Task Description

**Task Title :** Predicting Glass Transition Temperature of Polymers Using Machine Learning, Deep Learning, and Physics-Informed Neural Networks.

We will develop an AI model to predict the glass transition temperature (Tg) of polymers directly from their SMILES representations. By extracting meaningful molecular descriptors such as molecular weight, flexibility, and polarity, the model will learn patterns that influence Tg.

**What is Tg?**

**Tg (Glass Transition Temperature)** is the temperature at which a polymer changes from a hard, glassy state to a soft, rubbery state. It is a critical thermal property that determines a polymer’s flexibility, durability, and usability under various conditions.

* Higher Tg → The polymer stays rigid and strong at higher temperatures (good for structural or heat-resistant materials).
* Lower Tg → The polymer becomes flexible and soft at lower temperatures (good for packaging, films, or soft plastics).

**Why Predict Tg?**

**Material Design:** Helps in designing polymers for specific applications (e.g., packaging, electronics, aerospace).

**Cost-Efficient**: Reduces the need for expensive and time-consuming lab experiments.

**Performance:** Ensures the material will perform safely and effectively in real-world environments.

Dataset Overview

**Dataset Overview**

The dataset is collected from the [NeurIPS 2025 Open Polymer Prediction Challenge on Kaggle](https://www.kaggle.com/competitions/neurips-open-polymer-prediction-2025/data).

It contains **7973** polymer samples.

Each represented by a SMILES string and associated with various physical and thermal properties.

Features in dataset: SMILES, FFV, Tc, Density, Rg, Id, Tg

**Target feature: Tg, Only 511 data points. It is very difficult to use Deep Learning models.**

### **SMILES** (7973 samples available): String notation representing the molecular structure of the polymer. All key structural features (size, flexibility, polarity) that influence **Tg** can be extracted from SMILES.

### **Tg-** Glass transition temperature (511 samples available) : The point where a polymer transitions from a rigid to a rubbery state. It is influenced by molecular weight, polarity, flexibility, and packing.

**FFV**- Free Volume Fraction (7030 samples available) : Fraction of unoccupied space in the polymer structure.

* Higher FFV usually lowers Tg, as more free space allows easier molecular motion.

**Tc** - Crystallization Temperature (737 samples available): Temperature at which a polymer crystallizes upon cooling.

* Positively correlated with Tg; polymers with higher Tc usually have more ordered structures and potentially higher Tg.

**Density** (613 samples available): Mass per unit volume of the polymer.

Higher density often indicates tighter packing and less mobility, which may increase Tg.

**Rg**- Radius of Gyration (614 samples available): Describes the spread of the polymer chain in space (chain size/shape).

Influence: Larger Rg may indicate more flexible or extended structures, which can lead to lower Tg depending on polymer architecture.

**Id** (7973 samples available): Identifier for each polymer entry (row index).

Influence: None. Used only for indexing — should be ignored during model training.

# **Tg-Related Features for SMILES-Based Prediction (Extracted features from SMILES)**

## **MolWt**

Description: Molecular weight of the molecule.

Influence: Higher molecular weight leads to higher Tg due to reduced chain mobility.

How to Extract: Use RDKit: Descriptors.MolWt(mol)

## **LogP**

Description: Partition coefficient between octanol and water (hydrophobicity).

Influence: Lower LogP (more polar) often leads to higher Tg due to stronger intermolecular forces.

How to Extract: Use RDKit: Descriptors.MolLogP(mol)

## **TPSA**

Description: Topological polar surface area.

Influence: Higher TPSA indicates more polar regions, increasing Tg through stronger interactions.

How to Extract: Use RDKit: Descriptors.TPSA(mol)

## **NumHDonors**

Description: Number of hydrogen bond donors.

Influence: More hydrogen bonding increases Tg due to stronger intermolecular forces.

How to Extract: Use RDKit: Descriptors.NumHDonors(mol)

## **NumHAcceptors**

Description: Number of hydrogen bond acceptors.

Influence: More hydrogen bonding increases Tg due to stronger intermolecular forces.

How to Extract: Use RDKit: Descriptors.NumHAcceptors(mol)

## **NumRotatableBonds**

Description: Number of rotatable bonds (chain flexibility).

Influence: More rotatable bonds reduce Tg due to increased flexibility.

How to Extract: Use RDKit: Descriptors.NumRotatableBonds(mol)

## **FractionCSP3**

Description: Fraction of sp3-hybridized carbon atoms.

Influence: Higher sp3 fraction means more flexibility, reducing Tg.

How to Extract: Use RDKit: Descriptors.FractionCSP3(mol)

## **RingCount**

Description: Total number of rings in the molecule.

