2022

Machine Learning for the Prediction of Polymer Glass Transition Temperature

Projet 7 Parcours Ingénieur Machine Learning

OPENCLASSROOMS

Alfred Bazin

Mentor: **Amine Hadj-Youcef**

Introduction

Machine Learning applied to Polymer Science

As it is often observed in literature, Machine Learning (ML) can be applied to a wide variety of domains. As an example, because of its high adaptability, deep learning has dramatically changed the domain of image processing for objects recognition. ML has also been applied to other data types as text and sound for language processing tasks.

A domain that could take advantage of ML is chemistry (see Figure 1) [1]. *In-silico* chemistry has been developed for modeling molecules properties and even interactions. These simulations often (if not always) rely on calculation issued from quantum mechanics and based on the Density Functional Theory (DFT). However, such calculations are costly in processing time, requires expensive software and high knowledge in chemoinformatics as well as in quantum mechanics.

Because of this complexity, and since this method still does not always provide perfect results, most of the chemical research is today based on experimentation and empirical reasoning (for properties or reactivity fine-tuning as an example).

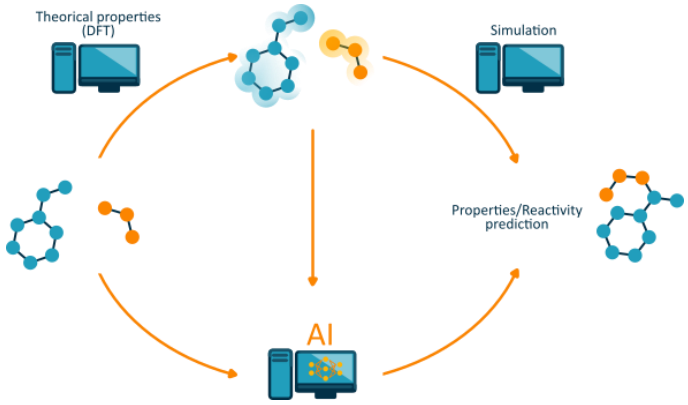

Figure 1 : Illustration of the tools used for *in silico* chemistry (theoretical simulation *versus* the use of machine learning).

As it can be expected, the difficulty for performing DFT calculations and simulations increases with the molecule size. Polymers are large molecules that could be compared to a chain where the links are molecules attached together (see Figure 2a). The number of links in a polymer chain can range from around 10 to multiple thousands. Polymers are the main constituents of all plastic materials and are widely present in nature. The most famous example of natural polymer in probably DNA (see Figure 2b), but others are common as starch or cellulose (the main constituent of cotton wool).

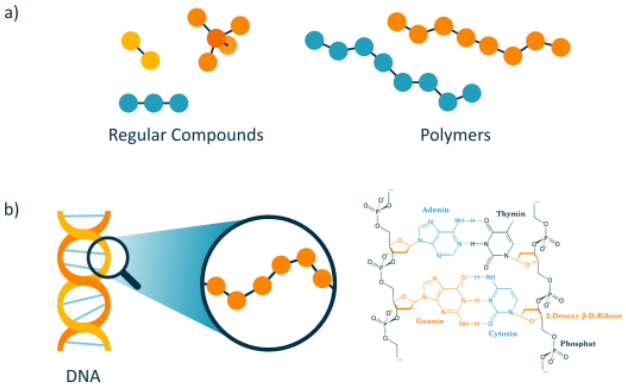

Figure 2 : a) Illustration of polymers compared with common compounds. b) Illustration of the DNA as a polymer

Because of their atypical structure, the modeling of the interaction of polymer molecules together can be difficult. Moreover, these materials possess highly specific properties. Where common chemical compounds often present a melting and boiling temperature, polymers can undergo a specific thermal transition called glass transition temperature (T_g). The T_g corresponds to a point where the polymeric chains can slide together, often transforming the material from hard and brittle (called the glassy state) to softer and more resilient (called the rubbery state). Controlling the T_g of a polymer-based material is critical for producing materials with optimal performances. As an example, a material with a T_g below room temperature could be too soft for everyday uses and *vice versa*.

The Prediction of Polymer T_g

The modeling of polymer properties has already been studied from a long time ago. Theories based as an example on the chains rotation angles and free volumes have achieved to partially explain the Tg of the most common polymers[2]. However, a lot of parameters as the shape, the polarity, chain length and so on can affect the T_g.

