IOP Institute of Physics

Physical Aspects of Polymer Science 2024 (PAPS 2024)

The Tony Davies High Voltage Laboratory

University of Southampton

Machine Learning-Enhanced Metadata Analysis for Identifying Polymer Compositions

Sunny Chaudhary*, Orestis Vryonis and Paul Lewin

The Tony Davies High Voltage Laboratory, Department of Electronics and Computer Science, University of Southampton, Southampton, UK

Introduction

In materials science, accurately predicting and identifying polymer compositions from limited experimental data could potentially help in advancing material design, sustainability, and faster development. Traditional methods for analysing aged or complex polymers are often time-consuming and costly, requiring extensive measurements. Machine learning along with meta-data analysis and curation of an un-biased dataset offers a promising alternative, enabling quicker and notably accurate identification of materials or material properties.

The aim is to integrate machine learning with metadata analysis to predict polymer-filler combinations from *independently measured property values*. As an introductory study, a machine learning *pipeline using ensemble methods*, including *KNN, Random Forest and Neural Networks* (MLP) with *XGBoost* as the final estimator combined via *stacking,* was developed. The model was fine-tuned using *Bayesian optimization* for efficient *hyper-parameter tuning*. All individual and combinations of models were evaluated for accuracy and computational efficiency.

Evaluation was done on a dataset of various polymer-filler combinations obtained from *NanoMine*, the performance was assessed using *top-k accuracy metrics* and *cross-validation score,* based on the ability to correctly identify likely compositions based on *user input*. The model predicts the top three polymer-filler combinations with associated confidence levels.

Model achieved an accuracy score of 0.968 (96.8%) where the top three predictions consistently included the correct polymer-filler combinations.

Dataset Visualization and Analysis

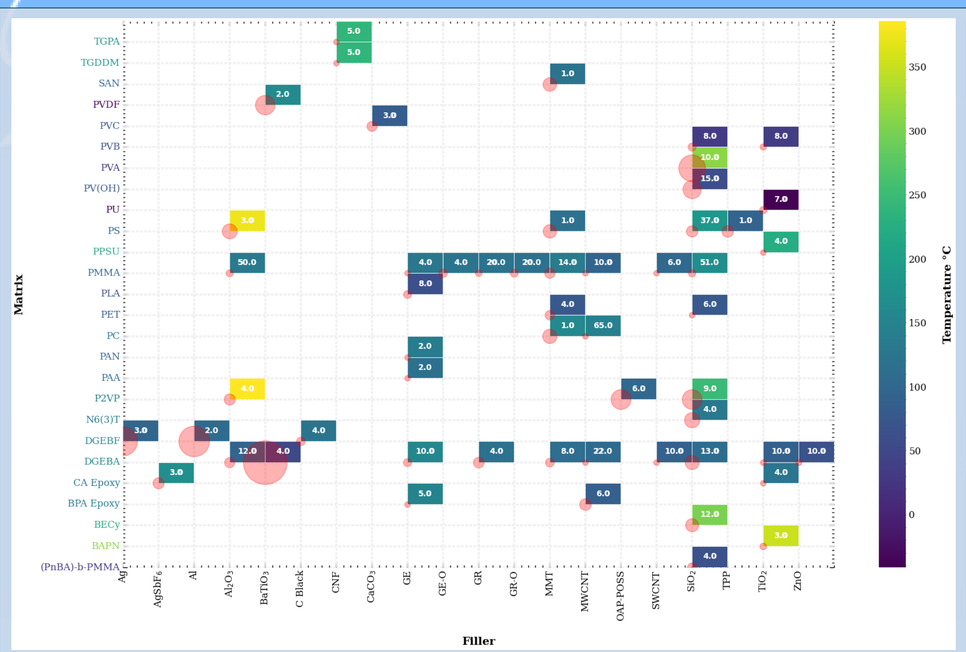

Figure 1*: The dataset for Glass Transition Temperature (T_g) obtained from NanoMine for various matrix and filler combinations. Each point represents a polymer-filler combination, with circle size indicating the average filler volume fraction and colour denoting the average T_g. Annotations show the number of samples per combination. The x-axis colour reflects the T_g of the unfilled polymer matrix.*

