JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

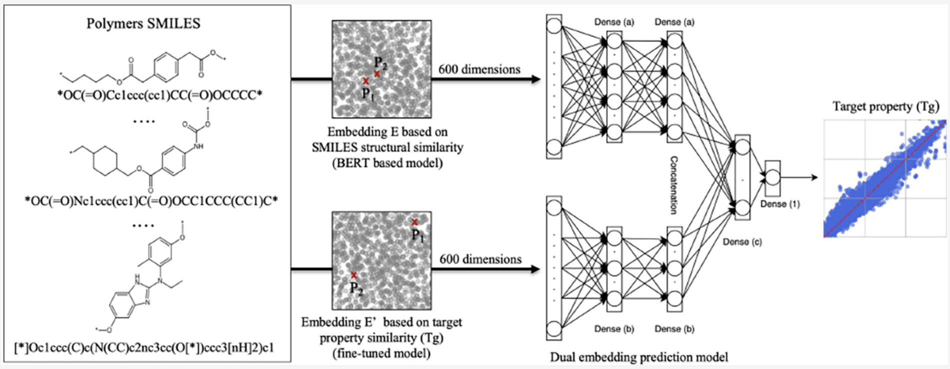
pubs.acs.org/jcim Article

Dual Embedding: A Fine-Tuned Language Model Approach for Accurate Polymer Glass Transition Temperature Prediction

Aymar Tchagoue,* Véronique Eglin, Jean-Marc Petit, Sébastien Pruvost, Jannick Duchet-Rumeau, and Jean-François Gérard

Cite This: J. Chem. Inf. Model. 2025, 65, 12342−12352 Read Online

ACCESS | Metrics & More | Article Recommendations


Polymers SMILES
OC(=O)Cc1ccc(cc1)CC(=O)OCCCC
....
OC(=O)Nc1ccc(cc1)C(=O)OCC1CCC(CC1)C
....
[*]Oc1ccc(C)c(N(CC)c2nc3cc(O[*])ccc3[nH]2)c1
P₂
P₁
600 dimensions
Embedding E based on SMILES structural similarity (BERT based model)
P₁
P₂
600 dimensions
Embedding E' based on target property similarity (Tg) (fine-tuned model)
Dense (a)
Dense (a)
Concatenation
Dense (b)
Dense (b)
Dense (c)
Dense (1)
Target property (Tg)
Dual embedding prediction model

ABSTRACT: Recent years have witnessed major advances in polymer informatics, yet accurately predicting polymer properties, such as the glass transition temperature (T_g), remains a challenge. Language models like BERT have been leveraged to derive embeddings from polymer representations (e.g., SMILES). However, similarity between embedding vectors in these latent spaces primarily reflects chemical structural similarity, with limited alignment to physicochemical properties. Here, we introduce a dual-embedding framework that enhances T_g prediction by combining a conventional BERT-based embedding with a fine-tuned counterpart explicitly trained so that vector similarity reflects proximity in T_g values. We evaluate our approach across four benchmarks: a heterogeneous data set compared against 25 machine learning baselines, along with three additional data sets focused on homopolymers and polyimides. The dual embedding outperforms standard BERT-based embeddings, achieving up to a 20% reduction in RMSE and surpassing alternative models such as graph-based and descriptor-based approaches. These results demonstrate that embedding molecular properties directly into representations can advance polymer informatics beyond structure-centric paradigms.

1. INTRODUCTION

Recent advances at the intersection of artificial intelligence and materials science have enabled the modeling of complex structure−property relationships from textual representations of molecules, such as SMILES, Simplified Molecular Input Line Entry System introduced in 1988 in.[1] This compact chemical language has opened new avenues for representing molecular structures computationally and predicting their associated physicochemical properties. In polymer informatics, this synergy is particularly promising: data-driven models offer the potential to predict key physicochemical properties without costly and time-consuming experiments.

This growing opportunity has already led to significant advances in the prediction of polymer properties.[2−6] Among these properties, one stands out for its particular interest: the glass transition temperature (T_g). This temperature represents the point at which a polymer transitions from a glassy state to a rubbery state. It is a crucial property for selecting materials for practical applications, yet it remains extremely difficult to predict. Many studies have aimed to predict it based on chemical structure.[2,7−10] Among these approaches, the use of language models for polymer encoding is becoming increasingly predominant.[2,6,11−13]

However, existing language model embeddings group polymers in the latent space mainly according to structural similarity. While this organization can enable predictions, it

Received: October 10, 2025
Revised: October 23, 2025
Accepted: October 24, 2025
Published: November 4, 2025

ACS Publications

https://doi.org/10.1021/acs.jcim.5c02469

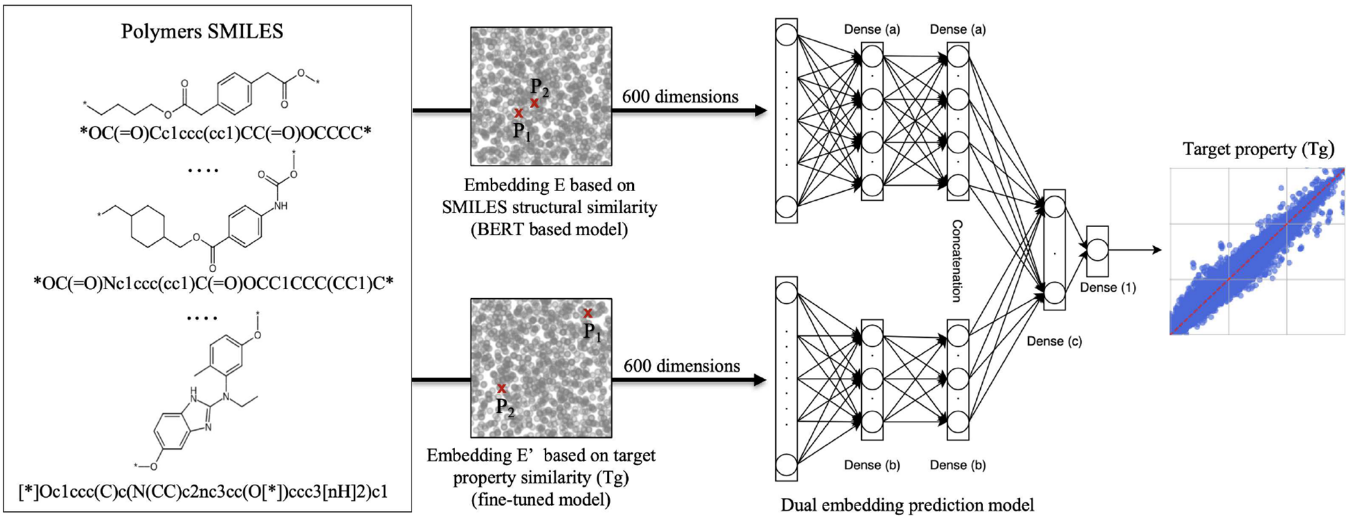

Polymers SMILES
OC(=O)Cc1ccc(cc1)CC(=O)OCCCC
....
OC(=O)Nc1ccc(cc1)C(=O)OCC1CCC(CC1)C
....
[*]Oc1ccc(C)c(N(CC)c2nc3cc(O[*])ccc3[nH]2)c1
P2
P1
600 dimensions
Embedding E based on SMILES structural similarity (BERT based model)
P1
P2
600 dimensions
Embedding E' based on target property similarity (Tg) (fine-tuned model)
Dense (a)
Dense (a)
Concatenation
Dense (b)
Dense (b)
Dense (c)
Dense (1)
Dual embedding prediction model
Target property (Tg)

Figure 1. Graphical abstract of the dual-embedding method.

does not fully exploit the potential of BERT-based embeddings for property prediction, particularly for T_g.

As illustrated in the graphical abstract Figure 1, we propose a dual-embedding approach that augments standard polymer embeddings with a fine-tuned version, in which vector proximity reflects similarity in T_g values rather than solely structural similarity derived from SMILES. By aligning the geometry of the latent space with property-based similarity, this method establishes a property-aware embedding strategy that more effectively captures relevant functional relationships.

We validate this approach on several data sets, including heterogeneous polymers, polyimides, and homopolymers, comparing our results with existing state-of-the-art models. Our findings demonstrate consistent improvements in prediction accuracy, with reductions in RMSE of up to 20% over standard embeddings. This shows the promise of fine-tuned language models not only for structure-based but also for property-driven representation learning in polymer design.