The prediction of polymer properties with ML models have been studied in the literature[3], [4]. To do so, multiple input data and models have been considered, including calculated descriptors, structural fingerprints[5]. The input data can be separated into two main categories: the simulation-based parameters (sometimes based on DFT calculations) and the input only based on structural parameters (as the position of the atoms in the space).

Multiple model types have also been studied as Support Vector Machine (SVM), Gaussian Process Regression (GPR), Random Forest[6], [7]. Deep neural networks have also been employed for regression[7]. More surprisingly, language processing methods as long short-term memory networks have also been studied and yielded interesting result[8].

In this project, two strategies have been compared. First using a model developed by Y. Zhang et al.[6] in 2020 and relying on DFT calculations. Then, the second part of the project focuses on a benchmark study realized by L. Tao et al. [7] in 2021. This time, only structural parameters have been employed and the two best-performing models have been assessed. In this part, a new dataset that has not been used by the authors have been used in order to evaluate the models.

Finally, the model development in the last part has been compared and applied to atypical polymers that I studied during my Ph.D.

DFT-based models

The first model was based on the study produced by Y. Zhang et al.[6]. The authors used as input features parameters obtained through DFT calculations. As described earlier, DFT calculations can be difficult and time consuming to obtain. However, such features are closely related to the molecules physical and chemical properties. The authors considered the molecular quadrupole moment and hexadecapole moment as features. These features were directly used for T_g prediction without, according to the authors, further feature engineering.

The Dataset

As depicted earlier, DFT calculation requires expensive proprietary software. Thus, new data couldn't be provided to this project and the same dataset as the authors was used for the model evaluation.

The dataset is composed of 60 samples with two numerical predictive features, the polymer names and a numerical target. The first 5 rows of the dataset are represented in Table 1.

Table 1: DFT-based dataset

Index	Polymer	Θ (Debye Ang)	Φ (Debye Ang3)	T_g (K)
1	Polyethylene	0.6776	-50.0178	195
2	Poly (vinyl acetate)	7.7949	-324.9730	301
3	Poly (vinyl butyral)	7.3902	-889.1739	324
4	Poly (vinylidene fluoride)	3.8068	-98.8344	233
5	Poly (vinylidene chloride)	5.0194	-223.2520	256

As only two features and one target are considered, the dataset can be represented in a 3D graph (see Figure 3).

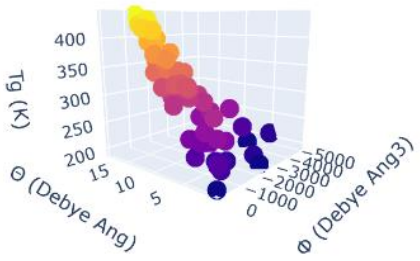

Figure 3: 3D plot representing the DFT-based dataset

The different models

Gaussian Process Regressor (GPR)

The model studied by the authors in the original study is the GPR. This model is often used because of its flexibility (induced by its use of a variety of kernels). However, its high flexibility can also lead it to important overfitting. The authors used described the kernel used as the isotropic exponential kernel and linear basis without a clear description of its formula. Thus, the kernel considered in this project was the Radial Basis Function (RBF), a popular kernel set by default in the scikit-learn library.

As described in the original study, the hyper-parameters of the model were optimized through cross-validation (with the 10 folds described by the authors). The R^2 score obtained with the best model on the validation set during the cross-validation (and thus on each fold) is given in Table 2. The obtained values are compared with the values obtained by the authors.

Table 2: R^2 score obtained on the different folds in this study and by the authors.

Fold N°	This study	The authors
Fold 1	0.64	0,87
Fold 2	0.95	0,96
Fold 3	0.88	0,95
Fold 4	0.87	0,98
Fold 5	0.95	0,97
Fold 6	0.90	0,95
Fold 7	0.70	0,94
Fold 8	0.94	0,99
Fold 9	0.88	0,97
Fold 10	0.74	0,90

Moreover, the mean absolute error (MAE) (on average) has been calculated at **22.1** K. This value can be compared to the one obtained by the authors at **18.3** K. We can observe that the model produced high performances with a high R^2 score and a low MAE. However, these performances seem to be lower to the one obtained by the authors in the original study. This difference of performances could be attributed to the different type of kernel employed.

