A Physics-Enforced Neural Network to Predict Polymer Melt Viscosity

Ayush Jain
Department of Materials Science and Engineering
Computational Science and Engineering
Georgia Institute of Technology
ayush.jain@gatech.edu

Rishi Gurnani
Department of Materials Science and Engineering
Georgia Institute of Technology

Arunkumar Rajan
Department of Materials Science and Engineering
Georgia Institute of Technology

H.Jerry Qi
Department of Mechanical Engineering
Georgia Institute of Technology

Rampi Ramprasad
Department of Materials Science and Engineering
Georgia Institute of Technology
rampi.ramprasad@mse.gatech.edu

Abstract

Achieving superior polymeric components through additive manufacturing (AM) relies on precise control of rheology. One key rheological property particularly relevant to AM is melt viscosity (η). Melt viscosity is influenced by polymer chemistry, molecular weight (M_w), polydispersity, induced shear rate ($\dot{\gamma}$), and processing temperature (T). The relationship of η with M_w, $\dot{\gamma}$, and T may be captured by parameterized equations. Several physical experiments are required to fit the parameters, so predicting η of a new polymer material in unexplored physical domains is a laborious process. Here, we develop a Physics-Enforced Neural Network (PENN) model that predicts the empirical parameters and encodes the aforementioned equations to calculate η as a function of polymer chemistry, M_w, polydispersity, $\dot{\gamma}$, and T. We benchmark our PENN against physics-unaware Artificial Neural Network (ANN) and Gaussian Process Regression (GPR) models. Finally, we demonstrate that the PENN offers superior values of η when extrapolating to unseen values of M_w, $\dot{\gamma}$, and T for sparsely seen polymers.

1 Introduction

Additive Manufacturing (AM) enables the rapid creation of metal or polymer parts with previously-unimaginable features and topologies, and is poised to disrupt industries [1, 2]. Achieving desired properties in the final component is determined by the appropriate choices of material chemistries with suitable rheological properties, as well as conditions adopted during the AM process such as temperature, extrusion rates, etc. At present, a limited palette of chemistries, properties, and conditions is utilized, generally guided by experience, intuition, and empiricism.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

In this contribution, we adopt an informatics approach relevant to AM across the chemical and process condition space, to predict one critical rheological property of polymers, namely, the melt viscosity η. Informatics approaches have made major inroads in recent years within materials research [3, 4, 5], leading to accelerated means for property predictions and providing guidance for the design of new materials [6, 7, 8, 9, 10]. These methods start with available materials data on properties of interest. The materials are then represented numerically to capture and encode their essential features in a machine-readable format. The numerical representations, or fingerprints, are then mapped to available property data using machine learning (ML) algorithms, ultimately producing predictive models for the property considered [6, 11, 12, 13, 14, 15]. Within the AM space, similar methods have been used for process monitoring [16] and optimization of printing parameters (albeit mainly for powder-bed AM [17, 1], but not as much for polymer melt extrusion AM [2]). Extrusion AM relies on the precise control of polymer melts, which currently requires data from extensive rheological experiments for each new chemistry. This is a bottleneck in the ink development process [2]. Therefore, predictive capabilities for rheological properties, such as η, are useful to reduce the number of physical experiments aimed at optimization and design.

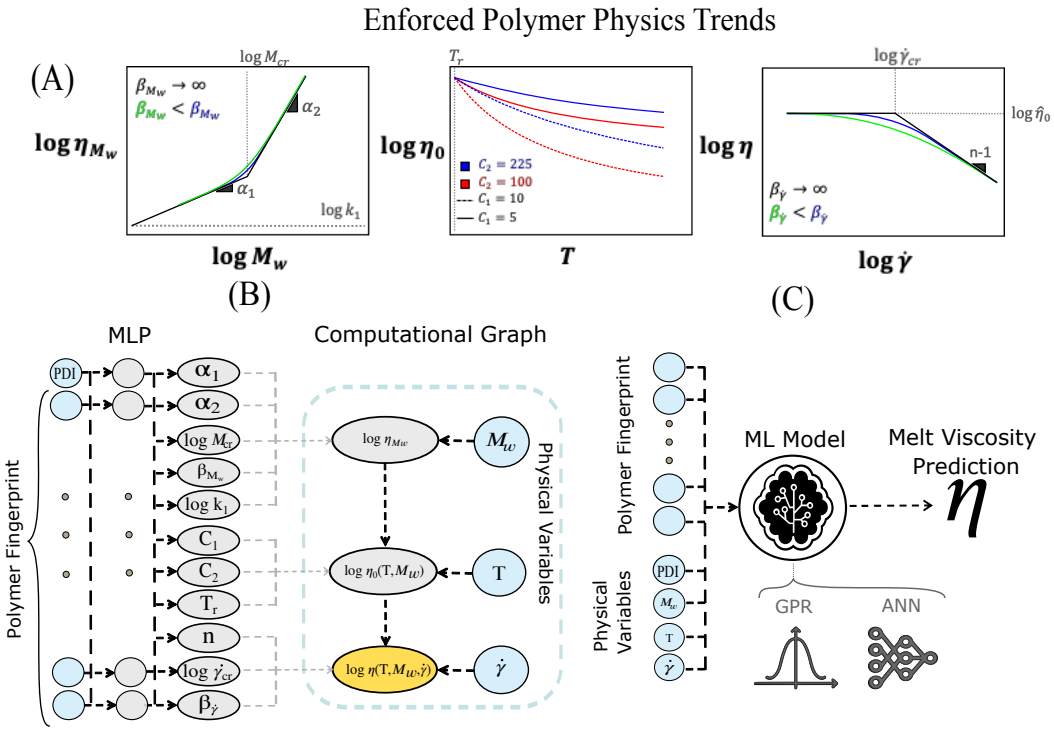

Figure 1: The melt viscosity (η) learning problem and machine-learning workflow. (A) Depictions of the functions used to describe the behavior of η with respect to temperature (T), molecular weight (M_w), and shear rate ($\dot{\gamma}$). The functions are parametrized by empirical parameters with physical significance, elaborated in Table 1 and in Appendix A. (B) The Physics-Enforced Neural Network (PENN) architecture starts with an input containing the polymer fingerprint and the PDI. A Multi-Layer Perceptron (MLP) uses the concatenated input to predict the empirical parameters. Next, the computational graph uses the predicted empirical parameters to calculate η, via the encoded $\log \eta_{M_w}$, $\log \eta_0(T, M_w)$, and $\log \eta(T, M_w, \dot{\gamma})$ functions. The physical condition variables $\log M_w$, $\log \dot{\gamma}$ and T are input to their respective functions. (C) Physics unaware Artificial Neural Network (ANN) and a Gaussian Process Regression (GPR) are baselines to compare with the PENN model. The input features to the ANN and GPR models are the concatenated polymer fingerprint, T, M_w, $\dot{\gamma}$, and PDI.