To demonstrate the effectiveness of our approach, this paper is organized as follows: Section 2 reviews the related work that is used as baselines; Section 3 presents the proposed dual-embedding methodology with a proof-of-concept example; Section 4 evaluates its performance across multiple data sets and compares it to existing methods; Section 5 provides additional analyses to further explore its potential; Section 6 offers a broader discussion of the results, including their limitations; and Section 7 concludes the study. The data and software used in this work are available in Section Data and Software Availability.

2. RELATED WORK

2.1. Standard Polymer Embeddings.

RDKit[14] is a widely used open-source cheminformatics toolkit that offers a broad range of tools for molecular modeling, data analysis, and machine learning in chemistry. One of its main features is the calculation of molecular descriptors, which quantitatively represent different aspects of molecular structure.

These RDKit descriptors are commonly employed in Quantitative Structure−Property Relationship (QSPR) models. They span a broad range of types, including topological, geometric, electrostatic, and solubility-related descriptors. While powerful, their effective use often requires careful feature engineering tailored to the specific property of interest. In the context of polymer informatics, such descriptors have been successfully used to predict properties like the glass transition temperature (T_g).[8,9] However, a known limitation is that QSPR descriptors may overlook subtle structural motifs or patterns not captured by predefined numerical features.

To address this, alternative representations such as Extended-Connectivity Fingerprints (ECFP)[15] have become increasingly popular. Implemented in RDKit as Morgan fingerprints, these representations encode molecular substructures using a circular hashing algorithm. Each substructure, defined by a central atom and its surrounding atoms within a chosen radius, is mapped to a unique identifier. For each molecule, the result is a vector representation in which each component corresponds to the frequency of a distinct substructure. Morgan fingerprints are generated directly from SMILES strings and have proven effective in property prediction tasks, particularly when used with deep learning models such as graph neural networks (GNNs) and transformers.[10] Nevertheless, they primarily capture local structural patterns and may struggle to encode more global molecular features. Additionally, their dimensionality increases with the diversity of patterns in the data set, raising the risk of the curse of dimensionality.

Beyond handcrafted or pattern-based descriptors, unsupervised representation learning has emerged as a powerful alternative. Techniques like Word2Vec,[16] originally developed for natural language processing, have been adapted for chemistry. A notable example is Mol2Vec,[17] which learns vector representations of molecular substructures in an unsupervised manner. Building further on this idea, deep sequence-based models such as Smiles2Vec[18] use Recurrent Neural Networks (RNNs) to directly process SMILES strings and learn features optimized for property prediction.

More recently, BERT-based models adapted to molecular data[2,6,11−13,19] have achieved state-of-the-art results in property prediction, encoding SMILES into fixed-size latent spaces where cosine similarity reflects sequence similarity. However, these embeddings remain globally unsupervised and do not guarantee alignment with property-based relationships, such as T_g proximity, a limitation we directly address in this work.

2.2. PolyBERT.

While general-purpose molecular language models like SMILES-BERT[12] and ChemBERTa[13] have demonstrated promising results, most remain trained on generic molecular data sets and do not account for the structural and functional specificity of polymers. This gap has recently been addressed by PolyBERT, a dedicated model tailored to polymeric structures. PolyBERT[2] is a polymer chemical embedding model based on DeBERTa[20] that treats the chemical structure of polymers as a chemical language. The proposed approach surpasses the best available methods for polymer property prediction based on handcrafted fingerprint schemes. It is trained on 100 million hypothetical PSMILES (Polymers SMILES). To the best of our knowledge, PolyBERT is the most recent open-source advancement in polymer language models. In the following, we analyze our novel approach applied to this model.

2.3. Sentence Transformers. To guide the embedding space toward capturing property-driven similarities, we draw inspiration from methods developed in natural language processing, specifically, Sentence-BERT (SBERT),[21] which aligns embeddings based on semantic closeness rather than surface form. Sentence-BERT is a modification of the pretrained BERT model that employs siamese and triplet network architectures to generate semantically meaningful sentence embeddings, which can be compared using cosine similarity. This approach significantly reduced the time to compute semantic similarity from 65 h with BERT to just 5 s, while maintaining similar accuracy. SBERT outperformed prior sentence embedding methods on semantic similarity and transfer learning benchmarks upon its release in 2019.[21] We draw on the core principles of this approach to design embeddings where vector similarity reflects property relevance rather than structural similarity between SMILES.

3. DUAL-EMBEDDING METHOD

From a chemical standpoint, it may seem counterintuitive to claim that structural similarity does not always correspond to similarity in properties, since molecular structure determines material properties. For instance, the glass transition temperature (T_g) mainly depends on two physical factors: chain stiffness and the strength of cohesive interactions.[22]

However, it is crucial to clarify what is meant by structural similarity, within the PolyBERT latent space, structural similarity is quantified using the cosine similarity between polymer embedding vectors. This metric essentially measures the resemblance between two character strings: the more similar the strings, the closer their cosine similarity is to 1. Yet, this measure carries an inherent bias with respect to property similarity. Small structural modifications—such as replacing a functional group, introducing a double bond instead of a single bond, or shifting a substituent from the meta to the para position—can have minor effects on string similarity, but substantial effects on a property. For example, Figure 2a illustrates the inequality between structural modifications, cosine similarity, and T_g variations in polyimides. Similarly, Figure 2b highlights the disproportion between PolyBERT latent space geometry and T_g differences: two polymer pairs may exhibit nearly identical cosine similarity values (0.59 vs 0.63), while their T_g differences are drastically distinct (263 vs 1 °C).

Therefore, predicting properties such as T_g directly from structural embeddings is theoretically consistent, but these embeddings do not explicitly encode the structural modifications that drive property changes. Thus, it is essential to exploit both raw and fine-tuned embeddings within predictive models to preserve complementary structural information that may otherwise be lost during fine-tuning. Similar strategies have been successfully implemented in recent studies. For instance, *MolPROP*,[23] which integrates SMILES syntax with graph-based structural features, demonstrated that simple fine-tuning of a general embedding is often insufficient. Likewise, dual-channel models for drug property prediction[24] effectively learn molecular representations that better capture property variations and fragment semantics. Finally, multitask generative modeling[25] has shown that training on multiple complementary objectives produces synergistic representations, enhancing the capture of complex structure−property relationships.

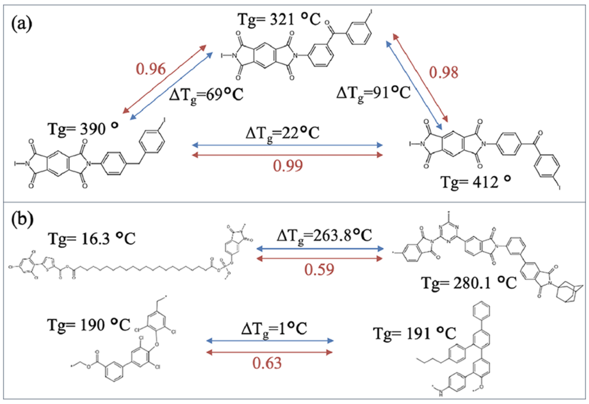


Figure 2. Cosine similarity (in red) of polyBERT embeddings for two polymer groups: (a) polyimides, shown by their repeating units with *I* denoting the junction, and (b) hypothetical polymers.

The proposed Dual-Embedding Method, illustrated in Figure 3, consists of training a glass transition temperature (T_g) prediction model using both a standard language model embedding (E) and a fine-tuned counterpart (E') that has been optimized to reflect T_g similarity, thereby capturing the substructural patterns that influence variations in T_g.

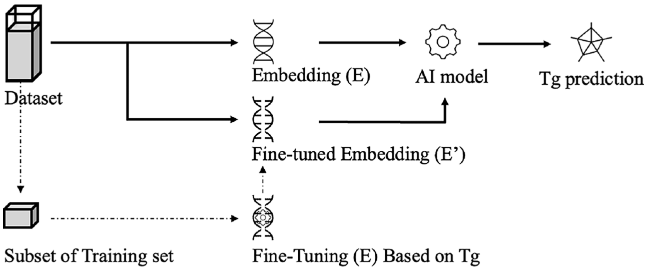


Figure 3. Dual-embedding method for T_g prediction.