Ridge Regression

Now that the method employed in the original study has been (partially) reproduced, other architectures can be investigated. It feels intuitive that the data in Figure 3 could be organized as a plan. Thus, a classical linear model could in theory fit the data. In order to test this theory, a linear model has been applied to the data for prediction of the T_g. The Ridge regressor have been selected as it is a linear model which provides a control on a regulation parameter (l2) which could allow reducing overfitting of the model. In a second step, the influence of the addition of non-linear features have been assessed.

Then, the assessment of a model only based on cross-validation scores can be limiting and do not completely attest of the model generalization. Thus, the data have been split between a train and a test set for assessing the model performances.

After training, the model as well as the dataset are presented in a 3D plot in the Figure 4.

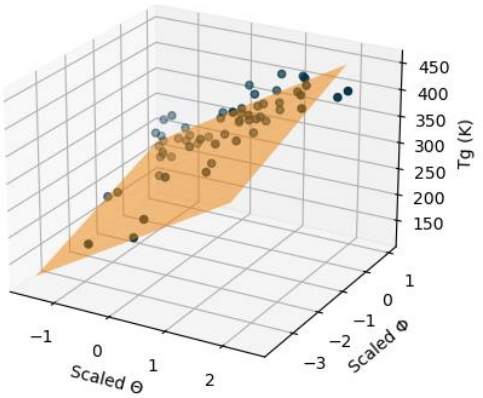

Figure 4: 3D plot representing the ridge model (as a plane) and the dataset (as a scatter plot)

A R^2 score **0.90** have been obtained both on the train and the test set. The MAE was calculated at **18.0** K for the train set and **16.4** K for the test set.

In conclusion, the features issued from DFT calculation were used for the prediction of T_g with various models. The model showing the best performances in the original paper was based on GPR. A similar model (but with a different kernel and optimization method) has been developed in this project and showed good performances (with an average test MAE of **22.1** K). However, the 3D representation of the data points suggests that the data could be fitted by a plane. Thus, a model of Ridge regression has been developed and showed even better prediction performances (with a test MAE of **16.4** K). These performances were even better than the values obtained by the authors for the best model (with an average test MAE of **18.0** K).

The use of non-linear features has also been assessed but resulted in not improvement of the model performances.

MFF-based models

The authors L. Tao et al. have recently published a study for benchmarking several input data, encoding methods and models for the prediction of polymers T_g [7]. A simple input type considered by the authors that resulted in impressive results is the SMILE code of the repetitive unit of the polymer.

Figure 5: Illustration of the conversion of molecular structures to SMILE code for two molecules.

The SMILE code is a code often used in chemoinformatics that allow carrying as a string information about a molecule structure (see Figure 5). In order to obtain more information about the molecules structure, the authors obtained from the SMILE a feature called the Morgan Fingerprint. This feature is also often employed in chemoinformatics and describe in more details the structure of a molecule by attributing values to each atom depending on its surrounding environment (more details can be found here or in the paper of L. Tao et al. [7]). The appearance of certain structure types (given as codes) is generally one hot encoded (OHE). In their study, the authors added the frequency of apparition of a given structure to the Morgan fingerprint (replacing the 1 of the one hot encoding by the number of apparitions of the structure in the given molecule). The process of encoding of a given polymer structure is summarized in Figure 6.

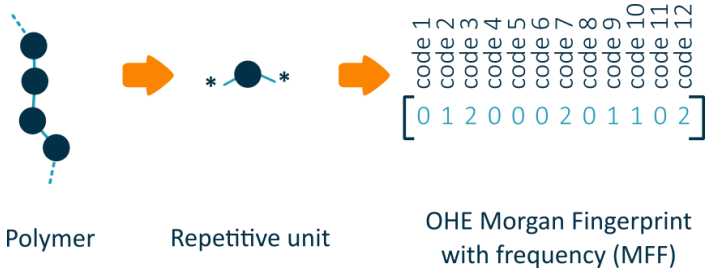

Figure 6: Illustration of the encoding of a given polymer structure to Morgan Fingerprint with frequency (MFF)

The dataset

The dataset that was used in this study contains 219 samples and was collected from a popular polymer database with the aid web scraping. The first 5 rows of the dataset are represented in Table 3.