Melt viscosity of polymers, beyond being a critical property, is attractive to model with ML because there is a reasonable amount of related literature data, although with limited chemical diversity compared to other polymer property datasets [12, 18, 5, 15, 19]. Additionally, there are known physical equations (albeit with empirical parameters) that describe the dependence of η on its governing conditions: temperature (T), weight average molecular weight (M_w), and shear rate ($\dot{\gamma}$) (Figure 1A). For instance, it is known that the melt viscosity increases with increasing M_w (via piece-wise power

law dependencies), decreases (non-linearly) with increasing $\dot{\gamma}$, and decreases (exponentially) with increasing T. Explicit functional forms and additional background on the behaviors are provided in Appendix A. Molecular weight distributions, quantified by the polydispersity index (PDI), are also known to affect melt viscosity [20, 21, 22]. With this situation in mind, previous works have also addressed the modeling of η using ML [23, 24, 25, 26]. While promising, the majority of these works have focused on specific scenarios or are shown to predict unphysical results [26], making them difficult to apply. For example, [23] uses coarse-grained molecular dynamics simulation data of polymer nanocomposites with ML frameworks to predict η, and validated the results with empirical equations, but this hasn't been applied to modeling across chemistries.

Additionally, physics-informed ML frameworks have shown great promise recently to apply ML models to physical and real-world applications, including atomic modeling, chemistry-informed materials property prediction, and solving partial differential equations that describe physical phenomena [27, 28, 29, 30, 31]. A notable framework is the Physics Informed Neural Network (PINN) in which a neural network is considered to be the solution to a boundary value problem, and initial and boundary conditions comprise the loss function such that the neural network fits the data and the underlying principles within the equations [31]. However, the PINN approach assumes that boundary values or constraints remains the same for all inputs. In the case of η and other materials informatics problems, the governing equations' parameters depend on the input polymer chemistry. If a PINN were to be applied to this problem, the loss surface would drastically change for different polymer chemistries, making optimization difficult.

Given this, we approach this problem with a physics-enforced neural network (PENN) framework for η which explicitly encodes the known physical equations while also learning the empirical parameters for new chemistries directly from available data. Our PENN for polymer melt viscosity prediction involves a Multi-Layer Perceptron (MLP) that takes as input the polymer chemistry (fingerprinted using our Polymer Genome approach [6]) along with the PDI of the sample, and predicts the empirical parameters as a latent vector (listed in Table 1) used to estimate η as a function of T, M_w, and $\dot{\gamma}$. A computational graph then encodes the dependence of M_w, $\dot{\gamma}$, and T on η (see Figure 1A) using the equations described in Appendix A. Smooth transition functions are used to combine the different regimes observed across M_w and $\dot{\gamma}$, ensuring that the PENN captures the full physical behavior. By combining the dependencies on shear rate, temperature, and molecular weight, the model provides a comprehensive expression for $\eta(M_w, T, \dot{\gamma})$. The entire framework (Figure 1B) is trained on our dataset.

Parameter	Physical Representation	Relevant Equation(s)
C_1	Empirical Parameter for the η - T relationship	5
C_2	Empirical Parameter for the η - T relationship	5
T_r	Reference temperature for the η - T	5
M_{cr}	Critical M_w, associated with the onset of polymer chain entanglement	6,7,8
α_1	Slope of zero-shear viscosity (η_0) vs. M_w when $M_w < M_{cr}$ (approximately 1)	6,7,8
α_2	Slope of η_0 vs. M_w when $M_w > M_{cr}$ (approximately 3.4)	6,7,8
β_{M_w}	Measure of transition from α_1 to α_2 at M_{cr}	8
k_1	η_0 when $M = 0$ and $T = T_r$	6,7,8
$\dot{\gamma}_{cr}$	Critical Shear Rate when $T = T_r$, associated with the onset of shear-thinning	2,3,4
n	Slope of shear thinning, typically has a value of (0.2-0.8 for polymer melts)	2,3,4
$\beta_{\dot{\gamma}}$	Measure of transition between the zero-shear and shear-thinning regions	4

Table 1: Definitions of empirical parameters predicted by the Physics Enforced Neural Network (PENN) and their relevance to temperature (T), molecular weight (M_w), and shear rate ($\dot{\gamma}$) when calculating melt viscosity (η). The relevant equations in the Methods section are provided for each parameter.

We find that this strategy is critical to obtain results that are physically meaningful in extrapolative regimes (e.g., ranges of T, M_w and $\dot{\gamma}$ where there is no training data for chemistries similar to the queried new polymer). This ability is vital given our benchmarking dataset's sparsity, containing only 93 unique repeat units, although the total number of datapoints is 1903 (including T, M_w, $\dot{\gamma}$, and composition variations. As baselines to assess this PENN, we trained artificial neural network (ANN) and Gaussian process regression (GPR) models without any physics encoded. We find that the PENN model is more useful in obtaining credible extrapolative predictions. Our results indicate that

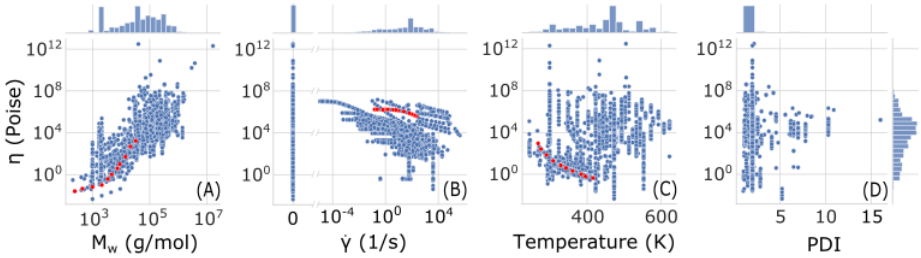

Figure 2: The joint distributions of A) molecular weight (M_w), B) shear rate ($\dot{\gamma}$), C) temperature (T), and D) polydispersity index (PDI) with respect to melt viscosity (η) are presented. The single distributions for the physical conditions are given on the top axes and the distribution of η is given on the right-most axis. Each subplot contains all 1903 datapoints from the dataset. A, B, and C have highlighted samples in red that exemplify the dependencies depicted in Figure 1A.

informatics-based data-driven and physics-enforced (when possible) strategies can aid and accelerate extrusion AM innovations in sparse data situations.

2 Results and Discussion

2.1 Polymer Fingerprinting and Representation

The input to the ML models consists of a vector to represent the chemical attributes of a polymer material (Figure 1B and C). The vector, known as a fingerprint (FP) contains features derived from atomic-level, block-level, chain-level, and morphological descriptors of a polymer as described at length earlier[6]. Thus, the FP features vary across the polymer chemistries in the dataset to help build chemical intuition. The dataset contains homo- and co-polymers, and miscible polymer blends. Co-polymers and blends contain multiple repeating units, each with a separate FP. For co-polymers, the FP of each unit was aggregated to a single copolymer FP using a weighted average (with weight equal to composition percentage) [12]. Similar to previous work[12], all co-polymers were treated as random. For miscible polymer blends, the FP of each unit was aggregated to a single FP using a weighted harmonic average (with the weight equal to composition percentage) [18]. For blends containing units with different M_w and/or PDI, the weighted average over each unit was used.

2.2 Dataset

We define a melt viscosity datapoint as having an input of FP for a polymer and its processing conditions (M_w, $\dot{\gamma}$, T, and PDI), with η being output. Melt viscosity data was collected from the PolyInfo repository [32] and from the literature cited by PolyInfo. Cited literature data was extracted from tables and figures with the help of the WebPlotDigitizer tool [33]. The final dataset shown in Figure 2 includes a total of 1903 datapoints composed of 1326 homopolymer datapoints, 446 co-polymer datapoints, and 113 miscible polymer blend datapoints. The dataset spans a total of 93 unique repeat units with variations in M_w, $\dot{\gamma}$, T, and PDI. For datapoints without a recorded PDI, we impute 2.06, the median PDI of the dataset.