Let A and B, be the embedding vectors of polymers P_1 and P_2 in E and A' and B' their corresponding embeddings in E'. We define the cosine similarity between two vectors V and W in an embedding space of dimension n as (eq 1):

$$\cos(V,\ W) = \frac{V\cdot W}{\sqrt{\sum_{i=1}^{n} V_i^2}\cdot\sqrt{\sum_{i=1}^{n} W_i^2}} \tag{1}$$

Our objective is to create the embedding E' such that $\cos(A', B')$ is correlated with the difference in T_g values between P_1 and P_2, while the original $\cos(A, B)$ primarily reflects the syntactic similarity of the polymers' SMILES representations.

3.1. Hypothetical Polymers Data Set. To perform our proof of concept, we used a subset of the open-source

PolyBERT data set,[2] which comprises 100 million hypothetical polymers derived from 13,000 experimental polymers data (not publicly available).

This subset contains data on more than 4000 polymers, including their PSMILES and nine properties, such as the T_g, the degradation temperature (T_d), the heat capacity (C_p), and their electron affinity (Eea). The histogram of the T_g values is provided below in Figure 4.

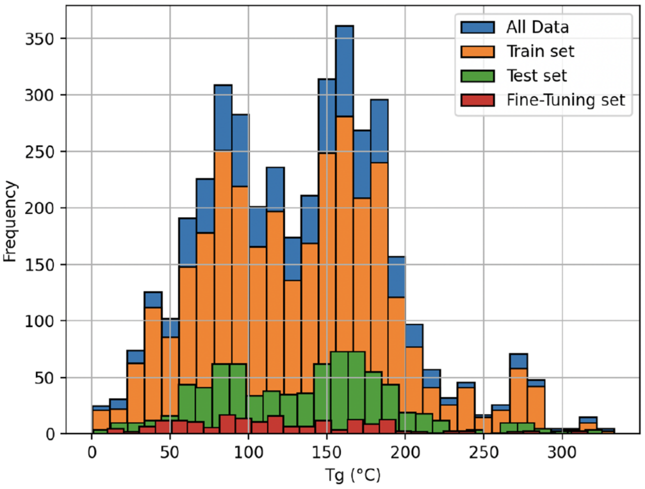


Figure 4. Combined T_g histograms with overlap.

3.2. Building a Data Set for Embedding Fine-Tuning.

3.2.1. Embedding Fine-Tuning Data Set Constitution. To avoid bias and ensure fair evaluation, the embedding fine-tuning data set is either distinct from the test set or drawn as a subset of the training set. Nandan Thakur and Gurevych[26] demonstrated that with more than 3000 data pairs, it is possible to reach the performance plateau when fine-tuning a well-structured encoder model. Our experiments confirmed this observation for the PolyBERT model. By selecting 200 polymers whose T_g distribution spans the entire range of the training set, we construct, through pairwise combination, a fine-tuning data set containing up to $\binom{200}{2} = 19{,}900$ unique pairs, which is sufficient to enable effective fine-tuning. For each pair, the difference between their T_g values is computed using various metrics. Table 1 illustrates the structure of the fine-tuning data set, where (*x*) denotes the difference in T_g, and (Score) represents the target similarity between the polymer vectors in the latent space of embedding E'.

Table 1. Example of Fine-Tuning Dataset

polymer 1	polymer 2	x	score
∗OCcccc1···∗	∗OCCOC···∗	15	0.8879
∗OCOccc1···∗	∗OCOcc1···∗	181	0.1321

3.2.2. Definition of Difference Metrics on Polymer T_g Pairs. The most natural way to compute the difference between two elements $a > 0$ and $b > 0$ is to use the absolute Euclidean (Eu) difference. However, other difference metrics can also be considered, such as the logarithmic (ln) difference, which emphasizes proportional rather than absolute differences and helps to relativize variations in T_g values. Similarly, the quadratic (*Q*) difference provides additional insight by penalizing large discrepancies more heavily.

Below are the corresponding metrics (eq 2) for (Eu), (ln), and (*Q*):

$$|a - b|, \quad |\ln(a/b)|, \quad \sqrt{|a^2 - b^2|} \tag{2}$$

3.2.3. Normalizing the Difference between T_g Values. The T_g differences vary widely in scale depending on the data set and the measurement units. To ensure meaningful comparison with cosine similarity, it is necessary to rescale these differences to the interval]0, 1[, such that smaller differences correspond to values closer to 1 (higher similarity), while larger differences approach 0 (lower similarity). Let x denote the difference between two T_g values. To prevent degradation during training, the data are analyzed using quantiles to remove outlier pairs with extreme values of x. Once this cleaning step is completed, let x_m denote the maximum T_g difference. We then define two normalization approaches, using a factor λ to control the sensitivity of the transformation.

We define the parametric scaling methods Min-max and Exponential as follows:

$$\mathrm{MM}(x) = 1 - (1 - \lambda)\frac{x}{x_m}, \quad \mathrm{Exp}(x) = e^{(x/x_m)\cdot\ln(\lambda)}$$

where $\mathrm{Exp}(x_m) = \mathrm{MM}(x_m) = \lambda$, with $\lambda < 1$. While MM provides a linear mapping, Exp penalizes larger x more heavily than smaller ones. Depending on the data set structure, one approach may be more advantageous than the other.

3.3. Embedding Fine-Tuning: Models and Results.

After constructing the fine-tuning data set with multiple scores, all using a fixed $\lambda = 0.90$, the BERT-based model is fine-tuned as illustrated in Figure 5, within a Siamese network. The loss function is the mean squared error between the cosine similarity of the embedding vectors V and W and the target score, which corresponds to the normalized difference of the polymers' T_g values. The fine-tuning is conducted using the pretrained checkpoint `kuelumbus/polyBERT` over 5 epochs with 10 warm-up steps and a fixed random seed of 42 to ensure reproducibility. We denote this by

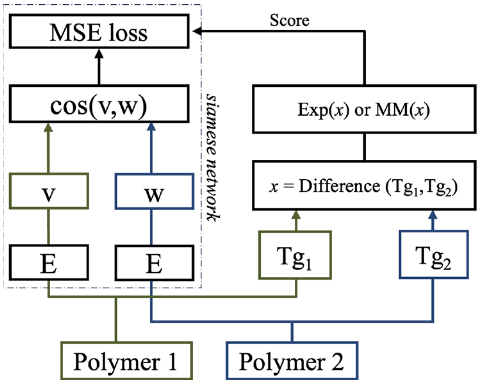


Figure 5. Fine-tuning embedding (E) with cosine similarity loss.[21]

- `PB`: Original PolyBERT model.
- `PB_Eu_MM` and `PB_Eu_Exp`: Fine-tuned PolyBERT using Euclidean difference with Min-Max normalization and Exponential normalization, respectively.
- `PB_ln_MM` and `PB_ln_Exp`: Fine-tuned PolyBERT using logarithmic difference with Min-Max normalization and Exponential normalization, respectively.

