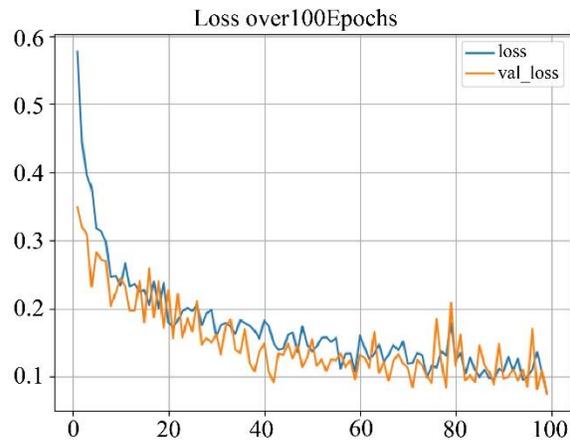
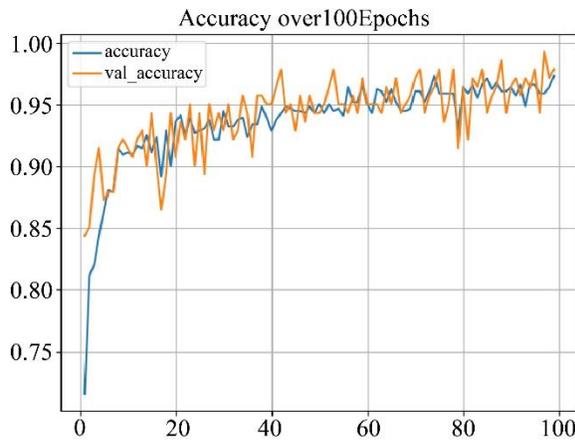
**Table.1. Hayperparameter specifications**

Hyperparameters	Values	
Epochs	100	
Optimizer	ADAM	
Learning Rate	0.0001	
Loss function	Binary cross entropy	
Dropout	CNN	0.3
	Transformer	0.5

#### 4. Results and Discussion

The accuracy plot illustrates the efficacy of the model in accurately predicting outcomes across epochs, while the loss plot reflects the model's predictive error, with both ideally tending towards increased accuracy and decreased loss as training advances. At first epoch, the model exhibited a training accuracy of 59.5% and a validation accuracy of 78.0%. The training accuracy increased steadily to about 93% by epoch 30, while the validation accuracy fluctuated around the same amount as the training progressed. The model achieved a training accuracy of roughly 96.3% and a validation accuracy of 96.8% throughout numerous epochs, as illustrated in Figure 18, demonstrating its adeptness in learning and generalizing nanoparticle toxicity predictions.

Initially, the loss is elevated at 0.6 at epoch 0, but it decreases rapidly, falling below 0.3 by epoch 10. The validation loss displays a comparable trend, commencing at a high value and thereafter decreasing to 0.3 by the identical epoch. Following epoch 20, both training and validation losses exhibit a continuous decline despite intermittent increases in the validation loss; however, an overarching decreasing trend remains evident. By epoch 100, both training and validation losses have approximated 0.1 or below, signifying that the model has successfully minimized loss and exhibits robust learning behavior.



**Fig. 18 Accuracy and loss plot of the proposed study**

As the loss diminishes, the model's accuracy markedly enhances, as illustrated in Figure 19, commencing with a lower accuracy of approximately 0.60 when the loss is around 0.7. As the loss decreases to roughly 0.4, accuracy rises over 0.80, indicating a considerable improvement in performance. When the loss decreases below 0.2, accuracy stabilizes at approximately 0.95, signifying the model's convergence to elevated predictive performance with negligible error. This steady behavior demonstrates that the model proficiently learns from the data, attaining ideal accuracy near 95% as the loss approaches its minimum.