Table 3: First five rows of the dataset containing SMILES Codes

Index	COMMON NAMES	SMILES	T_g
16330-21-5	Propyl vinyl thioether, 1-(Vinylsulfanyl)propane	*CC*(SCCC)	253.0
1822-73-7	Vinyl phenyl sulfide, Phenylthioethene, (Vinyl...	*CC*(Sc1ccccc1)	398.0
1822-74-8	Methyl vinyl thioether, (Methylthio)ethene	*CC*(SC)	270.0
627-50-9	Ethyl vinyl thioether, (Ethylthio)ethene, Ethy...	*CC*(SCC)	265.0
4789-70-2	Butyl vinyl thioether	*CC*(SCCCC)	254.0

The SMILES code considered in this study, similarly to the dataset used by L. Tao et al. [7], have particularity. A star ('*') indicates the beginning and the end of the repetitive unit (as the moieties are included in a larger chain).

The molecule structures have been encoded from SMILES codes to Morgan fingerprint as described by the authors. The code used in this part has been published by the author on GitHub. The authors indicated that the 98 most common structures were selected as input for the ML models (which corresponds to features in the OHE). Unfortunately, the structures that were selected were not indicated and this information could not be retrieved in the data provided by the authors. Thus, it is impossible to have data with the same structure as the original study. For this reason, the models cannot be used as provided by the authors and part of the model layers must me retrained. The first three rows of the dataset after OHE of the Morgan Fingerprints with Frequency (MFF) is represented in Table 4.

Table 4: first Three rows of the MFF-based dataset

Index	40083374	30993531	13074597	71751290	34578844	14240815	18626320	86467448	23479672	13528442	...	16196312	84796121	39999069	19178202	29947487	14697426	41217553	32186939	25161972	target
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	253.0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	2	0	0	0	0	5	0	398.0
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	270.0

Finally, the dataset has been shuffled and split between a training and a validation dataset in the proportion of 8/2.

The different models

Neural Network (NN)

The first model to be assessed is a NN with an input layer of size 94, two hidden layers containing 8 neurons and an output layer of one neuron (for regression). Each layer was also attributed the activation function 'ReLU'. The structure of the model is depicted in .

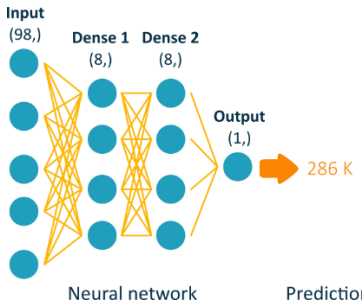

Figure 7: Schematic Structure of the NN Model

As depicted earlier, the model has been retrained on the dataset. The performance of the model along the epochs was evaluated on the validation set with the MAE as metric. The training history was recorded with tensorboard and are represented in Figure 8.

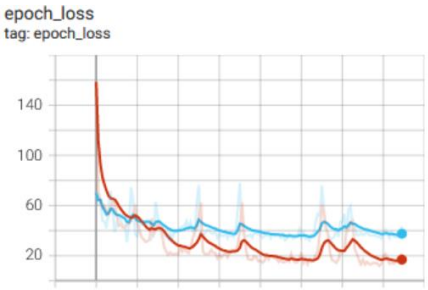

Figure 8: Training History of the NN

The model was the evaluated. The validation MAE has been measured at **24.3** K for an R^2 of **0.83** while the training MAE is **19.1** for an R^2 measured at **0.89**. First, these values indicate that the model presents good generalization. Then these scores are similar to the scores obtained in the original study of L. Tao et al. [7]. Indeed, with an identical model architecture, the authors obtained a MAE of **31.8** on the train set with an R^2 of **0.84** and a MAE of **34.3** with an R^2 of **0.81** on a test set.

However, three elements do not allow us to fully compare the results obtained in this project with the values obtained by the authors. First, as explained earlier, the input data structure is not similar to the one the authors provided. Then, the amount of data used in this project is thirty times lower when compared to the one used in the original study. This difference of sample number could explain the difference in R^2 score. Finally, the authors evaluated the model performances on a test set (unseen by the model), while our model was evaluated on a validation dataset (due to the low amount of data available) that was used for selecting the best model.