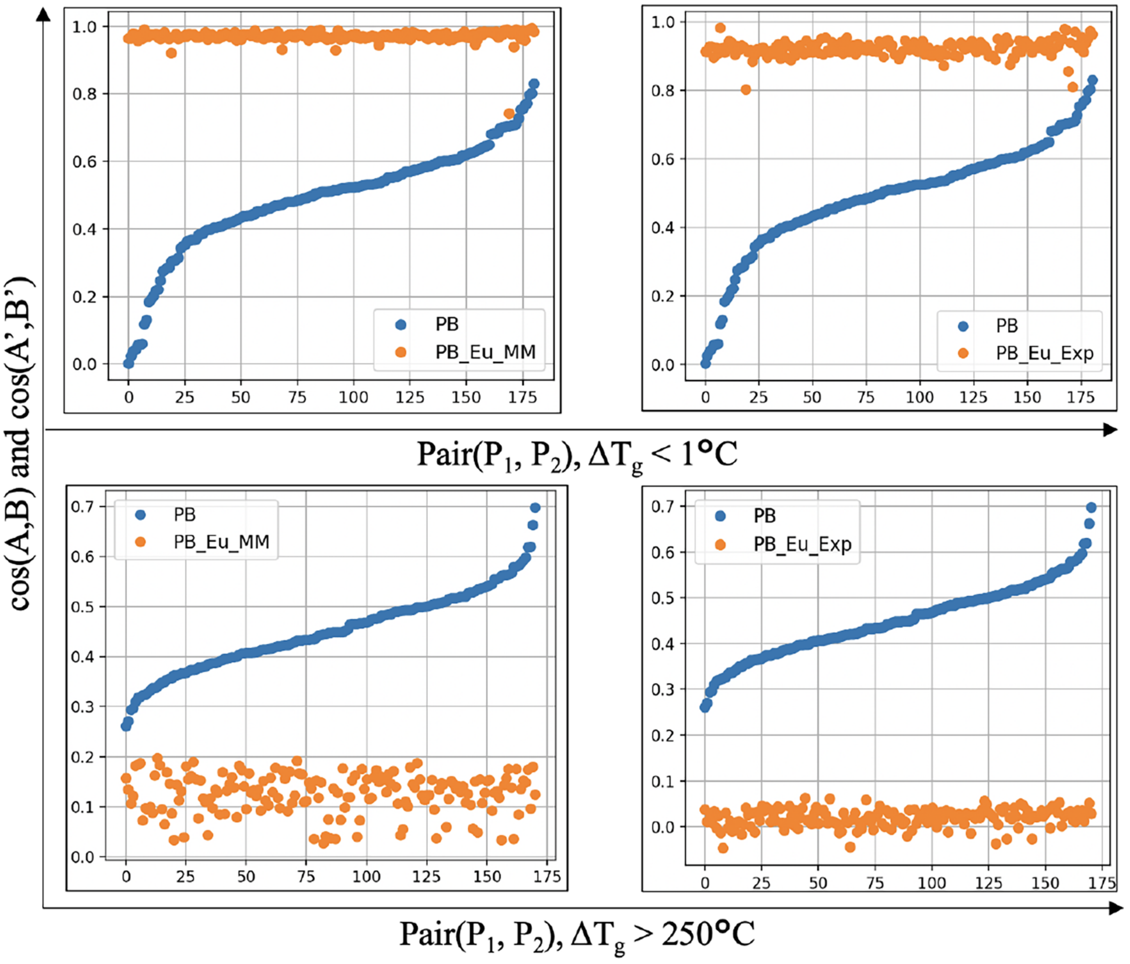


Figure 6. Cosine similarity of polymer pair embeddings from PB and fine-tuned PB (*PBEuMM* and *PBEuexp*), for two groups: $\Delta T_g < 1$ °C and $\Delta T_g > 250$ °C.

- PB_Q_MM and PB_Q_Exp: Fine-tuned PolyBERT using quadratic difference with Min-Max normalization and Exponential normalization, respectively.

To visualize the effectiveness of our fine-tuning, we formed two groups of polymer pairs (P_1, P_2) through filtering: one group contains polymers with similar glass transition temperatures (T_g), where $|T_{g1} - T_{g2}| < 1$ °C, and the other group contains polymers with significantly different T_g values, where $|T_{g1} - T_{g2}| > 250$ °C.

Figure 6 shows the cosine similarity of polymer pairs from both groups in the original *PB* embedding space and in the fine-tuned embedding spaces *PBEuMM* and *PBEuExp*. In the fine-tuned space, polymer pairs with similar T_g values exhibit cosine similarities close to 1, unlike in the original embedding. Conversely, pairs with different T_g values show similarities approaching 0 in the fine-tuned space, further confirming the effectiveness of the fine-tuning process.

Figure 7 shows the t-SNE projection of polymer vectors in both the PB and *PBEuMM* embeddings. Two groups of polymer pairs are highlighted: those with similar T_g values, which become closer in the fine-tuned embedding, and those with very different T_g values, which are now placed in diametrically opposite regions. The t-SNE visualization reveals a clear rearrangement of the latent space, indicating that the fine-tuned embeddings capture a different geometry of polymer relationships compared to the original PolyBERT representations.

3.4. T_g Prediction Models.

Given n samples with true values y_i and predicted values $\hat{y}_i$, we define:

The Mean Squared Error (MSE): $\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

The Root Mean Squared Error (RMSE): $\text{RMSE} = \sqrt{\text{MSE}}$

The Mean Absolute Error (MAE): $\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

The Coefficient of Determination (R^2): $R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$ where $\bar{y}$ is the mean of the true values.

After obtaining the fine-tuned models (E') from the initial embeddings (E), each SMILES string is converted into two 600-dimensional vectors, each corresponding to the structural and ΔT_g-based embeddings. We then design a model that takes these vectors as input and predicts the T_g as output. To integrate these complementary representations, we designed a dual-branch feedforward neural network (FFNN) in which each embedding is processed separately through two dense layers before being merged into a joint latent layer for T_g prediction. This design allows the model to capture both the full chemical structure (from E) and the property-based structure (from E'), while also enabling an explicit assessment of their individual contributions.

The choice of the number of neurons per layer reflects the complexity of the information to be processed. Since the first

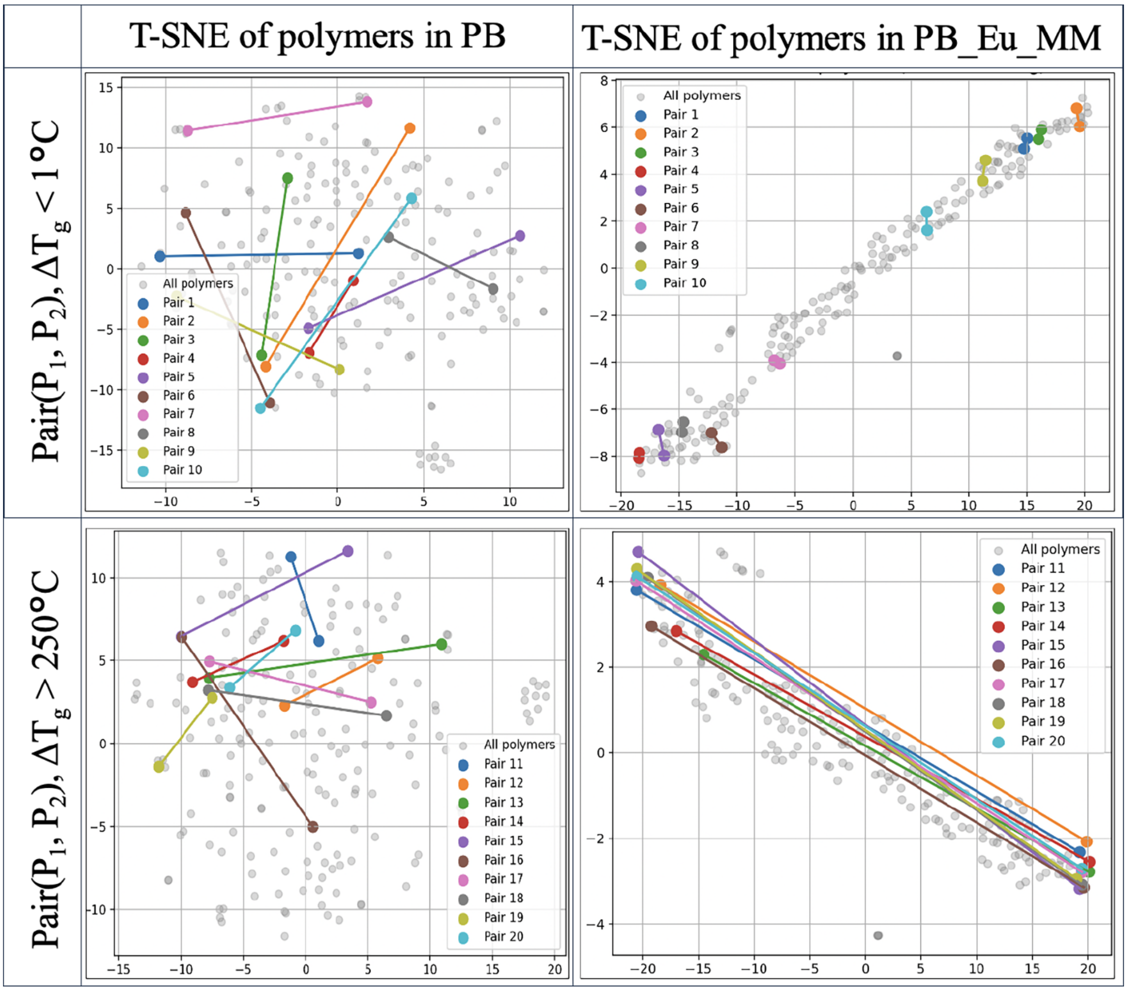


Figure 7. t-SNE of polymer embeddings from PB and the fine-tuned model (*PBEuMM*). Two groups of polymer pairs are highlighted: ΔT_g < 1 °C and ΔT_g > 250 °C.

layer corresponds to the initial PB structural embedding, it is assigned more neurons per layer in its branch compared to the fine-tuned embedding branch. This modeling strategy, illustrated in Figure 8, was compared with alternative architectures, and the results indicate that the dual-branch design provides optimal performance across all evaluations.