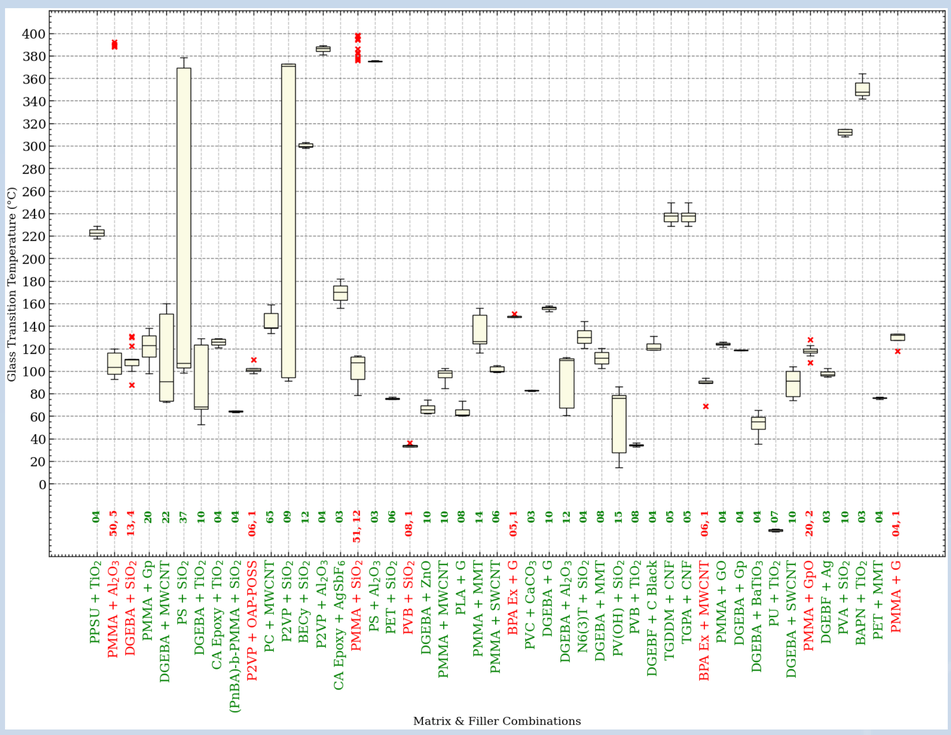

Figure 2*: Boxplot of T_g for various polymer matrix and filler combinations. The plot illustrates the distribution of T_g values for combinations. Outliers are marked with red crosses. The numbers below each combination indicate the sample size and, where applicable, the number of outliers identified.*