Influence: More rings increase rigidity and usually raise Tg.

How to Extract: Use RDKit: Descriptors.RingCount(mol)

## 

## 

Data Preprocessing

**Data Preprocessing**

**Target Filtering:** Removed rows where the target value Tg was missing, resulting in 511 valid samples out of 7973.

**Feature Extraction from SMILES:**

* Used RDKit to extract key physicochemical descriptors (e.g., MolWt, LogP, TPSA, HDonors, etc.).
* Used Mordred to extract additional molecular descriptors (e.g., topological, geometrical, charge-based).

**Handling Invalid Descriptors:**

* Dropped descriptors with missing or invalid values to ensure clean inputs.
* Combined valid RDKit and Mordred features.

**Feature Selection:**

* Performed correlation analysis between descriptors and Tg.
* Dropped weakly correlated and redundant features to reduce noise.

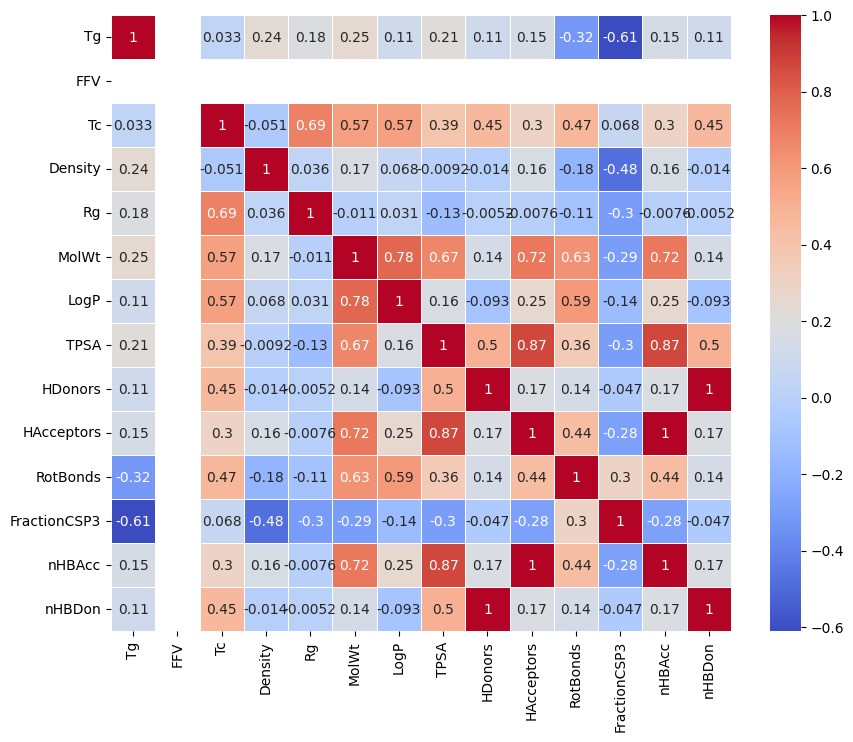
**Data Scaling:**

Applied StandardScaler to normalize features for effective model training.

**Train-Validation Split:**

Split the dataset into training and validation sets (e.g., 80% train, 20% validation) after preprocessing.

**Feature Correlation Heatmap**



Model Selection

**Model Selection**

**Linear Regression & Random Forest**

* Used as baseline models for comparison.
* Random Forest showed slightly better performance than Linear Regression.
* R² Score of Linear Regression: 0.40, R² Score of Random forest: 0.42

**Artificial Neural Network (ANN)**

* Built with 4 hidden layers (64 neurons each, ReLU activation).
* Loss: Mean Squared Error (MSE), Optimizer: Adam.
* Evaluated using RMSE, MAE, and R² Score.
* R² Score of ANN: 0.50

**Physics-Informed Neural Network (PINN)**

* Extended ANN by adding **physics-based loss** to guide predictions.
* Enforced known Tg trends:  
  + Tg increases with **MolWt**, **TPSA**
  + Tg decreases with **LogP**
* Final loss = MSE + 0.05 × Physics Loss

**Model Selection & Evaluation**

* Best performance observed with the PINN model.
* Performance Metrics of PINN models :

RMSE: 66.33

Mean Absolute Error (MAE): 52.09

R² Score: 0.54

Result

**Result**

| **Model** | **RMSE** | **MAE** | **R² score** |
| --- | --- | --- | --- |
| Linear Regression | 75.86 | 61.69 | 0.40 |
| Random Forest | 74.11 | 59.30 | 0.42 |
| ANN | 68.83 | 53.50 | 0.50 |
| PINN | 66.33 | 52.09 | 0.54 |