For the hypothetical polymer data set, the model was trained with dense architecture of sizes $a = 32$, $b = 8$, and $c = 2$, using a ReLU activation function. Training was performed with a learning rate of 0.0001, a batch size of 16, and optimized using the MAE loss. The data set was split into 80% training and 20% testing, with a validation split of 20% applied to the training set. The number of epochs is not specified because to avoid overfitting we use Keras early stopping with a patience of 5 to restore the best weights, and a maximum epochs of 1,000. The use of early stopping ensures that hyperparameters such as batch size or learning rate do not exert a significant influence on the final performance, as training is halted once convergence is achieved regardless of these variations.

The following table presents the results of T_g prediction, trained on 2566 samples, validated on 641 samples, and tested on 803 samples. The reported RMSE and R^2 values correspond to the test set. In Table 2, we denote $(E|E')$ as the model built using both embeddings (one per branch) and $(E|)$ or $(|E')$ as the model built using only one of the two branches.

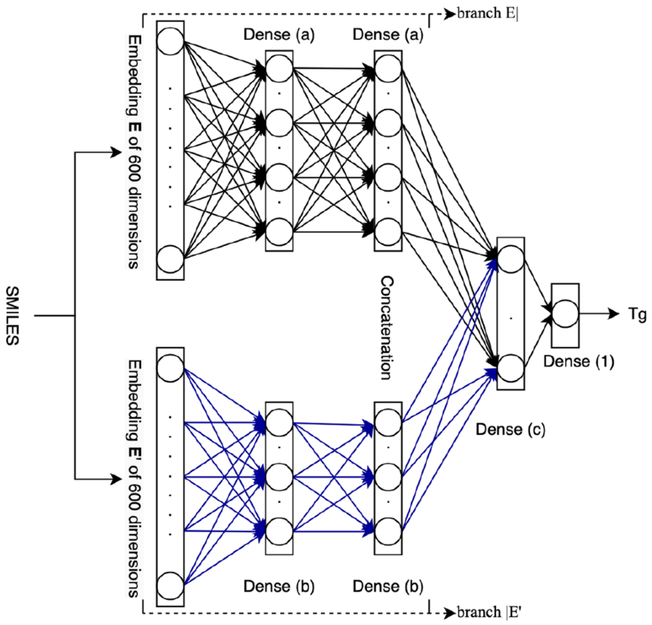


Figure 8. T_g prediction model.

We can observe in Table 2 that when considering only the branch consisting of a fine-tuned model $(|E')$, the RMSE

Table 2. Ablation Study on T_g Prediction Using Different Embedding Configurations on the Hypothetical Polymer Data Set[a]

models/embeddings	R^2	RMSE
PB|	0.893	19.31
|PB_Eu_MM	0.837	23.91
|PB_Eu_Exp	0.853	22.72
|PB_Q_MM	0.870	21.33
|PB_Q_Exp	0.865	21.73
|PB_ln_MM	0.846	23.25
|PB_ln_Exp	0.873	21.07
PB| PB_Eu_MM	0.910	17.74
PB|PB_Eu_Exp	**0.918**	**16.93**
PB|PB_Q_MM	0.917	17.08
PB|PB_Q_Exp	0.917	17.07
PB|PB_ln_MM	0.912	17.53
PB| PB_ln_Exp	0.910	17.70

[a] Mean and Std of the test set: 135 and 59 °C.

values are all worse compared to the original model ($E|$ = PB|). However, the combination of fine-tuned embeddings with the original one results in a direct improvement in the performance of all models containing a Dual Embedding ($E|E'$). Moreover, all these models exhibit an RMSE performance superior to that of the original model.

From this proof of concept, we conclude that the fine-tuned embedding based on T_g provide complementary information in the Dual-Embedding model.

We also conclude that the difference metric does not necessarily have a significant influence. Therefore, moving forward, we consider only the Euclidean difference to quantify differences in T_g values.

4. DUAL-EMBEDDING APPLICATIONS

4.1. Benchmark Models. In order to compare our work with existing studies on T_g (°C) prediction, which cover various models and embeddings, we referred to an established benchmark on heterogeneous polymer T_g prediction.[10]

This benchmark stands out because it presents 25 different methods for predicting T_g using diverse embeddings, such as the Morgan fingerprint,[27] a molecular embedding (ME) based on Word2Vec, molecular graph embeddings, the conversion of polymers into 2D images, and the use of RDKit descriptors. Furthermore, this study combines these embeddings with various models, including RNN, Support Vector Machines (SVM), Feedforward Neural Networks (FFNN), 1D and 2D Convolutional Neural Networks (CNN), Graph Convolutional Neural Networks (GCNN), and many others.

Only the benchmark test set, their labels, and the predictions are provided as open source. To enable comparison with this benchmark without access to their training set, we propose as presented in Figure 9 to use a pretrained model based on the architecture and data described in Section 3.4, which we fine-tune on a portion of the benchmark test set.

To ensure a fair evaluation, we performed a random split of the benchmark test set into fine-tuning `(30% (170))` and test subsets `(70% (396))`, without stratification based on polymer family, chemical structure, or property distribution. Consequently, the fine-tuning and test sets are fully disjoint at the sample level, avoiding any direct data leakage. The Kolmogorov–Smirnov test revealed no significant difference between the two distributions (D = 0.067, p = 0.63), indicating that neither set was favored.

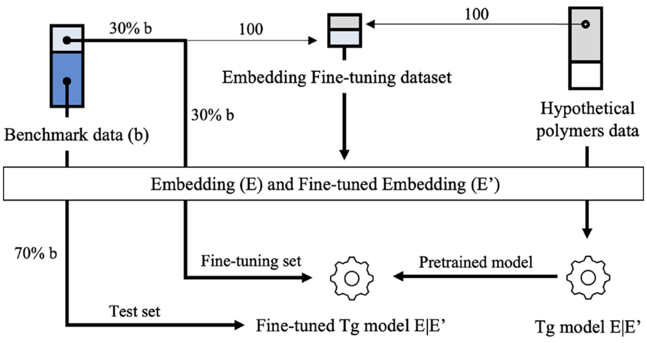


Figure 9. Benchmark T_g prediction model.

4.1.1. Embedding Fine-Tuning. To fine-tune our embedding while ensuring fair evaluation, we select 100 samples each from the hypothetical polymer training set and the benchmark fine-tuning set, in such a way that their T_g distribution spans the entire range of the data set, resulting in 200 representative data. This is essential because we need to adjust our embedding model to the two types of data, which are considerably different. Once this is done, we apply, as in our proof of concept, a pairwise combination of the polymers to create the fine-tuning data set for the embedding.

We note the following: `PB_Merge_MM` and `PB_Merge_Exp` refer to PolyBERT fine-tuned on 200 merged data, using euclidean difference with Min-Max and Exponential normalization, respectively.

4.1.2. T_g Model Prediction Fine-Tuning. The overall T_g prediction model is illustrated in Figure 9, where a pretrained model is constructed following the approach presented in Section 3.4, except that it utilizes the fine-tuned embeddings `PB_Merge_MM` or `PB_Merge_Exp`. The pretrained models are then fine-tuned on the benchmark fine-tuning subset (30%) and evaluated on the remaining (70%). To optimize prediction performance, the only modifications applied to all models were the use of mean squared error as the loss function (`loss = MSE`) and a reduced learning rate (`lr = 0.001`).

Table 3 summarizes the results, where BM refers to the best model among the 25 proposed in,[10] and BM70 denotes the performance of BM on the selected 70% test subset.