The Random Forest (RF)

Another model that presented good performance in the study of L. Tao et al. is the Random Forest (RF). The RF is an ensemble method that reduces overfitting of the model by combining the output of multiple regression trees. An illustration of the structure of a random forest is given in Figure 9. Again, the exact model trained by the authors couldn't be used in our study because of the difference of data structure. Thus, a new RF has been trained. However, the model hyper-parameters were identical to the ones set by the authors.

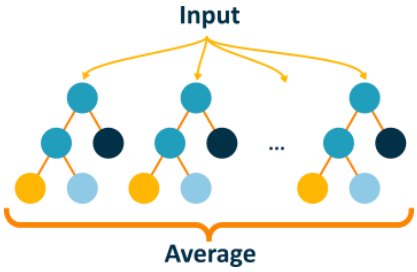

Figure 9: Illustration of a RF Model

After evaluation, the validation MAE has been measured at **27.7** for an R^2 of **0.78** while the training MAE was **13.6** for an R^2 measured at **0.95**. This time, the value indicates that the model present overfitting of the training data. Lowering the model max depth or using other ensemble methods as AdaBoost or XGBoost did not allow reducing the model overfitting.

These scores are again similar to the scores obtained in the original study of L. Tao et al.[7]. Indeed, with an identical model architecture, the authors obtained a MAE of **23.2** on the train set with an R^2 of **0.91** and a MAE of **31.0** with an R^2 of **0.85** on a test set. Again, as explained earlier, the input data are not and the difference of R^2 could be explained by the difference in the number of samples considered.

Challenging the Models With Atypical Datapoints

The MFF-based polymers could be applied to more peculiar polymers as the MFF is easy to calculate for a given polymer structure. Both the neural network and the RF have been challenged with polymers issued from my Ph.D. thesis for which I calculated the MFF and measured the T_g. The structure of the repetitive units is given in Figure 10.

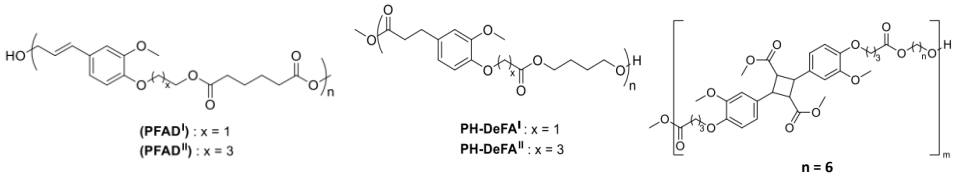

Figure 10: Structure of the five atypical polymers chosen for challenging the MFF-based models

The prediction of the model *versus* the true values are given in Figure 11 for both the NN and the RF. The newly added polymers are represented in orange.

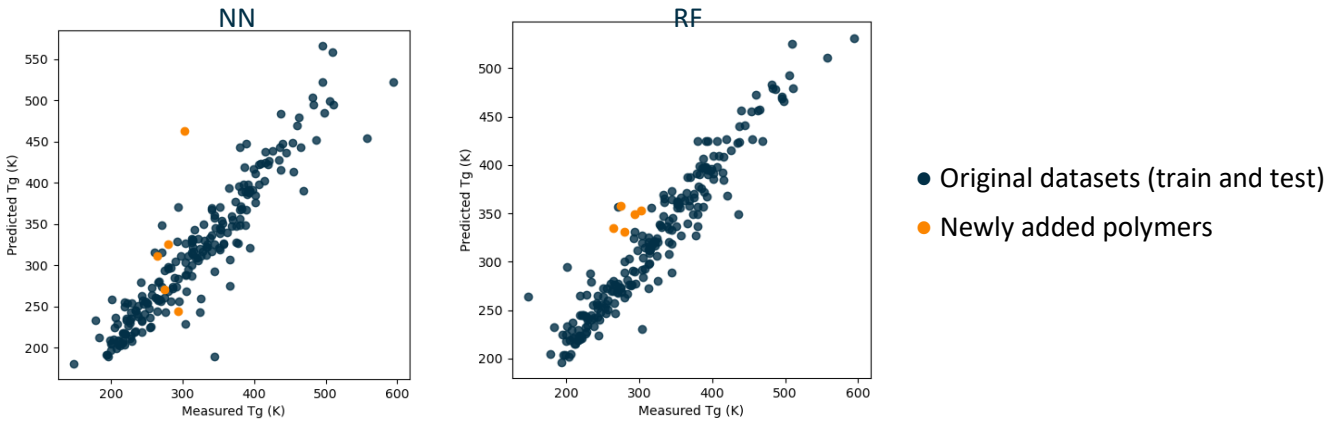

Figure 11: Experimentally measure T_g *versus* predicted T_g for the NN and RF models