Figure 20 illustrates a distinct negative correlation between validation loss and accuracy. Initially, with an accuracy of roughly 0.60, the validation loss is elevated at about 0.45. As the model's accuracy surpasses 0.70, the validation loss declines significantly, demonstrating the model's ability to generalize while integrating input. When accuracy exceeds 0.90, the validation loss stabilizes between 0.10 and 0.15, signifying a robust equilibrium between fitting the training data and achieving remarkable performance on unknown validation data. This pattern demonstrates the model's strong generalization potential, as the validation loss persistently decreases while accuracy nears 0.95.

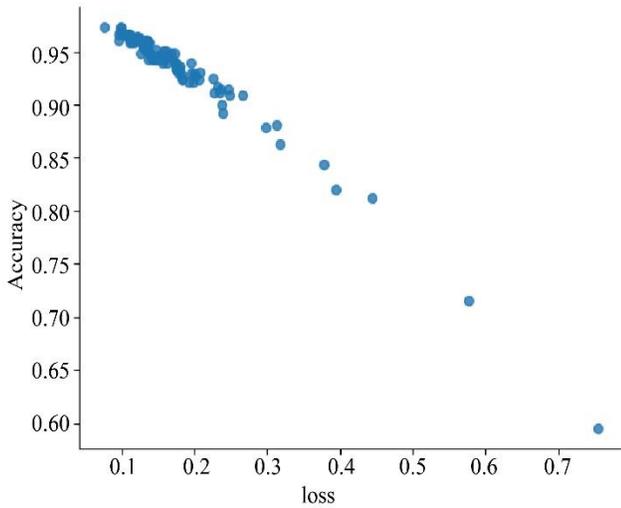


Fig. 19 Loss vs Accuracy

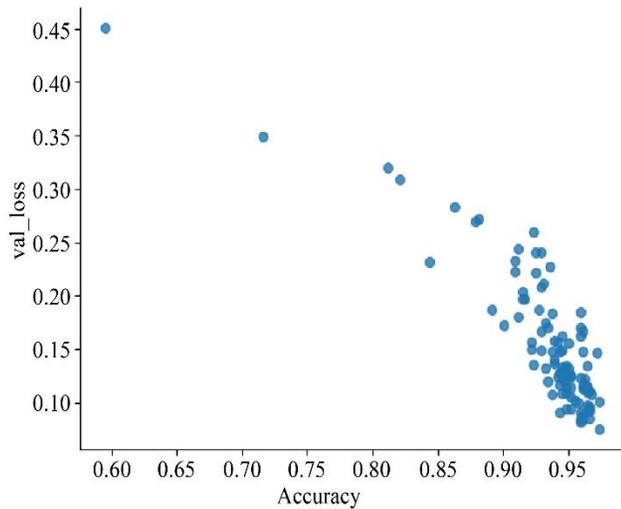


Fig. 20 Accuracy Vs validation loss

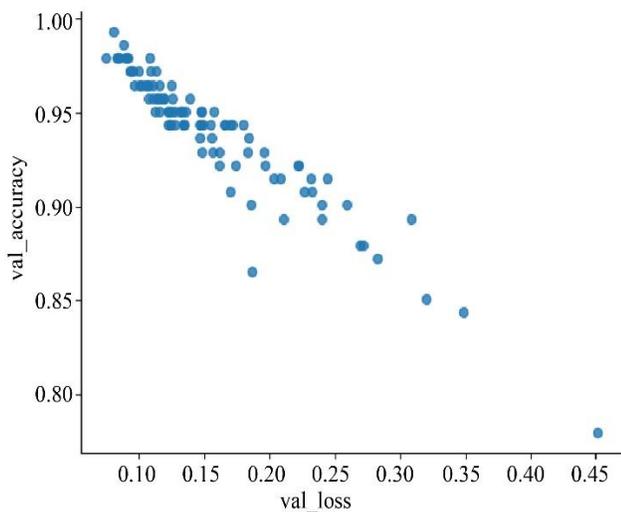


Fig. 21 Validation loss vs Validation accuracy

Figure 21 demonstrates a distinct inverse correlation between validation loss and validation accuracy in the proposed study. When the validation loss is minimal, between 0.1 and 0.2, the validation accuracy attains its highest point, ranging from 0.95 to 1.0, indicating excellent model performance throughout training. As validation loss surpasses 0.3, accuracy diminishes below 0.9 and continues to decrease. The outlier, with a loss of approximately 0.45, indicates a substantial decline in the model's predictive efficacy at that point.