We observe that PolyBERT alone already performs competitively, slightly outperforming the best model (BM) from the benchmark, which uses a Word2Vec-based molecular embedding combined with a Deep Neural Network (DNN). While the training data sets are not identical, the comparison

Table 3. Ablation Study on T_g Prediction Using Different Embedding Strategies on the Benchmark Polymer Data Set[a]

models/embeddings	R^2	RMSE
BM[10]	0.64	66.58
BM_70	0.63	64.46
PB|	0.64	64.20
|PB_Merge_Exp	0.62	66.21
|PB_Merge_MM	0.64	63.76
PB|PB_Merge_Exp	0.65	63.66
PB|PB_Merge_MM	**0.68**	**60.76**

[a] Mean and Std of the test set: 98 and 108 °C.

Table 4. Ablation Study on T_g Prediction Using Different Embedding Configurations on Polyimides Data Set[a,b]

<table>
<tr><th>models</th><th>data</th><th>R^2</th><th>RMSE</th><th>MAE</th></tr>
<tr><td>GCNN/Syn[8]</td><td>syn</td><td></td><td></td><td>14.9</td></tr>
<tr><td>PB|</td><td>syn</td><td>0.67 ± 0.1</td><td>23.65 ± 3.2</td><td>4.15 ± 0.25</td></tr>
<tr><td>|PB_PI_MM</td><td>syn</td><td>0.77 ± 0.06</td><td>20.19 ± 3.3</td><td>3.77 ± 0.25</td></tr>
<tr><td>|PB_PI_Exp</td><td>syn</td><td>0.71 ± 0.07</td><td>22.42 ± 2.85</td><td>4.07 ± 0.19</td></tr>
<tr><td>PB|PB_PI_MM</td><td>syn</td><td>0.76 ± 0.1</td><td>19.69 ± 4.0</td><td>3.69 ± 0.38</td></tr>
<tr><td><b>PB|PB_PI_Exp</b></td><td><b>syn</b></td><td><b>0.80 ± 0.05</b></td><td><b>18.54 ± 2.4</b></td><td><b>3.65 ± 0.19</b></td></tr>
<tr><td>GCNN/exp[8]</td><td>exp</td><td></td><td></td><td>28.1 ± 5.7</td></tr>
<tr><td>PB|</td><td>exp</td><td>0.56 ± 0.18</td><td>33.13 ± 8.8</td><td>4.85 ± 0.64</td></tr>
<tr><td>|PB_PI_MM</td><td>exp</td><td>0.31 ± 0.31</td><td>41.74 ± 11.87</td><td>5.30 ± 0.75</td></tr>
<tr><td>|PB_PI_Exp</td><td>exp</td><td>0.50 ± 0.25</td><td>35.59 ± 11.33</td><td>4.93 ± 0.67</td></tr>
<tr><td><b>PB|PB_PI_MM</b></td><td><b>exp</b></td><td><b>0.74 ± 0.09</b></td><td><b>26.04 ± 8.0</b></td><td><b>4.13 ± 0.51</b></td></tr>
<tr><td>PB|PB_PI_Exp</td><td>exp</td><td>0.70 ± 0.09</td><td>27.78 ± 7.9</td><td>4.27 ± 0.49</td></tr>
<tr><td>GCNN/Poly[8]</td><td>syn + exp</td><td></td><td></td><td>22.5 ± 5.1</td></tr>
<tr><td>PB|</td><td>syn + exp</td><td>0.71 ± 0.10</td><td>27.36 ± 9.0</td><td>4.39 ± 0.64</td></tr>
<tr><td>|PB_PI_MM</td><td>syn + exp</td><td>0.58 ± 0.21</td><td>31.91 ± 7.71</td><td>4.64 ± 0.62</td></tr>
<tr><td>|PB_PI_Exp</td><td>syn + exp</td><td>0.65 ± 0.13</td><td>29.31 ± 6.68</td><td>4.44 ± 0.57</td></tr>
<tr><td>PB|PB_PI_MM</td><td>syn + exp</td><td>0.74 ± 0.10</td><td>25.44 ± 7.6</td><td>4.16 ± 0.54</td></tr>
<tr><td><b>PB|PB_PI_Exp</b></td><td>syn + exp</td><td><b>0.74 ± 0.13</b></td><td><b>24.82 ± 7.0</b></td><td><b>4.13 ± 0.54</b></td></tr>
</table>

[a] Mean and Std of the synthetic data: 500 and 42 °K. [b] Mean and Std of the experimental data: 546 and 53 °K.

remains informative, as both approaches are evaluated on the same test set (BM_70). Nevertheless, the core argument remains valid: the dual-embedding approach outperforms the single-embedding model, with the best results obtained when using Min-Max normalization.

4.2. Polyimides Models.

Polyimides are a class of polymers characterized by the presence of at least one *imide linkage*, which can appear either as part of an open-chain structure or as a heterocyclic unit integrated into the polymer backbone. An imide linkage is typically represented as `'R−CO−NH−CO−R'`, where two carbonyl groups (`CO`) are bonded to the same nitrogen atom (`NH`).

Volgin et al.[8] develop the prediction of polyimide glass transition temperature T_g (°K), focusing on GCNN. They also provide, as open-source data, two data sets: a synthetic data set containing more than 6 million data and an experimental data set with 214 data.

They developed multiple approaches, in this study we consider three: a first predictive model built on 1,000 synthetic data (`GCNN/Syn`), a second one based on experimental data (`GCNN/exp`), and a third one consisting of the fine-tuning of (`GCNN/Syn`) on the experimental data, referred to as (`GCNN/Poly`). All these models are built using the 10-fold cross-validation technique.

In order to perform our dual-embedding approach, we selected 200 unique data whose T_g distribution spans the entire range of the synthetic data set for embedding fine-tuning. We denote by `PB_PI_MM` and `PB_PI_Exp` the PolyBERT model fine-tuned on polyimide synthetic data using euclidean difference with Min-Max and Exponential normalization, respectively.

We constructed the corresponding dual-embedding T_g prediction model as described in Section 3.4, using the following modified parameters for optimization: We use lr = 0.001 for models trained on synthetic data (`syn`) or pretrained on synthetic data and then trained on experimental data (`syn+exp`), and lr = 0.0001 for models trained only on experimental data (`exp`). The results provided in Table 4 represent the mean ± standard deviation (Std) from the 10-fold cross-validation.

As presented in Table 4, in terms of MAE, the model built by considering only the PolyBERT branch already surpasses the GCNN approach developed in.[8] Furthermore, all the dual-embedding approaches perform better on all three data sets, with a significant improvement observed in the R^2 and RMSE values on both the synthetic and experimental data sets, with up to 20% reduction in RMSE (from 23.65 to 18.54).

4.3. Homopolymers Models.

Homopolymers are polymers whose structure is made up of repeating units of a single kind of monomer. Casanola-Martin et al.[9] address the prediction of homopolymers log(T_g) using quantitative structure−property relationship (QSPR) descriptors. They utilized various machine learning models, including multiple linear regression (MLR), and support vector machine (SVM). The open-source data set contains 683 samples for training and 219 for testing.

We extracted 200 data from the training set to perform PolyBERT fine-tuning. We denote by `PB_Hm_MM` and `PB_Hm_Exp` the PolyBERT model fine-tuned on homopolymer data using euclidean difference with Min-Max and Exponential normalization, respectively.

We constructed the corresponding dual-embedding log(T_g) prediction model as indicated in Section 3.4 with the following modified parameters for optimization: lr = 0.02, and batch = 13.

Table 5 presents the results, which we explain as follows. One of the particularities of this study is that the prediction is performed on log(T_g) rather than T_g directly, this is done to ensure comparability, as the publication with which we compare developed their model using log(T_g) to optimize prediction performance.

The PolyBERT model alone achieves the lowest performance. However, when incorporated into the dual-embedding process, its performance improves significantly, surpassing QSPR-MLR and approaching that of QSPR-SVM in terms of R^2 on this data set. These results demonstrate that the dual-embedding approach consistently outperforms the original embedding, highlighting the effectiveness of property-aware fine-tuning.