We found that η at low M_w were underrepresented when compared to η measurements at high M_w. Using the zero-shear viscosity (η_0) relationship with M_w (Figure 1A), we added 126 datapoints at low M_w (included in the 1903 datapoints). This was achieved by identifying polymer chemistries with more than five η_0 datapoints at high M_w and a recorded M_{cr}[34]. Equation 6 (Appendix A) was fit to each chemistry and extrapolated to estimate η values at low M_w.

Because the viscosity values span several orders of magnitude (Figure 2), we use the Order of Magnitude Error (OME) to assess ML model accuracy. OME is calculated by taking the Mean Absolute Error of the logarithmically scaled η values. Models with lower OME exhibit more accurate predictions.

2.3 Overall Assessment of Physical Intuition with Sparse Chemical Knowledge

An important future use case of our ML models is to estimate the melt viscosity in new physical regimes, given a small amount of knowledge of a given polymer and other chemistries. For example, given a few costly tests of a new polymer at a few molecular weights, a scientist should be able to predict the viscosity at remaining molecular weights, and, likewise, across different shear rates and temperatures. This ability was tested through a unique splitting of data into test/train sets across the chemical and physical regimes. First, the monomers were split into train (90%) and test (10%) sets. Within the test monomers, the median of the distributions of the test monomers with respect to a variable in the physical space was calculated. The median was used to split all datapoints containing that monomer: half for a final test split, and the other half for training. The upper or lower half going to testing was randomly chosen. This approach ensures that all the test data focuses on predicting in new physical regimes given a sparse amount of monomer data. This process was repeated three times for each of M_w, $\dot{\gamma}$, and T to ensure that diverse tests were used for evaluation.

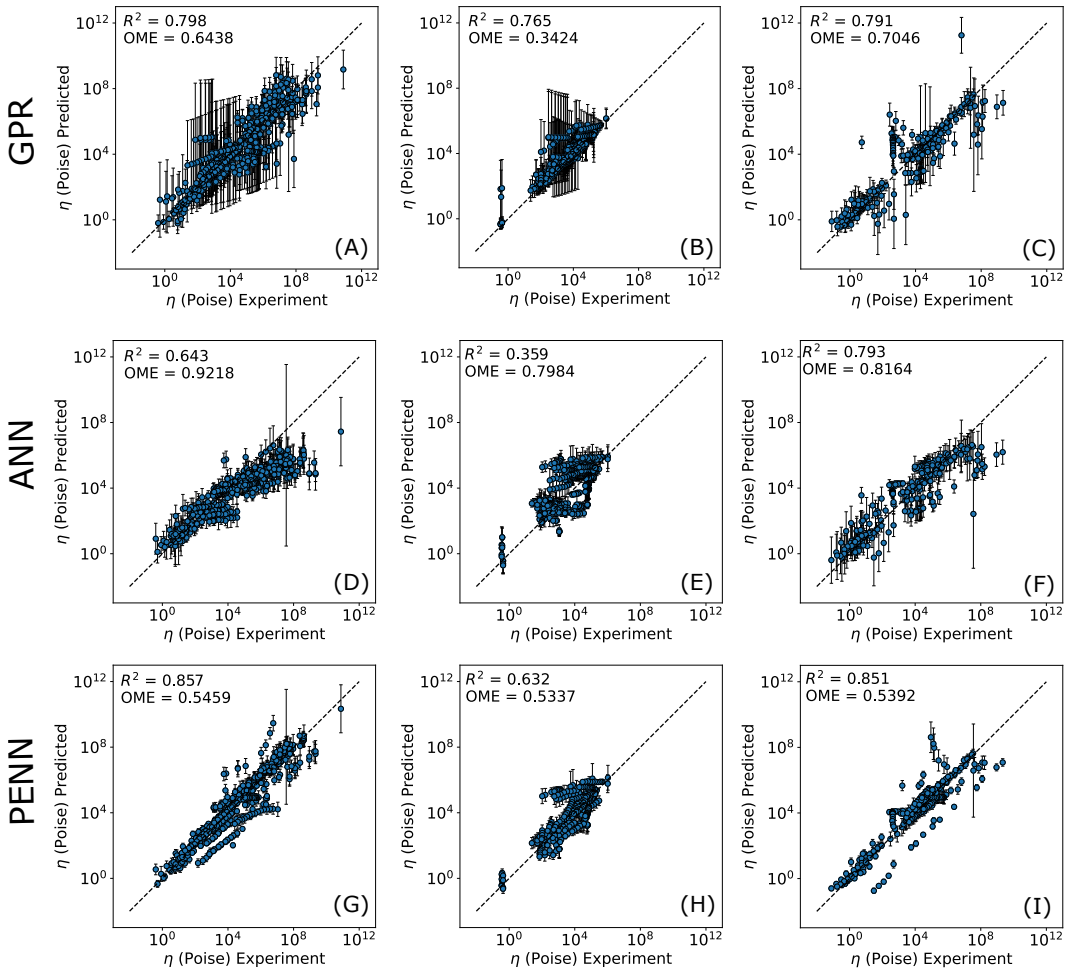

Figure 3: Parity plots are used to assess the models' overall predictive capabilities in new physical regimes based on the physical variable split for molecular weight (M_w), shear rate ($\dot{\gamma}$), and temperature T. Results are compared between Gaussian Process Regression (GPR), Artificial Neural Network (ANN), and Physics Enforced Neural Network (PENN) models. Each plot compares experimental values for melt viscosity (η) to the predicted η across 3 unique test-train splits for each physical variable. The top row (A-C) contains GPR results for A) the M_w split, B) the $\dot{\gamma}$ split C) the T split. The middle row (D-F) contains ANN results for D) the M_w split, E) the $\dot{\gamma}$ split F) the T split. The bottom row (G-I) contains PENN results for G) the M_w split, H) the $\dot{\gamma}$ split I) the T split. The dotted black lines represent perfect predictions. The coefficient of determination (R^2) and Order of Magnitude Error (OME) are reported over these test sets.

Figure 3 shows the combined results of three trials for splits across all three physical variables. SI Section 1 shows parity plots that specify the results from each trial. The GPR, ANN, and PENN predictions have acceptable OMEs, indicating that all three can capture some chemical information and physical trends. The PENN results in a distinct decrease in OME (an average of 35.97% improvement), and an increase in R^2 (up to 79% for the $\dot{\gamma}$ split) from the ANN. The PENN also outperforms the GPR for the M_w and T splits, but the GPR is more accurate on the test set for $\dot{\gamma}$. In further analysis, we show how the physical viability of these predictions is scrutinized beyond the high-level trends of the parity plot.

2.4 Distribution of Predicted Empirical Parameters

Despite the high overall performance of all three models, only the PENN model can produce physically credible predictions in regimes with restricted and sparse data. A comparison of the GPR, ANN, PENN models in estimating crucial empirical parameters (found in Table 1) from sparse data in the held-out set is detailed in Figure 4. We compare these with ground truth values of the parameters from the dataset. The process to obtain these ground truth values is described in Appendix C.