To comprehensively assess the efficacy and operational efficiency of the suggested model, many aspects have been outlined to quantify critical performance metrics, as illustrated in the subsequent equations. These metrics, grounded in the concepts of False Positive (FP), True Negative (TN), False Negative (FN), and True Positive (TP), are essential for assessing the model's efficacy. The mathematical formulations for these performance parameters are shown in Equations (14), (15), (16), and (17).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

$$F1 - score = 2 \times \frac{precision \times Recall}{Precision + Recall} \tag{17}$$

The performance assessment of the suggested design exhibits exceptionally effective categorization results, as depicted in Figure 22. The model attained a remarkable accuracy of 96.61%, demonstrating its capability to reliably predict the toxicity of nanoparticles.

The precision, evaluated at 100%, signifies that every case identified as toxic was indeed toxic, demonstrating the model's proficiency in reducing false positives. The recall value of 92.85% underscores the model's robust sensitivity, indicating its efficacy in identifying a significant proportion of actual toxic occurrences. The model demonstrates a balanced performance between precision and recall, evidenced by an F1-score of 96.29%, indicating its robustness in managing imbalanced toxicity data.

Figure 23 depicts the confusion matrix, which functions as a tool for assessing the classification efficiency of the proposed framework. The model accurately detected 93 toxic samples and 78 non-toxic samples, with no FP, demonstrating remarkable precision. The occurrence of six false negatives indicates that certain toxic samples were inaccurately labeled as non-toxic.