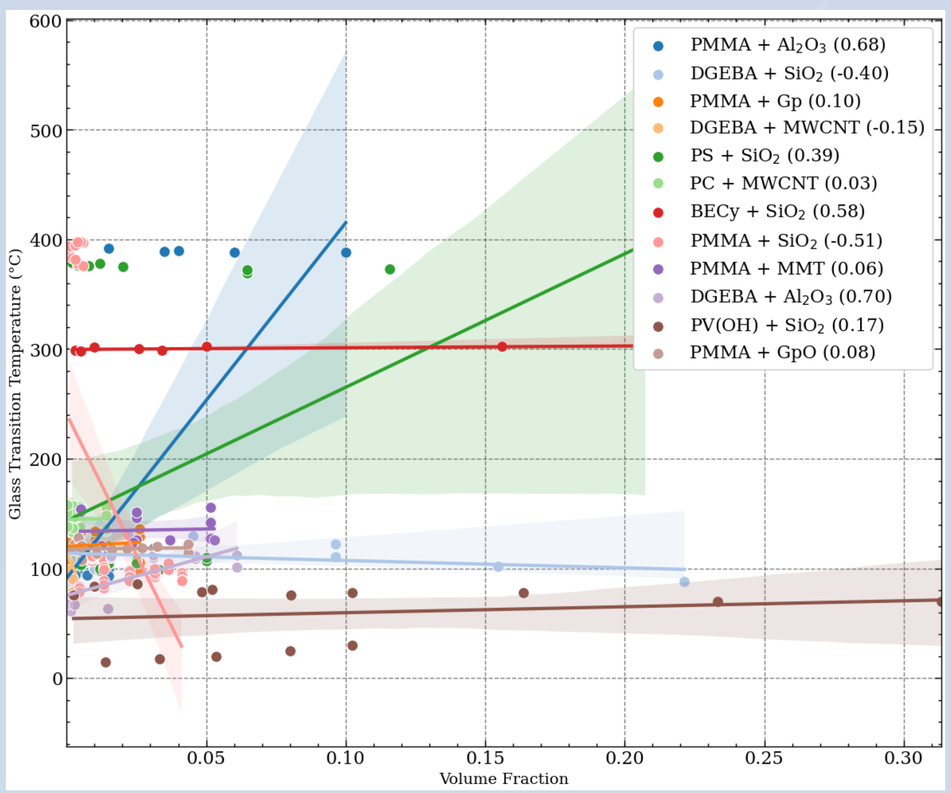

Figure 3: *Scatter plot with regression lines showing the correlation between Volume Fraction and Glass Transition Temperature (Tg) for various polymer matrix and filler combinations. with Corresponding correlation coefficient (r) provided in the legend. The regression lines highlight the strength and direction of these correlations.*

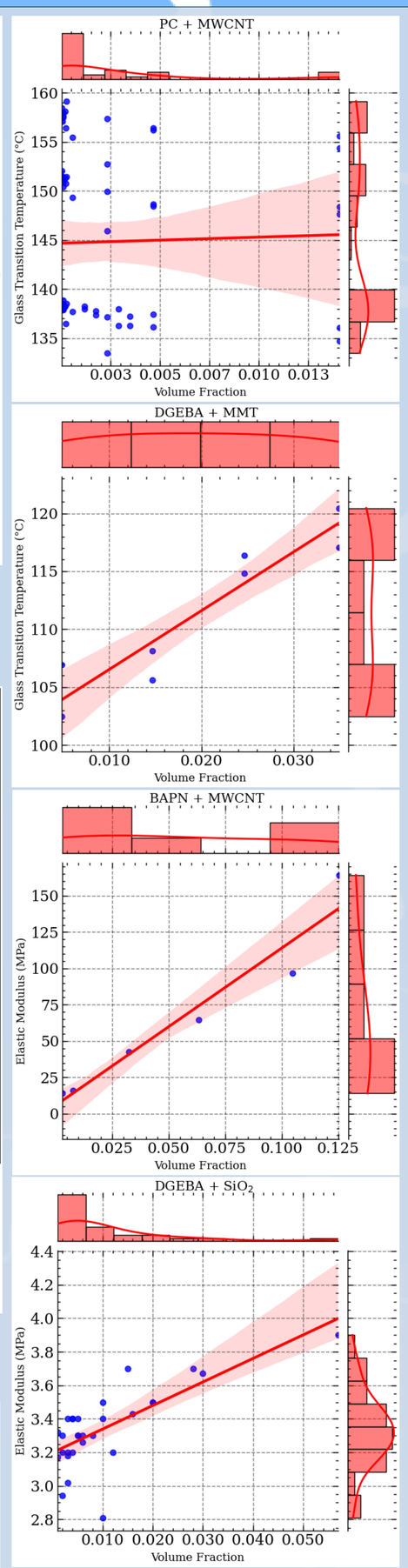

Figure 4: *Joint plot illustrating the relationship between Volume Fraction and different properties for various nanocomposites. The main plot shows correlation with a regression line indicating the trend, while the shaded area represents the confidence interval. The marginal histograms display the distribution of Volume Fraction (top) and Measured Property (right) values. Each blue data point corresponds to an individual study for the specific polymer-filler combination.*

Model Selection

KNN: Simple and effective for small datasets. Performance degrades with larger datasets.
RF: Robust and effective for small to moderate datasets. Struggles with very large datasets dimensionality.
XGB: High accuracy and efficiency with large datasets. Complex to tune.
MLP: Model complex patterns with multiple layers. Requires extensive tuning; computationally expensive.