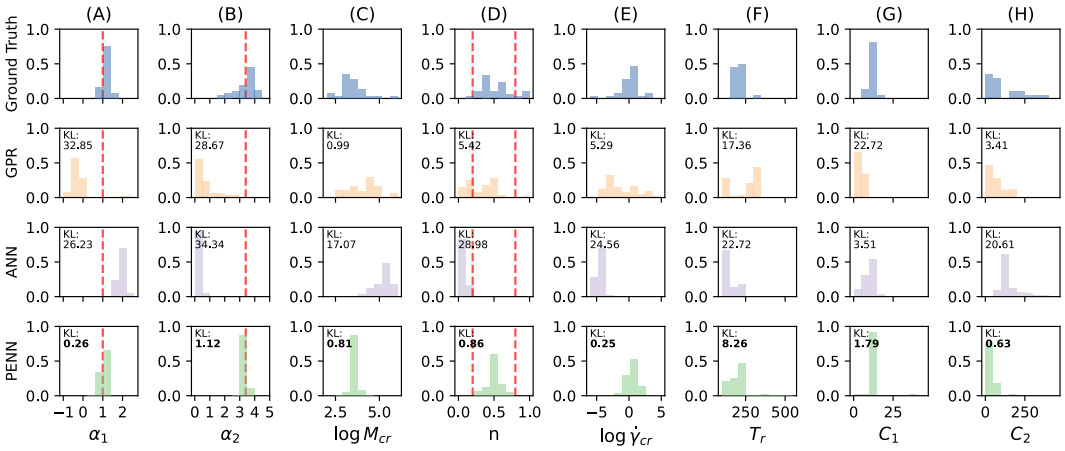

Figure 4: Normalized distributions of empirical parameter values found in the dataset (Ground Truth) are compared to parameter values predicted by Gaussian Process Regression (GPR), Artificial Neural Network (ANN) and Physics Enforced Neural Network (PENN) models. Each column compares a different parameter for the melt viscosity (η) relationship with molecular weight (M_w), shear rate ($\dot{\gamma}$), and temperature (T). The examined parameters include: A) α_1, the slope of zero-shear viscosity (η_0) vs. M_w correlation at low M_w (accepted value of 1 depicted by the red dashed line) B) α_2, the slope of η_0 vs. M_w at high M_w (accepted value of 3.4 depicted by the red dashed line), C) critical molecular weight (M_{cr}), D) n, the rate of shear thinning (accepted range of 0.2-0.8 depicted by the dashed red lines), E) critical shear rate ($\dot{\gamma}_{cr}$), F) reference temperature (T_r) of a polymer. G and H) show distributions for the C_1 and C_2 fitting parameters for the η-T trend. The ground truth distributions represent 41 samples for M_w parameters, 33 samples for $\dot{\gamma}$ parameters, and 22 samples for T parameters. The Kullback–Leibler (KL) divergence of the model estimation distributions from the ground truth is given in the top left of each histogram. The lowest KL divergence among the three models is bolded for each parameter.

We used two different methods to obtain parameter estimations from the models: one method is unique to the PENN model, and another approach for the purely data-driven ANN and GPR. The PENN model automatically predicts each of the empirical parameters (see Figure 1B), which are used in the computational graph to predict η. The ANN and GPR do not directly predict the parameters, so we used a fixed extrapolation procedure. The procedure involved selecting an unseen data point and varying a physical variable (one of M_w, $\dot{\gamma}$, and T) within a predetermined range while holding the other two constant. The ranges for each variable encompass similar orders of magnitude as those present in the training dataset (Figure 2). For M_w extrapolation, a range of $10^2 - 10^7$ g/mol was used to encompass low and high M_w. For shear rate extrapolation, a range of $10^{-5} - 10^6$ 1/s was used to model behaviors in zero-shear and shear-thinning regimes. For temperature extrapolation, ranges of ± 20 K from the original data point's temperature were used to stay within the boundary constraints

of Equation 5. Using this procedure, sets of predictions were made on every unseen datapoint and fit using Equations 5, 6, or 2, yielding estimated values of the empirical parameters.

In Figure 4, we show the feasibility of the models' empirical parameter predictions evaluated against the ground truth values and accepted values. For parameters where a theoretical value is well-defined, the Root Mean Square Error (RMSE) of the predictions' deviation from this value is calculated. The parameter prediction distribution is also compared to the ground truth distribution through a discrete Kullback–Leibler (KL) divergence.

From Figure 4, it can be seen that GPR struggles to predict expected parameter values. The GPR predictions for α_1 deviate from 1 by an RMSE of 1.26. For some polymers, GPR predicts $\alpha_1 \leq 0$. The GPR predictions for α_2 deviate from 3.4 by an RMSE of 2.87, and are significantly lower than the ground truth values in the dataset. Most predicted values for $\log M_{cr}$ are within the same range as the ground truth, but the proper low and high entanglement behavior is not captured which decreases the credibility of these fittings. For the shear thinning parameter n, some values fall within the expected range of $0.2 - 0.8$ [35] for polymer melts, but others are closer to 0, indicating that the expected shear thinning behavior is not always predicted. The predicted $\dot{\gamma}_{cr}$ distribution is lower than the ground truth, indicating that the GPR model forecasts the onset of shear-thinning at a significantly lower $\dot{\gamma}$ than observed (if shear thinning is predicted at all). On temperature dependence, some T_r values are predicted higher than what is seen in the dataset.

The ANN model's failure to capture correct physical trends is also evident in the distributions of its fitted parameters. The RMSEs for the ANN's estimated α_1 and α_2 values are 1.31 and 2.79, respectively. ANN model overestimates α_1 and underestimates α_2 and therefore doesn't capture the effects of high M_w chain entanglement. The ANN also estimates a low n for a subset of polymers, which goes against the definition of shear thinning. The predicted $\dot{\gamma}_{cr}$ values are lower than the ground truth distribution, indicating that the ANN struggles to capture the shear-thinning transition region from the dataset. ANN predictions for the T trend are closest to the ground truth in comparison to its trends of the other variables, because T is a smoother, exponential function (Figure 1A), enabling an easier average fitting.

The PENN model outperforms the ANN in estimating feasible empirical parameters as depicted by lower KL Divergence values in PENN row of Figure 4. The RMSEs of the predicted α_1 and α_2 values are 0.05 and 0.17, which are substantially smaller than that of the ANN. Moreover, all the predicted values of $\log M_{cr}$ are within the ground truth range of $2.5 - 5$. The PENN can also learn the correct shear thinning phenomenon by predicting n values between $0.2 - 0.8$[35] and a $\dot{\gamma}_{cr}$ distribution that mirrors the dataset. The PENN's predicted range of T_r is closest to the ground truth. For the C_1 parameter, the PENN distribution is closest to the proposed value of $C_1 = 7.60$ [36], also having the lowest divergence from the ground truth. For C_2 predictions, although the KL Divergence of the PENN is lower than the ANN, the PENN is confined to much lower values of C_2, and has an average much lower than some experimentally derived values, such as C_2 = 227.3 K [37].

Overall, the average KL divergence across all parameter distributions for the GPR, ANN, and PENN are 14.59, 22.24, and 1.74, respectively. The overall distributions of empirical parameters points to the PENN having greater capabilities for producing physically correct results, than a purely data-driven model.

2.5 Performance in Extrapolative Regimes

In Table 2, we summarized the performance of predicted η profiles over wide ranges of M_w (256 extrapolations), $\dot{\gamma}$ (71 extrapolations), and T (127 extrapolations) for all three models considered. We define a successful extrapolation as a model that can predict the correct trends while maintaining accuracy over the train and test points within the uncertainty. If either of these criteria is not met, the extrapolation is considered to be unsuccessful. Overall, the PENN successfully predicts 80.4% of M_w extrapolations, 49.2% of $\dot{\gamma}$ extrapolations, and 54.1% of T extrapolations. The ANN rarely achieves correct physical trends for M_w or $\dot{\gamma}$ extrapolations in the span of the dataset and only predicts successful profiles for 17.2% of T extrapolations. The GPR model also exhibits a low performance in extrapolation. There are several instances (given in the brackets in Table 2) where the ANN and GPR successfully fit the data points but fail to extrapolate correctly beyond the dataset. This underscores the need for information beyond experimental data to enable extrapolation to new physical regimes.