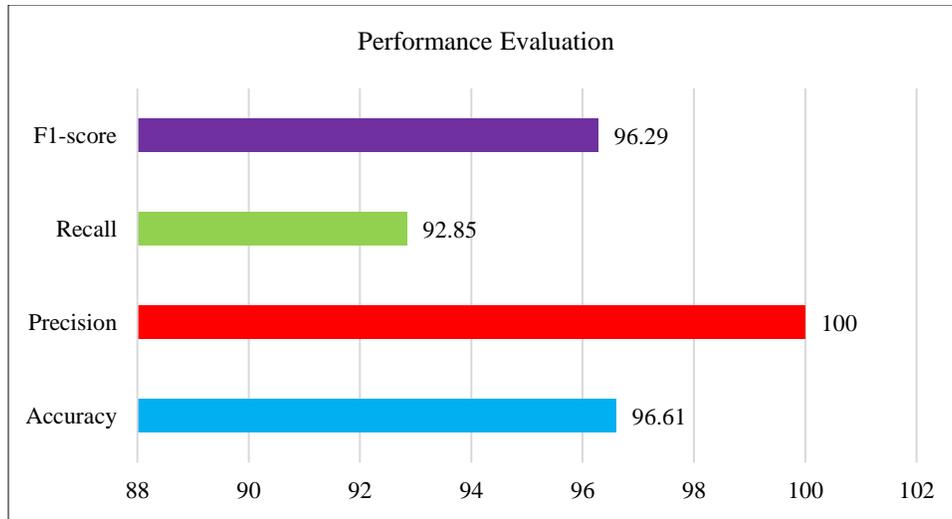


Fig. 22 Performance evaluation of the proposed study

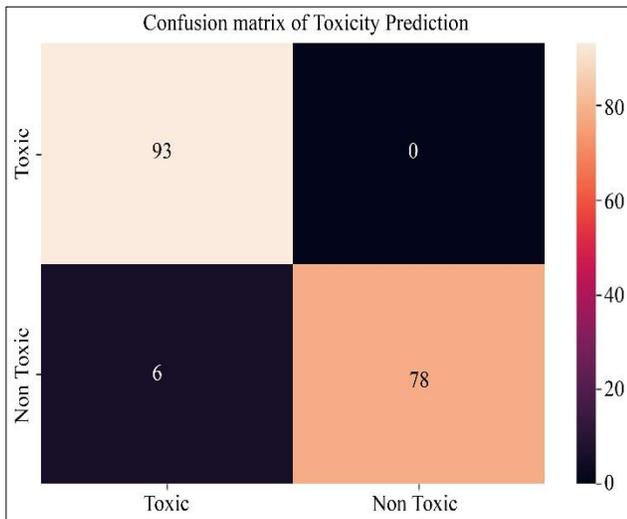


Fig. 23 Confusion matrix

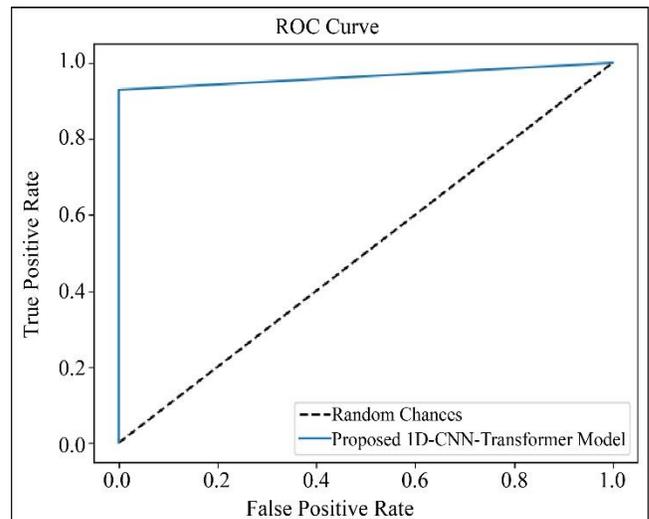


Fig. 24 ROC curve

Table 2. Performance comparison of the proposed model with existing methods

Author	Methodology	Accuracy (%)
Marvin et al. [21]	Bayesian Network	72
Nilesh Anantha Subramanian and Ashok Palaniappan [14]	LGR, RF, SVM	96 (RF)
Yang Huang [13]	C4.5 decision tree (C4.5), LGR, RF, kNN, DT, locally weighted learning (LWL), Bayesnet, SVM	82 (SVM)
<b>Proposed model: 1-D CNN with transformers</b>		<b>96.61</b>

The Receiver Operating Characteristic (ROC) curve of the model demonstrates exceptional efficacy in differentiating between toxic and non-toxic nanoparticles, as shown in Figure 24. The curve's steep rise and closeness to the y-axis signify a robust equilibrium between sensitivity and specificity, along with negligible false positives. This indicates that the model

substantially exceeds random chance, evidenced by a high Area Under the Curve (AUC), hence reinforcing its efficacy in properly detecting toxic nanoparticles while reducing incorrect classifications. The performance comparison of the suggested model to existing models for nanoparticle toxicity prediction demonstrates its enhanced accuracy, as illustrated in Figure 25

and Table 2. The proposed model attains an accuracy of 96.61%, marginally surpassing the RF model's accuracy of 96%. This indicates that although RF is a powerful option, the incorporation of transformers with 1-D CNN yields a slight enhancement, signifying superior feature extraction and

decision-making abilities. The SVM and Bayesian Network models exhibit markedly inferior performance, with accuracies of 82% and 72%, respectively, so underscoring the efficacy of deep learning models for this task.