Table 1: *Performance comparison of tested machine learning models against accuracy and training time metrics*

Model Name	Accuracy (cross-validation score)	~Training Time (s)
RF	0.908	30
KNN	0.887	22
XGB	0.924	300
MLP	**0.852**	4700
RF & KNN	0.916	$t_{rf} + t_{knn} + 3$
RF & XGB	0.956	$t_{rf} + t_{xgb} + 12$
XGB & KNN	0.931	$t_{xgb} + t_{knn} + 7$
XGB, KNN & RF	0.954	$t_{xgb} + t_{knn} + t_{rf} + 8$
Ensemble (Stacked)	**0.968**	5200

Machine Learning Algorithm

Input: Dataset $D \in R^{n \times p}$, Model parameters m, K, T, Initial top k combinations $\text{TopCombinations}_{\text{orig}} \in R^k$, User-provided property value $\text{UserProperty} \in R$.

- **Load Data:** $D_i = \text{read\_csv}(\text{file\_path}_i)$, $D = \bigcup_{i=1}^{N} D_i$.
- **Create Target Variable:** $\text{Filler\_Matrix}_i = \text{Filler}_i + \_ + \text{Matrix}_i$.
- **Encode Labels:** $y = L(\text{Filler\_Matrix})$.
- **Prepare Features:** $X = [\text{VolumeFraction}, \text{GlassTransitionTemperature}]$.
- **Split Data:** $D_{\text{train}}, D_{\text{test}} = \text{train\_test\_split}(D)$.
- **Standardize Features:** $Z_{\text{train}} = S(X_{\text{train}})$, $Z_{\text{test}} = S(X_{\text{test}})$.
- **Hyperparameter Ranges:**
  - **XGBoost:** $\theta_{\text{XGB}} = \{n_{\text{estimators}} \in [50, 300], \textit{max\_depth} \in [3, 10], \textit{learning\_rate} \in [0.01, 0.3], \textit{subsample} \in [0.6, 1.0]\}$.
  - **RandomForest:** $\theta_{\text{RF}} = \{n_{\text{estimators}} \in [50, 300], \textit{max\_depth} \in [3, 10], \textit{min\_samples\_split} \in [2, 10]\}$.
  - **KNN:** $\theta_{\text{KNN}} = \{n_{\text{neighbors}} \in [3, 15], \textit{weights} \in \{\text{uniform, distance}\}, p \in [1, 2]\}$.
  - **MLP:** $\theta_{\text{MLP}} = \{\textit{hidden\_layer\_sizes} \in [(50, 200)], \textit{learning\_rate\_init} \in [1e-4, 1e-2], \textit{alpha} \in [1e-5, 1e-3]\}$.
- **Optimize using Optuna:** $\theta^* = \arg\max_\theta f(\theta)$, where $f(\theta) = \text{cross\_val\_score}(M_\theta, X_{\text{train}}, y_{\text{train}}, cv = 3)$.
- **Train Models:** $M_{\theta^*} = \text{StackingClassifier}(RF_{\theta^*}, KNN_{\theta^*}, MLP_{\theta^*}]$, final_estimator $= XGB_{\theta^*}$).
- **XGBClassifier:**
  - **Objective Function:** $\text{Obj}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$.
  - **Prediction:** $\hat{y}_i = \sum_{k=1}^{K} f_k(X_i)$.
- **RandomForestClassifier:**
  - **Ensemble Prediction:** $\hat{y} = \text{majority\_vote}(T_1(X), T_2(X), \ldots, T_M(X))$.
  - **Decision Tree:** $T_m(X) = \sum_{j=1}^{J} 1(X \in R_j^m) \cdot c_j^m$.
- **KNeighborsClassifier:**
  - **Prediction:** $\hat{y} = \arg\max_{c \in C} \sum_{i=1}^{k} 1(y_i = c)$.
  - **Distance Metric:** $d(x, x') = \left(\sum_{j=1}^{d} |x_j - x'_j|^p \right)^{\frac{1}{p}}$.
- **MLPClassifier:**
  - **Forward Propagation:** $\hat{y} = \sigma(W_2 \cdot \sigma(W_1 \cdot X + b_1) + b_2)$.
  - **Loss Function:** $L(y, \hat{y}) = -\sum_{c=1}^{C} y_c \log(\hat{y}_c)$.
- **Predict Confidence Levels:** $C_{\text{orig}} = M_{\theta^*}(X_{\text{test}})$.
- **Extract Top k Combinations:** $\text{TopCombinations}_{\text{orig}} = \text{Top k Combinations}(C_{\text{orig}})$.
- **Filter Data:** $D_{\text{filtered}} = D[\text{Filler\_Matrix} \in \text{TopCombinations}_{\text{orig}}]$.
- **Identify Property:** $\text{BestProperty} = \arg\max_{P_j} \text{Non-Null Count}(P_j)$.
- **Request Value:** $P_{\text{new}} = \text{User Input}$.
- **Prepare New Dataset:** $X_{\text{new}} = [\text{VolumeFraction}, P_{\text{new}}]$.
- **Apply SMOTE:** $X_{\text{resampled}}, y_{\text{resampled}} = \text{SMOTE}(X_{\text{new}}, y)$.
- **Standardize:** $Z_{\text{new}} = S(X_{\text{resampled}})$.
- **Optimize and Retrain:** $M_{\theta^*}^{\text{new}} = \text{StackingClassifier}([XGB_{\theta^*}^{\text{new}}, RF_{\theta^*}^{\text{new}}, KNN_{\theta^*}^{\text{new}}, MLP_{\theta^*}^{\text{new}}]$, final_estimator $= XGB_{\theta^*}^{\text{new}}$).
- **Combine:** $C_{\text{comb},i} = \frac{C_{\text{orig},i} + C_{\text{new},i}}{2}$.
- **Normalize:** $C_{\text{comb},i} = \frac{C_{\text{comb},i}}{\sum_{j=1}^{k} C_{\text{comb},j}} \times 100$.
- **Rank Combinations:** $\text{TopCombinations}_{\text{final}} = \text{Ranked}(\text{TopCombinations}_{\text{orig}}, C_{\text{comb}})$.
- **Return:** $\text{TopCombinations}_{\text{final}}$.