Model	M_w	$\dot{\gamma}$	T
PENN	**80.4%**	**49.2%**	**54.1%**
ANN	4.30%	4.22%	17.2%
	[64.4%]	[19.7%]	[19.5%]
GPR	0.0%	7.04%	7.03%
	[68.0%]	[40.8%]	[28.1%]

Table 2: Extrapolative predictive performance of the PENN, ANN, and GPR models along the unseen molecular weight (M_w), shear rate ($\dot{\gamma}$), and temperature T regimes. The values within brackets for the ANN and GPR show the percentages of extrapolations where the unseen data was predicted correctly, but the extrapolated trend beyond available data regimes was physically incorrect.

Figure 5 shows a few examples of the extrapolation results summarized in Table 2. A much larger set of examples of both successful and unsuccessful extrapolations by the PENN compared to GPR and ANN are provided in SI Section 2. Figure 5A-C shows examples of the PENN correctly extrapolating η given a small amount of information about a monomer in another part of the physical regime in unseen regimes. The ANN and GPR models are uncertain in these unseen regimes, resulting in large confidence intervals. In Figure 5A, the PENN model accurately predicts the region near M_{cr} where the η-M_w relationship transitions from unentangled to entangled, and can therefore accurately predict η values at high M_w, despite not having seen any data in this region. The errors for the ANN and GPR in Figure 5A are low, within approximately an order of magnitude of error. However, the ANN predictions have a near-constant slope around M_{cr} (implying $\alpha_1 \approx \alpha_2$) and are inconsistent with the effects of polymer chain entanglements at high M_w. The GPR model also fails to predict a higher α_2 slope. In Figure 5B, only the PENN model predicts a zero-shear and shear-thinning region when predicting the η-$\dot{\gamma}$ relationship of the given copolymer. The GPR model fits the training points but mispredicted shear-thinning at high shear rates. The ANN model predicts a decreasing relationship consistent with shear-thinning but doesn't predict the zero-shear region. This could be an example of spectral bias within neural networks [38], where the general decreasing trend of η-$\dot{\gamma}$ is "low frequency" and is captured by the ANN. In contrast, the transition regions are of a "higher frequency" and are not captured by the ANN. In Figure 5C, the PENN model predicts the correct η-T relationship. The ANN model also predicts an exponential relationship but with a higher inaccuracy. The GPR model fits both the training and unseen datapoints, but predicts an unphysical trend beyond this. Overall, the PENN model makes predictions that follow the expected behaviors (Figure 1A) of polymer melts.

Correctly extrapolated samples by the PENN model, such as the ones in Figure 5A-C make up 67.5% of the extrapolated test cases, which is a significant improvement relative to both the ANN and GPR. The PENN model also has room for improvement, especially when applied to datasets with low chemical diversity. Overfitting to a small set of chemistries in training can lead to the inaccurate prediction of parameters when making predictions for unseen chemistries. This behavior is demonstrated in Figure 5D-F, where the PENN predicts a plausible rheological trend but incorrect values for unseen polymers. However, the PENN model introduces a layer of interpretability unavailable to physics-unaware models. Based on the predictions we can reasonably infer which parameters were over- or under-estimated. In Figure 5D, the PENN model predicts near-correct α_1 and α_2 slopes, but the predicted M_{cr} and k_1 values are underestimated. Figure 5E depicts how an underestimated η_0 (caused by inaccuracies in predicted M_{cr}, α_1, α_2 and/or k_1) can cause inaccurate η predictions for all other $\dot{\gamma}$ values. We also see this phenomenon in Figure 5F, where T_r is likely underestimated. The propagating error causes the PENN model to predict an inaccurate trend across the entire spectrum of T. Despite these errors, the pinpointing of the PENN's weak spots can be used to add targeted training data to improve the model. This level of interpretation is unique to the PENN and cannot be done for the GPR and ANN.

These examples of extrapolations provide insights on the applicability of PENN versus pure data-driven methods when using datasets that contain limited chemistries. The equations used in the PENN are based on assumptions and generalizations, and may not account for all physical nuances. Furthermore, the use and the setup of the PENN framework is dependent on the informatics task at hand. The governing equations used must be well-established across the many material chemistries to be used. These must be considered when applying PENNs to future material design and process optimization problems.