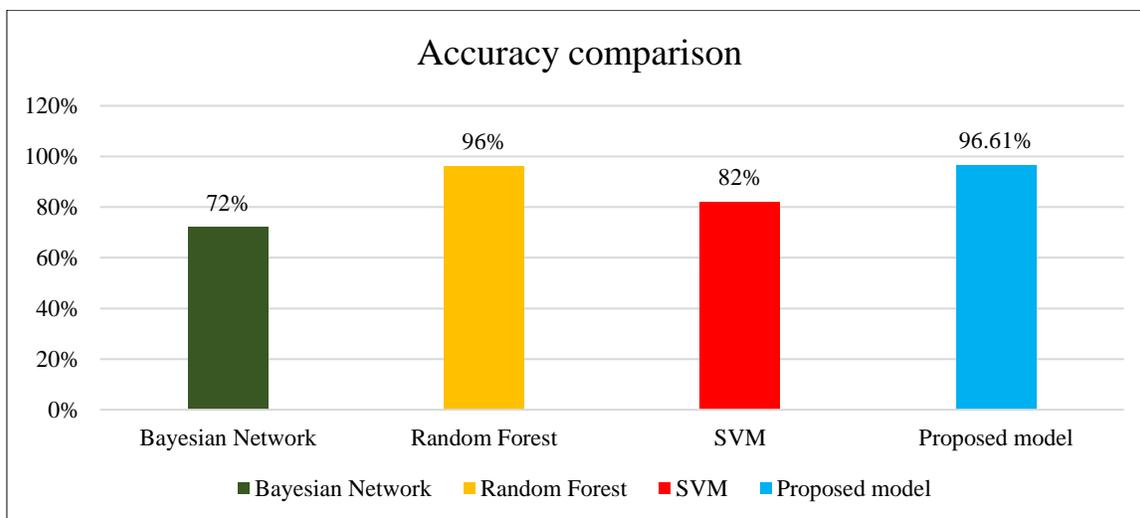


Fig. 25 Performance comparison of the proposed model with existing methods

## 5. Conclusion

The proposed research effectively illustrates the efficacy of a 1-D CNN combined with transformers in forecasting nanoparticle toxicity, with an excellent accuracy of 96.61%. This strategy improves the reliability of toxicity predictions and mitigates the drawbacks of existing techniques, including limited sample sizes and insufficient sensitivity. The model's remarkable precision and recall metrics underscore its capability to reduce false positives and accurately detect toxic nanoparticles. As nanotechnology advances, the necessity for rigorous toxicity evaluation approaches becomes paramount to

guarantee the safe production and utilization of nanomaterials. Subsequent research may build upon this study by investigating the incorporation of additional data sources, including omics data and environmental variables, to enhance predictive accuracy.

## Acknowledgements

The author expresses profound appreciation to the supervisor for providing guidance and unwavering support throughout the course of this study.

## References

- [1] Muhammad Rafique et al., *Chapter 1 - History and Fundamentals of Nanoscience and Nanotechnology*, Nanotechnology and Photocatalysis for Environmental Applications: Micro and Nano Technologies, Elsevier, pp. 1-25, 2020. [CrossRef] [Google Scholar] [Publisher Link]
- [2] Sharon Mitchell et al., "Nanoscale Engineering of Catalytic Materials for Sustainable Technologies," *Nature Nanotechnology*, vol. 16, no. 2, pp. 129-139, 2021. [CrossRef] [Google Scholar] [Publisher Link]
- [3] Safia Hameed et al., "Cannabis Sativa-Mediated Synthesis of Gold Nanoparticles and Their Biomedical Properties," *Bioinspired, Biomimetic and Nanobiomaterials*, vol. 9, no. 2, pp. 95-102, 2020. [CrossRef] [Google Scholar] [Publisher Link]
- [4] Ibrahim Khan, Khalid Saeed, and Idrees Khan, "Nanoparticles: Properties, Applications and Toxicities," *Arabian Journal of Chemistry*, vol. 12, no. 7, pp. 908-931, 2019. [CrossRef] [Google Scholar] [Publisher Link]
- [5] C.M. Vineeth Kumar et al., "The Impact of Engineered Nanomaterials on the Environment: Release Mechanism, Toxicity, Transformation, and Remediation," *Environmental Research*, vol. 212, no. 2, 2022. [CrossRef] [Google Scholar] [Publisher Link]
- [6] Murthy S. Chavali, and Maria P. Nikolova, "Metal Oxide Nanoparticles and their Applications in Nanotechnology," *SN Applied Sciences*, vol. 1, no. 6, pp. 1-30, 2019. [CrossRef] [Google Scholar] [Publisher Link]
- [7] Zhongjie Yu et al., "Reactive Oxygen Species-Related Nanoparticle Toxicity in the Biomedical Field," *Nanoscale Research Letters*, vol. 15, no. 1, pp. 1-14, 2020. [CrossRef] [Google Scholar] [Publisher Link]
- [8] Furkan Eker et al., "A Comprehensive Review of Nanoparticles: from Classification to Application and Toxicity," *Molecules*, vol. 29, no. 15, pp. 1-83, 2024. [CrossRef] [Google Scholar] [Publisher Link]