Table 5. Ablation Study on log(T_g) Prediction Using Different Embedding Configurations on Homopolymers Data Set[a]

models	R^2	RMSE
QSPR-SVM[9]	0.77	
QSPR-MLR[9]	0.617	0.086
PB|	0.547	0.093
|PB_Hm_Exp	0.570	0.091
|PB_Hm_MM	0.563	0.091
PB|PB_Hm_Exp	0.661	0.080
PB|PB_Hm_MM	**0.667**	**0.080**

[a] Mean and Std of the test set: 2.552 and 0.139.

Table 6. Ablation Study on T_d (°K), C_p (J/gK), and E_{ea} (eV) Prediction Using Different Embedding Configurations on Hypothetical Polymers Data Set

models	R^2	RMSE
	T_d (°K), mean = 652, std = 50	
PB|	0.64	30.46
|PB_Td_MM	0.67	29.34
|PB_Td_Exp	0.67	29.06
PB|PB_Td_MM	**0.71**	**26.97**
PB|PB_Td_Exp	0.71	27.05
	C_p (J/gK), mean = 1.31, std = 0.128	
PB|	0.85	0.05
|PB_Cp_MM	0.76	0.06
|PB_Cp_Exp	0.68	0.07
PB|PB_Cp_MM	**0.87**	**0.05**
PB|PB_Cp_Exp	0.85	0.05
	E_{ea} (eV), mean = 1.90, std = 0.46	
PB|	**0.92**	**0.13**
|PB_Eea_MM	0.87	0.16
|PB_Eea_Exp	0.84	0.18
PB|PB_Eea_MM	**0.92**	**0.13**
PB|PB_Eea_Exp	0.91	0.14

5. ADDITIONAL ANALYSES

5.1. Other Properties Predictions with the Dual Embedding. While our approach has shown promising results in predicting T_g in previous models, it is crucial to assess whether the same methodology can generalize to other polymer properties. Given that the T_g prediction relies heavily on molecular structure and interactions, it stands to reason that other properties may also benefit from the dual-embedding approach. To investigate this, we use the hypothetical polymers subset (4000) with three additional properties introduced in Section 3.1, namely: degradation temperature T_d (°K), heat capacity C_p (J g^{-1} K^{-1}), and electron affinity E_{ea} (eV). We denote by

- PB_Cp_MM and PB_Cp_Exp the PolyBERT model fine-tuned on 200 hypothetical polymers data using euclidean difference on C_p with Min-Max and Exponential normalization, respectively.
- PB_Eea_MM and PB_Eea_Exp the PolyBERT model fine-tuned on 200 hypothetical polymers data using euclidean difference on E_{ea} with Min-Max and Exponential normalization, respectively.
- PB_Td_MM and PB_Td_Exp the PolyBERT model fine-tuned on 200 hypothetical polymers data using euclidean difference on T_d with Min-Max and Exponential normalization, respectively.

For these new properties, we applied the same dual-embedding methodology as described in Section 3.4, adjusting the hyperparameters for the C_p and E_{ea} prediction models. More specifically, we set the learning rate to lr = 0.01 for the C_p models and adjusted the batch size for the E_{ea} models to batch = 3 to optimize performance. The corresponding results, along with the mean and standard deviation (Std) of the test set, are presented in Table 6.

Table 6 shows that the dual-embedding framework performs well for T_d, provides a minor improvement for C_p, and remains constant for E_{ea}. To understand the limited performance for C_p and E_{ea}, further analysis and additional experiments are necessary to verify the generality of this observation, as was done for T_g. We therefore consider this a direction for future studies. However, for this specific data set configuration, the prediction using a single embedding already appears to have reached a performance plateau for E_{ea}, with R^2 = 0.92. Nevertheless, implementing the dual-embedding approach remains worthwhile.

5.2. T_g Prediction by Fine-Tuning Embeddings With Properties Different from T_g. Another interesting idea we wanted to verify is whether fine-tuning the embedding on a property different from T_g captures relevant complementary information for the dual-embedding approach on T_g prediction. This is most likely the case for properties that are not correlated with T_g across a wide range of polymers, including T_d (thermal degradation temperature), C_p (heat capacity at constant pressure), and E_{ea} (electron affinity), which reflect distinct physicochemical dimensions.

To verify our hypothesis, we applied the same methodology as in the proof of concept (Section 3), using different fine-tuned embeddings centered around the properties T_d, C_p, and E_{ea}. The results of this analysis presented in Table 7 show that all dual embeddings based on properties different from T_g outperform the native embedding in predicting T_g, although they do not perform as well as the embedding fine-tuned specifically on T_g values. This demonstrates that fine-tuning language model embeddings on a specific property helps capture relevant patterns that enhance the prediction of T_g.

Table 7. T_g Prediction with Fine-Tuned Embedding on T_d, C_p, and E_{ea}[a]

models	R^2	RMSE
PB|	0.893	19.31
PB|PB_Eu_MM	0.910	17.74
PB|PB_Eu_Exp	**0.918**	**16.93**
PB|PB_Td_MM	0.900	18.71
PB|PB_Td_Exp	0.902	18.51
PB|PB_Cp_MM	0.905	18.26
PB|PB_Cp_Exp	0.906	18.13
PB|PB_Eea_MM	0.912	17.49
PB|PB_Eea_Exp	0.905	18.24
PB|PB_Td|PB_Cp|PB_Eea (MM)	0.913	17.43
PB|PB_Td|PB_Cp|PB_Eea (Exp)	0.913	17.41

[a] Mean and Std of the test set: 135 and 59 °C.

We also tested models incorporating more than two embeddings, combining PolyBERT with all fine-tuned embeddings on properties other than T_g. This expanded the architecture from two to four branches, improving performance. However, it still did not outperform the model using the

T_g-fine-tuned embedding. Given the additional computational cost, this approach may not be the most efficient for T_g prediction.

6. DISCUSSION

We have presented a dual-embedding method which leverages two embeddings: a standard language model embedding trained on polymer SMILES, and a fine-tuned version where vectors are organized according to the proximity of T_g values. Our results demonstrate consistent improvements across heterogeneous polymer data sets, as well as for specialized families such as polyimides and homopolymers, with reductions in RMSE of up to 20% in the best case. This highlights the potential of the proposed property-aware fine-tuning.

The effectiveness of the dual-embedding strategy lies in the complementarity of the information captured by each branch. While the structural embedding preserves global molecular information, the fine-tuned embedding emphasizes property-relevant dimensions. Their combination yields more accurate predictions than either embedding alone.

Furthermore, extending this framework to other physicochemical properties shows that the model has the potential to improve prediction, as observed for the degradation temperature (T_d), although performance stagnated for the heat capacity (C_p) and electron affinity (E_{ea}). A complete study similar to the one carried out for T_g, along with experiments on multiple data sets, would provide a more global understanding and conclusion.

6.1. Limitations. While the performance gain may appear modest in some cases, it is significant in the context of polymer property prediction, where even small improvements are valuable. Such accuracy gains are particularly important for generative polymer design with reinforcement learning, where the calibration of reward functions depends critically on precise property prediction.[28]

The dual-embedding framework increases computational cost compared to single-embedding models, as fine-tuning must be performed separately for each property. This additional complexity remains manageable, but it highlights a trade-off between performance and efficiency. Moreover, generalization to very small data sets remains challenging, since a portion of the training data must be used for fine-tuning. Indeed, we cannot guarantee the effectiveness of the fine-tuning procedure with fewer than 200 polymer samples; therefore, a specific analysis would be required.

6.2. Future Directions. Future work could focus on integrating explicit physicochemical descriptors—such as chain rigidity, cohesive energy density, or bonding energy—into the learning process to better capture the molecular mechanisms governing T_g. Incorporating uncertainty quantification would also allow the model to account for experimental variability in polymer property measurements. Moreover, extending the approach to multitask or multiproperty training (e.g., T_g, T_d, C_p) could yield a unified latent representation linking structure and thermodynamic behavior. Finally, improving model interpretability by associating latent dimensions with structural motifs and validating predicted trends through new experiments represents a promising step forward.

7. CONCLUSIONS

In this work, we introduced a dual-embedding framework that combines a standard polymer embedding with a fine-tuned counterpart aligned with T_g similarity. This property-aware representation consistently outperforms conventional embeddings across heterogeneous polymer data sets as well as for specialized families such as polyimides and homopolymers, achieving substantial reductions in prediction error and demonstrating the benefits of embedding physicochemical information directly into the latent space.