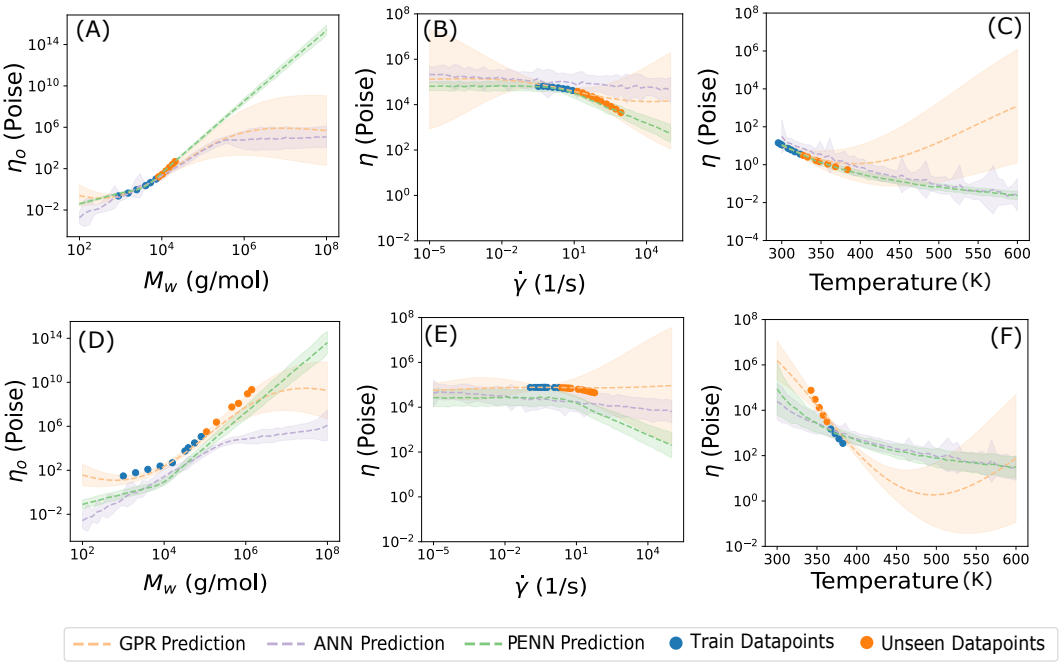

Figure 5: Examples of accurate (A-C) and inaccurate (D-F) melt viscosity (η) and zero-shear melt viscosity (η_0) predictions over wide ranges of molecular weight (M_w), shear rate ($\dot{\gamma}$), temperature (T) by the Physics Enforced Neural Network (PENN) models. The extrapolated predictions are compared to those by Gaussian Process Regression (GPR) and Artificial Neural Network (ANN) models given the same training information. A) is a good η_0-M_w extrapolation for [*]CCCCCCCC-CCOC(=O)CCCCC(=O)O[*] at $T = 382.15$ K. B) is a good η-$\dot{\gamma}$ extrapolation for a copolymer of [*]CC([*])CC(C)C and [*]CC([*])CCCCCCCC (0.968:0.032) ($M_w = 290000$ g/mol, PDI = 7.8) at $T = 543.15$ K. C) is a good η-T extrapolation for [*]CCOCCOCCOC(=O)CCCCCCCCC(=O)O[*] (M_w = 2000 g/mol, $\dot{\gamma}$ = 60 1/s). D) is an unsuccessful η_0-M_w extrapolation for [*]C=CCC[*] at $T = 490.15$ K, with possible mispredictions of M_{cr} and k_1. E) is an unsuccessful η-$\dot{\gamma}$ extrapolation for a copolymer of [*]C[*] and [*]CC([*])OC(C) (0.72:0.28) ($M_w = 60000$ g/mol), with possible misprediction of $\hat{\dot{\gamma}}_{cr}$ and η_0. F) is an unsuccessful η-T extrapolation for [*]CC(O)COc1ccc(C(C)(C)c2ccc(O[*])cc2)cc1 (M_w = 1696 g/mol, $\dot{\gamma}$ = 0.0 1/s) with a possible misprediction of T_r.

3 Conclusion

In this study, we introduce a Physics Enforced Neural Network (PENN), a strategy that combines data-driven techniques with established empirical equations, to predict the melt viscosity of polymer melts with better physics-guided generalization and extraplation. The PENN makes predictions across many chemical compositions and relevant physical parameters, including molecular weight, shear rate, temperature, and polydispersity index. We compared our PENN approach against the purely data-driven, physics-unaware, Artificial Neural Network and Gaussian Process Regression. In extrapolative regimes, our PENN model outperforms the physics-unaware counterparts and offers an elevated level of interpretability and generalizability. To enhance generalizability across chemistries, future work could increase the chemical space in the dataset through new experiments, molecular dynamics simulations, and/or more aggressive data acquisition from literature.

This work has profound implications for additive manufacturing (AM) and materials informatics. The PENN model's capability to guide the rheological control of diverse polymer resins accelerates the development of new printing materials, thereby expanding AM's utility. Our methodology offers a blueprint for modeling other properties governed by empirical equations. The initial success of the PENN architecture for melt viscosity is a powerful demonstration of how data-driven insights combined with established knowledge can propel us into a new era of rapid advancements in materials science and engineering.

Acknowledgments and Disclosure of Funding

This work was financially supported by the Office of Naval Research (ONR) through Grant N00014-21-1-2258 and the National Science Foundation (NSF) DMREF Grant 2323695. A.J. acknowledges S. Shukla for help with the fingerprinting of polymer blends. A.J. also acknowledges Dr. Lihua Chen and Prof. Christopher Kuenneth for valuable discussions on materials informatics approaches.

References

[1] C. Wang, X.P. Tan, S.B. Tor, and C.S. Lim. Machine learning in additive manufacturing: State-of-the-art and perspectives. *Additive Manufacturing*, 36:101538, 2020. ISSN 2214-8604. doi: https://doi.org/10.1016/j.addma.2020.101538. URL `https://www.sciencedirect.com/science/article/pii/S2214860420309106`.

[2] Arit Das, Eric L. Gilmer, Saeid Biria, and Michael J. Bortner. Importance of polymer rheology on material extrusion additive manufacturing: Correlating process physics to print properties. *ACS Applied Polymer Materials*, 3(3):1218–1249, 2021. doi: 10.1021/acsapm.0c01228. URL `https://doi.org/10.1021/acsapm.0c01228`.

[3] Lihua Chen, Ghanshyam Pilania, Rohit Batra, Tran Doan Huan, Chiho Kim, Christopher Kuenneth, and Rampi Ramprasad. Polymer informatics: Current status and critical next steps. *Materials Science and Engineering: R: Reports*, 144:100595, 2021.

[4] Rohit Batra, Le Song, and Rampi Ramprasad. Emerging materials intelligence ecosystems propelled by machine learning. *Nature Reviews Materials*, 6(8):655–678, 2021.

[5] Huan Tran, Rishi Gurnani, Chiho Kim, Ghanshyam Pilania, Ha-Kyung Kwon, Ryan P Lively, and Rampi Ramprasad. Design of functional and sustainable polymers assisted by artificial intelligence. *Nature Reviews Materials*, pages 1–21, 2024.

[6] Huan Doan Tran, Chiho Kim, Lihua Chen, Anand Chandrasekaran, Rohit Batra, Shruti Venkatram, Deepak Kamal, Jordan P Lightstone, Rishi Gurnani, Pranav Shetty, et al. Machine-learning predictions of polymer properties with polymer genome. *Journal of Applied Physics*, 128(17):171104, 2020.

[7] Arun Mannodi-Kanakkithodi, Anand Chandrasekaran, Chiho Kim, Tran Doan Huan, Ghanshyam Pilania, Venkatesh Botu, and Rampi Ramprasad. Scoping the polymer genome: A roadmap for rational polymer dielectrics design and beyond. *Materials Today*, 21(7):785–796, 2018. ISSN 1369-7021. doi: https://doi.org/10.1016/j.mattod.2017.11.021. URL `https://www.sciencedirect.com/science/article/pii/S1369702117307344`.

[8] Lihua Chen, Joseph Kern, Jordan P Lightstone, and Rampi Ramprasad. Data-assisted polymer retrosynthesis planning. *Applied Physics Reviews*, 8(3):031405, 2021.

[9] Rishi Gurnani, Deepak Kamal, Huan Tran, Harikrishna Sahu, Kenny Scharm, Usman Ashraf, and Rampi Ramprasad. Polyg2g: A novel machine learning algorithm applied to the generative design of polymer dielectrics. *Chemistry of Materials*, 33(17):7008–7016, 2021.

[10] Di-Fan Liu, Qi-Kun Feng, Yong-Xin Zhang, Shao-Long Zhong, and Zhi-Min Dang. Prediction of high-temperature polymer dielectrics using a bayesian molecular design model. *Journal of Applied Physics*, 132 (1), 2022.

[11] R.G. Lorenzini, W.M. Kline, C.C. Wang, R. Ramprasad, and G.A. Sotzing. The rational design of polyurea & polyurethane dielectric materials. *Polymer*, 54(14):3529–3533, 2013. ISSN 0032-3861. doi: https://doi.org/10.1016/j.polymer.2013.05.003. URL `https://www.sciencedirect.com/science/article/pii/S0032386113004230`.

[12] Christopher Kuenneth, William Schertzer, and Rampi Ramprasad. Copolymer informatics with multitask deep neural networks. *Macromolecules*, 54(13):5957–5961, 2021.

[13] Hirotomo Moriwaki, Yu-Shi Tian, Norihito Kawashita, and Tatsuya Takagi. Mordred: a molecular descriptor calculator. *Journal of cheminformatics*, 10(1):1–14, 2018.

[14] Rishi Gurnani, Christopher Kuenneth, Aubrey Toland, and Rampi Ramprasad. Polymer informatics at scale with multitask graph neural networks. *Chemistry of Materials*, 35(4):1560–1567, 2023.