Conclusion

- *Significant variation exists among some studies* for a given filler, matrix, and volume fraction likely due to differences in sample preparation and experimental setups. *Most cases show general agreement*.
- From the *correlation analysis trends are not always clear* or within confidence intervals. *Strong trends align with established mechanisms*, such as T_g increasing with higher nanoparticle volume fractions where there is good compatibility and decreasing with poor compatibility.
- *KNN, XGB, MLP and RF yield similar results*, with *XGB showing the highest accuracy*. *KNN and RF are suited for smaller datasets but may struggle with larger ones. MLP* is *notably slower*. *XGB performs better overall.*
- *Stacking reduces overfitting and improves classification* by combining multiple models *but does not offer a significant accuracy advantage.*

References

Kurban, H., Kurban, M., Sharma, P., & Dalkilic, M. (2021). **Predicting Atom Types of Anatase TiO_2 Nanoparticles with Machine Learning**. *Key Engineering Materials*, 880, 89 - 94. https://doi.org/10.4028/www.scientific.net/KEM.880.89.

Appiah-Badu, N., Missah, Y., Amekudzi, L., Ussiph, N., Frimpong, T., & Ahene, E. (2022). **Rainfall Prediction Using Machine Learning Algorithms for the Various Ecological Zones of Ghana**. *IEEE Access*, 10, 5069-5082. https://doi.org/10.1109/ACCESS.2021.3139312.

Zheng, T., Yu, Y., Lei, H., Li, F., Zhang, S., Zhu, J., & Wu, J. (2021). **Compositionally Graded KNN-Based Multilayer Composite with Excellent Piezoelectric Temperature Stability. Advanced Materials**, 34. https://doi.org/10.1002/adma.202109175.

Jamie P. McCusker, Neha Keshan, Sabbir Rashid, Michael Deagen, Cate Brinson, and Deborah L. McGuinness. **NanoMine: A Knowledge Graph for Nanocomposite Materials Science.** 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part II. Springer-Verlag, Berlin, Heidelberg, 144–159. https://doi.org/10.1007/978-3-030-62466-8_10

*sc.chaudhary@soton.ac.uk