Concerning the prediction of the NN, we can see that one polymer in particular show a high deviation from the diagonal suggesting a low prediction accuracy. This polymer corresponds to the one on the far right in Figure 10 and could be interpreted as the one with the most peculiar structure, possibly explaining the difficulty of the model to predict it T_g.

When studying the prediction of the RF model, all the predictions for the newly added polymer are around 50 K above the actual measured value. One phenomenon that could explain this shift is that the newly added polymer present low number of repetitive units when compared with other common polymers. This low molecular weight could lead to a reduced experimental T_g. Indeed, all the models presented in this study did not consider the length of the polymer chains even though it is a parameter that highly influence the T_g.

Conclusion

In this study, the features issued from DFT calculation were used for the prediction of T_g with various models. As the features calculation requires high processing time, the data already calculated by the authors have been used in this project. The model showing the best performances in the original paper was based on GPR. A similar model (but with a different kernel and optimization method) has been developed in this project and also showed good performances (with an average test MAE of **22.1** K). However, the 3D representation of the data points suggests that the data could be fitted by a plane. Thus, a model of Ridge regression has been developed and showed even better prediction performances (with a test MAE of **16.4** K). These performances were even better than the values obtained by the authors for the best model (with an average test MAE of **18.2** K).

Thus, simple models as the Ridge regression are able to predict with good performances the T_g of polymers based on DFT-calculates parameters. However, as explained earlier, these features require high skills and processing times. For this reason, authors have looked for other simpler input types that could be considered for the prediction of polymers T_g. One example is the SMILE code that provides a lot of structural information.

From the SMILE codes, the MFF of various polymers have been calculated. This feature has been used for the prediction of the polymers T_g. The performances of two models have been compared: A NN and a RF. The NN showed good performances even though a low amount of data was available for training. The random forest showed even better results but with an overfitting of the train dataset. The authors indicated that using other models as AdaBoost do not allow correcting this tendency to overfit. It was confirmed in this study by assessing the performances of AdaBoost and XGBoost regressors.

References

[1] R. Gómez-Bombarelli *et al.*, « Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules », *ACS Cent. Sci.*, vol. 4, n° 2, p. 268-276, févr. 2018, doi: 10.1021/acscentsci.7b00572.

[2] V. I. Irzhak, G. V. Korolev, et M. E. Solov'ev, « Intermolecular interaction in polymers and the physical network model », *Russ. Chem. Rev.*, vol. 66, n° 2, p. 167-186, févr. 1997, doi: 10.1070/RC1997v066n02ABEH000256.

[3] L. Chen *et al.*, « Polymer Informatics: Current Status and Critical Next Steps », *Mater. Sci. Eng. R Rep.*, vol. 144, p. 100595, avr. 2021, doi: 10.1016/j.mser.2020.100595.

[4] W. Sha *et al.*, « Machine learning in polymer informatics », p. 9, 2020.

[5] R. A. Patel, C. H. Borca, et M. A. Webb, « Featurization Strategies for Polymer Sequence or Composition Design by Machine Learning », p. 25.

[6] Y. Zhang et X. Xu, « Machine learning glass transition temperature of polymers », *Heliyon*, vol. 6, n° 10, p. e05055, oct. 2020, doi: 10.1016/j.heliyon.2020.e05055.

[7] L. Tao, V. Varshney, et Y. Li, « Benchmarking Machine Learning Models for Polymer Informatics: An Example of Glass Transition Temperature », *J. Chem. Inf. Model.*, vol. 61, n° 11, p. 5395-5413, nov. 2021, doi: 10.1021/acs.jcim.1c01031.

[8] G. Chen, L. Tao, et Y. Li, « Predicting Polymers' Glass Transition Temperature by a Chemical Language Processing Model », *Polymers*, vol. 13, n° 11, p. 1898, juin 2021, doi: 10.3390/polym13111898.