[15] Rishi Gurnani, Stuti Shukla, Deepak Kamal, Chao Wu, Jing Hao, Christopher Kuenneth, Pritish Aklujkar, Ashish Khomane, Robert Daniels, Ajinkya A Deshmukh, et al. Ai-assisted discovery of high-temperature dielectrics for energy storage. *Nature Communications*, 15(1):6107, 2024.

[16] Ugandhar Delli and Shing Chang. Automated process monitoring in 3d printing using supervised machine learning. *Procedia Manufacturing*, 26:865–870, 2018.

[17] Zeqing Jin, Zhizhou Zhang, Kahraman Demir, and Grace X Gu. Machine learning for advanced additive manufacturing. *Matter*, 3(5):1541–1556, 2020.

[18] Shivank S Shukla, Christopher Kuenneth, and Rampi Ramprasad. Polymer informatics beyond homopolymers. *arXiv preprint arXiv:2303.12938*, 2023.

[19] Brandon K Phan, Kuan-Hsuan Shen, Rishi Gurnani, Huan Tran, Ryan Lively, and Rampi Ramprasad. Gas permeability, diffusivity, and solubility in polymers: Simulation-experiment data fusion and multi-task machine learning. *npj Computational Materials*, 10(1):186, 2024.

[20] Xiangnan Ye and Tam Sridhar. Effects of the polydispersity on rheological properties of entangled polystyrene solutions. *Macromolecules*, 38(8):3442–3449, 2005.

[21] Mahdi Boudaghi-Khajehnobar, Brian J Edwards, and Bamin Khomami. Effects of chain length and polydispersity on shear banding in simple shear flow of polymeric melts. *Soft Matter*, 16(28):6468–6483, 2020.

[22] D. Nichetti and I. Manas-Zloczower. Viscosity model for polydisperse polymer melts. *Journal of Rheology*, 42(4):951–969, 1998. doi: 10.1122/1.550908. URL `https://doi.org/10.1122/1.550908`.

[23] Haoxiang Li, Hao Tian, Yewei Chen, Sian Xiao, Xiuying Zhao, Yangyang Gao, and Liqun Zhang. Analyzing and predicting the viscosity of polymer nanocomposites in the conditions of temperature, shear rate, and nanoparticle loading with molecular dynamics simulations and machine learning. *The Journal of Physical Chemistry B*, 127(15):3596–3605, 2023.

[24] Bijaya Karki, Diwas Bhattarai, and Jian Zhang. Modeling viscosity of silicate melts using machine learning techniques. In *AGU Fall Meeting Abstracts*, volume 2021, pages MR45A–0078, 2021.

[25] Shingo Takada, Toru Suzuki, Yoshihiro Takebayashi, Takumi Ono, and Satoshi Yoda. Machine learning assisted optimization of blending process of polyphenylene sulfide with elastomer using high speed twin screw extruder. *Scientific reports*, 11(1):1–9, 2021.

[26] M. B. Abdullah, M. Delshad, K. Sepehrnoori, M. T. Balhoff, J. T. Foster, and M. T. Al-Murayri. Physics-Based and Data-Driven Polymer Rheology Model. *SPE Journal*, 28(04):1857–1879, 08 2023.

[27] GP Pun, R Batra, R Ramprasad, and Y Mishin. Physically informed artificial neural networks for atomistic modeling of materials. *Nature communications*, 10(1):1–10, 2019.

[28] Zhizhou Zhang and Grace X Gu. Physics-informed deep learning for digital materials. *Theoretical and Applied Mechanics Letters*, 11(1):100220, 2021. ISSN 2095-0349. doi: https://doi.org/10.1016/j.taml.2021.100220. URL `https://www.sciencedirect.com/science/article/pii/S2095034921000258`.

[29] Gabriel Bradford, Jeffrey Lopez, Jurgis Ruza, Michael A. Stolberg, Richard Osterude, Jeremiah A. Johnson, Rafael Gomez-Bombarelli, and Yang Shao-Horn. Chemistry-informed machine learning for polymer electrolyte discovery. *ACS Central Science*, 9(2):206–216, 2023. doi: 10.1021/acscentsci.2c01123. URL `https://doi.org/10.1021/acscentsci.2c01123`.

[30] M. Mattheakis, P. Protopapas, D. Sondak, M. Di Giovanni, and E. Kaxiras. Physical symmetries embedded in neural networks, 2019. URL `https://arxiv.org/abs/1904.08991`.

[31] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2018.10.045. URL `https://www.sciencedirect.com/science/article/pii/S0021999118307125`.

[32] Shingo Otsuka, Isao Kuwajima, Junko Hosoya, Yibin Xu, and Masayoshi Yamazaki. Polyinfo: Polymer database for polymeric materials design. In *2011 International Conference on Emerging Intelligent Data and Web Technologies*, pages 22–29, 2011. doi: 10.1109/EIDWT.2011.13.

[33] Ankit Rohatgi. Webplotdigitizer: Version 4.6, 2022. URL `https://automeris.io/WebPlotDigitizer`.

[34] George Wypych. ChemTec Publishing, 2016. ISBN 978-1-895198-92-8. URL `https://app.knovel.com/hotlink/toc/id:kpHPE00012/handbook-polymers-2nd/handbook-polymers-2nd`.

[35] Nickolas D. Polychronopoulos and John Vlachopoulos. *Polymer Processing and Rheology*, pages 133–180. Springer International Publishing, Cham, 2019. ISBN 978-3-319-95987-0. doi: 10.1007/978-3-319-95987-0_4. URL `https://doi.org/10.1007/978-3-319-95987-0_4`.

[36] Syed Ali Ashter. 6 - mechanics of materials. In Syed Ali Ashter, editor, *Thermoforming of Single and Multilayer Laminates*, pages 123–145. William Andrew Publishing, Oxford, 2014. ISBN 978-1-4557-3172-5. doi: https://doi.org/10.1016/B978-1-4557-3172-5.00006-2. URL `https://www.sciencedirect.com/science/article/pii/B9781455731725000062`.

[37] M. Mours and H.H. Winter. Chapter 5 - mechanical spectroscopy of polymers. In Toyoichi Tanaka, editor, *Experimental Methods in Polymer Science*, Polymers, Interfaces and Biomaterials, pages 495–546. Academic Press, Boston, 2000. ISBN 978-0-08-050612-8. doi: https://doi.org/10.1016/B978-0-08-050612-8.50011-8. URL `https://www.sciencedirect.com/science/article/pii/B9780080506128500118`.

[38] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pages 5301–5310. PMLR, 2019.

[39] CA Hieber and HH Chiang. Shear-rate-dependence modeling of polymer melt viscosity. *Polymer Engineering & Science*, 32(14):931–938, 1992.

[40] John Vlachopoulos, Nickolas Polychronopoulos, and M Kontopoulou. Basic concepts in polymer melt rheology and their importance in processing. *Applied Polymer Rheology: Polymeric Fluids with Industrial Applications*, pages 1–27, 2011.

[41] M. Rubinstein and R. H. Colby. *Polymer Physics*. Oxford University Press, Oxford, 2003.

[42] Montgomery T. Shaw. On finding the zero-shear-rate viscosity of polymer melts. *Polymer Engineering & Science*, 61(4):1166–1178, 2021. doi: https://doi.org/10.1002/pen.25661. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/pen.25661`.

[43] C. A. Hieber and H. H. Chiang. Shear-rate-dependence modeling of polymer melt viscosity. *Polymer Engineering & Science*, 32(14):931–938, 1992. doi: https://doi.org/10.1002/pen.760321404. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/pen.760321404`.