- [9] Stefania Gottardo et al., "Towards Safe and Sustainable Innovation in Nanotechnology: State-of-play for Smart Nanomaterials," *NanoImpact*, vol. 21, pp. 1-10, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Ajay Vikram Singh et al., "Artificial Intelligence and Machine Learning in Computational Nanotoxicology: Unlocking and Empowering Nanomedicine," *Advanced Healthcare Materials*, vol. 9, no. 17, pp. 1-19, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Iqra Yousaf, "AI and Machine Learning Approaches for Predicting Nanoparticles Toxicity the Critical Role of Physicochemical Properties," *arXiv*, pp. 1-20, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [12] Yunchi Zhou et al., "Using Machine Learning to Predict Adverse Effects of Metallic Nanomaterials to Various Aquatic Organisms," *Environmental Science & Technology*, vol. 57, no. 46, pp. 17786-17795, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] Yang Huang et al., "Use of Dissociation Degree in Lysosomes to Predict Metal Oxide Nanoparticle Toxicity in Immune Cells: Machine Learning Boosts Nano-Safety Assessment," *Environment International*, vol. 164, pp. 1-12, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [14] Nilesh Anantha Subramanian, and Ashok Palaniappan, "NanoTox: Development of a Parsimonious in Silico Model for Toxicity Assessment of Metal-Oxide Nanoparticles Using Physicochemical Features," *ACS omega*, vol. 6, no. 17, pp. 11729-11739, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Farooq Ahmad, Asif Mahmood, and Tahir Muhmood, "Machine Learning-integrated Omics for the Risk and Safety Assessment of Nanomaterials," *Biomaterials science*, vol. 9, no. 5, pp. 1598-1608, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] Hengjie Yu, Zhilin Zhao, and Fang Cheng, "Predicting and Investigating Cytotoxicity of Nanoparticles by Translucent Machine Learning," *Chemosphere*, vol. 276, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [17] Jossana A. Damasco et al., "Understanding Nanoparticle Toxicity to Direct a Safe-by-Design Approach in Cancer Nanomedicine," *Nanomaterials*, vol. 10, no. 11, pp. 1-41, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Bhavna Saini, Sumit Srivastava, and A.K. Bajpai, "Deep CNN Model for Nanotoxicity Classification Using Microscopic Images," *International Journal of Advances in Soft Computing and its Applications*, vol. 12, no. 2, pp. 1-22, 2020. [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Furui Wang, Gangbing Song, and Yi-Lung Mo, "Shear Loading Detection of Through Bolts in Bridge Structures Using a Percussion-Based One-Dimensional Memory-Augmented Convolutional Neural Network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 3, pp. 289-301, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Ashish Vaswani, "Attention is all You Need," *31<sup>st</sup> Conference on Neural Information Processing Systems*, Long Beach, CA, USA, pp. 1-11, 2017. [[Google Scholar](#)] [[Publisher Link](#)]
- [21] Hans J.P. Marvin et al., "Application of Bayesian Networks for Hazard Ranking of Nanomaterials to Support Human Health Risk Assessment," *Nanotoxicology*, vol. 11, no. 1, pp. 123-133, 2017. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]