Beyond improving property prediction, our approach opens promising avenues for materials discovery. Embedding spaces fine-tuned on specific properties could provide a foundation for inverse design, enabling the generation of novel polymer structures tailored to target properties. Multiproperty fine-tuning strategies may further enhance flexibility, while coupling dual embeddings with generative models, such as transformer-based SMILES generators,[29] could pave the way toward fully automated polymer design systems guided by property specifications.

■ ASSOCIATED CONTENT

Data Availability Statement

The code and data used in this article are available on *GitHub* (https://github.com/AMETHYST-2025/Dual_Embedding).

■ AUTHOR INFORMATION

Corresponding Author

Aymar Tchagoue − *INSA Lyon, UCBL, CNRS, LIRIS UMR 5205, 69100 Villeurbanne, France; INSA Lyon, UCBL, Université Jean Monnet, CNRS, IMP UMR 5223, 69100 Villeurbanne, France;* orcid.org/0009-0002-0250-2330; Email: aymar-lebeau.tchagoue-tchagoue@insa-lyon.fr

Authors

Véronique Eglin − *INSA Lyon, UCBL, CNRS, LIRIS UMR 5205, 69100 Villeurbanne, France*

Jean-Marc Petit − *INSA Lyon, UCBL, CNRS, LIRIS UMR 5205, 69100 Villeurbanne, France*

Sébastien Pruvost − *INSA Lyon, UCBL, Université Jean Monnet, CNRS, IMP UMR 5223, 69100 Villeurbanne, France;* orcid.org/0000-0002-3270-5668

Jannick Duchet-Rumeau − *INSA Lyon, UCBL, Université Jean Monnet, CNRS, IMP UMR 5223, 69100 Villeurbanne, France*

Jean-François Gérard − *INSA Lyon, UCBL, Université Jean Monnet, CNRS, IMP UMR 5223, 69100 Villeurbanne, France;* orcid.org/0000-0002-3096-2767

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.jcim.5c02469

Author Contributions

This work was made possible through a close collaboration between the LIRIS and IMP research teams. The LIRIS team (A.T., V.E., and J.-M.P.) contributed to the design of the machine learning methodologies and the analysis of model behavior. The IMP team (A.T., S.P., J.D.-R., and J.-F.G.) provided expertise in polymer science and contributed to the interpretation of the predicted results. All authors participated in the manuscript preparation and provided valuable feedback throughout the study.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work benefited from the state aid managed by the *National Research Agency* under the *France 2030* program, bearing the reference *ANR-22-PEXD-0004*.[30]

■ REFERENCES

(1) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Model.* **1988**, *28*, 31−36.

(2) Kuenneth, C.; Ramprasad, R. polyBERT: a chemical language model to enable fully machine-driven ultrafast polymer informatics. *Nat. Commun.* **2023**, *14*, 4099.

(3) Andraju, N.; Curtzwiler, G. W.; Ji, Y.; Kozliak, E.; Ranganathan, P. Machine-Learning-Based Predictions of Polymer and Postconsumer Recycled Polymer Properties: A Comprehensive Review. *ACS Appl. Mater. Interfaces* **2020**, *14*, 42771−42790.

(4) Miccio, L. A.; Schwartz, G. A. From chemical structure to quantitative polymer properties prediction through convolutional neural networks. *Polymer* **2020**, *202*, No. 122341.

(5) Tran, H. D.; Kim, C.; Chen, L.; Chandrasekaran, A.; Batra, R.; Venkatram, S.; Kamal, D.; Lightstone, J. P.; Gurnani, R.; Shetty, P.; Ramprasad, M.; Laws, J.; Shelton, M.; Ramprasad, R. Machine-learning predictions of polymer properties with Polymer Genome featured. *J. Appl. Phys.* **2020**, *128*, 171104.

(6) Qiu, H.; Liu, L.; Qiu, X.; Dai, X.; Ji, X.; Sun, Z.-Y. PolyNC: a natural and chemical language model for the prediction of unified polymer properties. *Chemical Science* **2024**, *15*, 534−544.

(7) Goswami, S.; Ghosh, R.; Neog, A.; Das, B. Deep learning based approach for prediction of glass transition temperature in polymers. *Mater. Today: Proc.* **2021**, *46*, 5838−5843.

(8) Volgin, I. V.; Batyr, P. A.; Matseevich, A. V.; Dobrovskiy, A. Y.; Andreeva, M. V.; Nazarychev, V. M.; Larin, S. V.; Goikhman, M. Y.; Vizilter, Y. V.; Askadskii, A. A.; Lyulin, S. V. Machine Learning with Enormous "Synthetic" Data Sets: Predicting Glass Transition Temperature of Polyimides Using Graph Convolutional Neural Networks. *ACS Omega* **2022**, *7*, 43678−43691.

(9) Casanola-Martin, G. M.; Karuth, A.; Pham-The, H.; González-Díaz, H.; Webster, D. C.; Rasulev, B. Machine learning analysis of a large set of homopolymers to predict glass transition temperatures. *Commun. Chem.* **2024**, *7*, 226.

(10) Tao, L.; Varshney, V.; Li, Y. Benchmarking Machine Learning Models for Polymer Informatics: An Example of Glass Transition Temperature. *J. Chem. Inf. Model.* **2021**, *61*, 5395−5413.

(11) Xu, C.; Wang, Y.; Farimani, A. B. TransPolymer: a Transformer-based language model for polymer property predictions. *npj Comput. Mater.* **2023**, *9*, 105.

(12) Wang, S.; Guo, Y.; Wang, Y.; Sun, H.; Huang, J. SMILES-BERT: Large Scale Unsupervised Pre-Training for Molecular Property Prediction. In *BCB '19: Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 2019.

(13) Chithrananda, S.; Grand, G.; Ramsundar, B. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv* **2020**,

(14) Landrum, G. *RDKit: Open-source cheminformatics.* 2013, https://www.rdkit.org/.

(15) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754.

(16) Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems (NeurIPS)*, **2013**, p. 26.

(17) Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *J. Chem. Inf. Model.* **2018**, *58*, 27−35.

(18) Goh, G. B.; Hodas, N. O.; Siegel, C.; Vishnu, A. SMILES2Vec: An Interpretable General-Purpose Deep Neural Network for Predicting Chemical Properties. *arXiv* **2017**.

(19) Zheng, X.; Tomiura, Y. A BERT-based pretraining model for extracting molecular structural information from a SMILES sequence. *J. Cheminform.* **2024**, *16*, 71.

(20) He, P.; Liu, X.; Gao, J.; Chen, W. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *arXiv* **2020**,

(21) Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *arXiv* 2019, .

(22) Xu, W.-S.; Douglas, J. F.; Sun, Z.-Y. Polymer Glass Formation: Role of Activation Free Energy, Configurational Entropy, and Collective Motion. *Macromolecules* **2021**, *54*, 3001−3016.

(23) Rollins, Z. A.; Cheng, A. C.; Metwally, E. MolPROP: Molecular Property Prediction with Multimodal Language and Graph Fusion. *J. Cheminform.* **2024**, *16*, 56.

(24) Wu, Y.; Ni, X.; Wang, Z.; Feng, W. Enhancing Drug Property Prediction with Dual-Channel Transfer Learning Based on Molecular Fragments. *BMC Bioinform.* **2023**, *24*, 293.

(25) Luu, R. K.; Wysokowski, M.; Buehler, M. J. Generative Discovery of Novel Chemical Designs using Diffusion Modeling and Transformer Deep Neural Networks with Application to Deep Eutectic Solvents. *Appl. Phys. Lett.* **2023**, *122*, 234103.

(26) Nandan Thakur, J. D.; Reimers, N.; Gurevych, I. Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks. *arXiv* 2021, .

(27) Morgan, H.L. The generation of a unique machine description for chemical structures—a technique developed at chemical abstracts service. *J. Chem. Doc.* **1965**, *5*, 107−113.

(28) Yue, T.; Tao, L.; Varshney, V.; Li, Y. Benchmarking study of deep generative models for inverse polymer design. *Digital Discovery* **2025**, *4*, 910−926.

(29) Qiu, H.; Sun, Z.-Y. On-demand reverse design of polymers with PolyTAO. *npj Comput. Mater.* **2024**, *10*, 273.

(30) *Projet AMETHYST, France 2030 PEPR DIADEM.* https://www.pepr-diadem.fr/projet/amethyst/.