[44] Hideroh Takahashi, Takaaki Matsuoka, and Toshio Kurauchi. Rheology of polymer melts in high shear rate. *Journal of Applied Polymer Science*, 30(12):4669–4684, 1985. doi: https://doi.org/10.1002/app.1985.070301214. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/app.1985.070301214`.

[45] GT Helleloid. On the computation of viscosity-shear rate temperature master curves for polymeric liquids. 2001.

[46] Peter Rhys Lewis. Chapter 2 - sample examination and analysis. In Peter Rhys Lewis, editor, *Forensic Polymer Engineering (Second Edition)*, Woodhead Publishing in Materials, pages 33–69. Woodhead Publishing, second edition edition, 2016. ISBN 978-0-08-101055-6. doi: https://doi.org/10.1016/B978-0-08-101055-6.00002-1. URL `https://www.sciencedirect.com/science/article/pii/B9780081010556000021`.

[47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[49] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. 2016. doi: 10.48550/ARXIV.1603.06560. URL `https://arxiv.org/abs/1603.06560`.

[50] Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*, 2018.

Appendix

A PENN Architecture

A.1 Enforced Polymer Physics Trends

In this section, we detail the physics-based correlations included within the Physics Enforced Neural Network (PENN). We enforce dependencies of η on temperature (T), molecular weight (M_w), and shear rate ($\dot{\gamma}$) through $\eta(M_w, T, \dot{\gamma})$, which we derive below.

A.1.1 Preamble: Smoothing of Piecewise Functions

When going from one function $g(a, b)$ in a low regime ($a < b$) to another function $h(a, b)$ in a high regime ($a > b$), we can use the smoothened Heaviside step function,

$$H_\beta = \frac{1}{1+\exp(-\beta x)}, \tag{1}$$

where β is a tunable rate of transition.

A function $f(a, b)$ that transitions from $g(a, b)$ to $h(a, b)$ is given by

$$f(a, b) = g(a, b) \cdot H_\beta(b - a) + h(a, b) \cdot H_\beta(a - b)$$

A.1.2 η dependence on $\dot{\gamma}$, T, and M_w

The η dependence on $\dot{\gamma}$ follows the physics of shear-thinning fluids [39, 40, 41]. In these fluids, at low $\dot{\gamma}$, there is not enough force between chains to break entanglements and cause movement, so η remains constant at η_0. At a critical shear rate, $\dot{\gamma}_{cr}$, the shear force is high enough to cause chain alignment, making chain diffusion easier. Beyond $\dot{\gamma}_{cr}$, η decreases according to a shear-thinning linear power law. [39]. This trend can be represented by a function (Equation 2) across both the zero-shear and shear thinning regimes [42, 43, 39, 44],

$$\begin{aligned} \eta(M_w, T, \dot{\gamma}) &= \frac{\eta_0(M_w, T)}{(1 + \frac{\dot{\gamma}}{\dot{\gamma}_{cr}})^{1-n}} \\ \log \eta &= \eta_0(M_w, T) + (n - 1) \log(1 + \frac{\dot{\gamma}}{\dot{\gamma}_{cr}}) \end{aligned} \tag{2}$$

where the parameter n describes the sensitivity to shearing [45]. For shear-thinning fluids, $n < 1$. For most polymer melts, n is empirically known to be in the range of $0.2 - 0.8$.[35]

Equation 2 is unfavorable to use directly because $\dot{\gamma}$ spans several orders of magnitude, so $\log \dot{\gamma}$ must be used as an input. Equation 2 cannot be adapted to use $\log \dot{\gamma}$ as an input (due to the +1 in the denominator), so we depict the relationship across the low $\dot{\gamma}$ and high $\dot{\gamma}$ regimes as a piecewise function on the log-scale,

$$\log \eta = \begin{cases} \log \eta_0 & \text{if } \dot{\gamma} << \dot{\gamma}_{cr} \\ \log \eta_0 + (n - 1) \log(\frac{\dot{\gamma}}{\dot{\gamma}_{cr}}) & \text{if } \dot{\gamma} >> \dot{\gamma}_{cr} \end{cases} \tag{3}$$

We smooth Equation 3 with $H_{\beta_{\dot{\gamma}}}$ to get $\log \eta(M_w, T, \dot{\gamma})$ (Equation 4),

$$\begin{aligned} &\log \eta(M_w, T, \dot{\gamma}) = \log \eta_0(M_w, T) \cdot H_{\beta_{\dot{\gamma}}}(\log \dot{\gamma}_{cr} - \log \dot{\gamma}) \\ &+(\log \eta_0(M_w, T) + (n - 1) \log(\frac{\dot{\gamma}}{\dot{\gamma}_{cr}})) \cdot H_{\beta_{\dot{\gamma}}}(\log \dot{\gamma} - \log \dot{\gamma}_{cr}), \end{aligned} \tag{4}$$

where $\beta_{\dot{\gamma}}$ is a parameter that dictates the rate of shift from zero-shear to shear-thinning. For our implementation, we found that optimization over the $\dot{\gamma}$ domain was optimal when $\beta_{\dot{\gamma}} = 30$.

$\log \eta_0(M_w, T)$ is defined by the T dependence. As temperature increases, so does the rate of molecular self-diffusion, resulting in lower η seen in fluidic polymer melts [41]. The William-Landel-Ferry (WLF) equation[36, 37] describes the exponential decrease in η as the temperature increases. Therefore, we can encode temperature dependence as

$$\begin{aligned} \eta_0 &= \eta_{M_w} \cdot 10^{\frac{-C_1(T-T_r)}{C_2+(T-T_r)}}, \forall T \geq T_r \\ \log \eta_0(M_w, T) &= \log \eta_{M_w} \cdot \frac{-C_1(T - T_r)}{C_2 + (T - T_r)}, \forall T \geq T_r \end{aligned} \tag{5}$$

where T_r is a reference temperature and C_1 and C_2 are material-dependent empirical parameters. The values for these are dependent on polymer chemistry. C_1 = 7.60 and C_2 = 227.3 K are examples of values that have been proposed [37] from observations of experiments on a small subset of polymers. The reference temperature T_r is within a few degrees of the glass transition temperature T_g. It has been proposed that the WLF relationship holds within the range of T_g to $T_g + 200$K [37].

η_{M_w} is defined by the M_w dependence. Longer and heavier polymer chains experience increased entanglements, which hinder chain reptation in the polymer melt at low shear. [46, 41] Equation 6 is a piece-wise power law that describes this phenomenon.

$$\eta_{M_w} = \begin{cases} k_1 M_w^{\alpha_1} & \text{if } M_w < M_{cr} \\ k_2 M_w^{\alpha_2} & \text{if } M_w \geq M_{cr} \end{cases} \tag{6}$$

where

$$k_2 = k_1 M_{cr}^{\alpha_1 - \alpha_2}.$$

M_{cr} is the critical molecular weight, above which entanglement density is high enough to increase the impact of M_w on η_0. The two power laws intersect at $M_w = M_{cr}$ [46]. M_{cr} is found to be approximately 2-4 times the molecular weight at which chain entanglement starts, but the exact value is polymer dependent.[41] α_1 is the slope of the $\log \eta_0$-$\log M_w$ curve if $M_w < M_{cr}$ and α_2 is the slope if $M_w \geq M_{cr}$. Typically, α_1 is theoretically and empirically determined to be about 1, while α_2 is found to be about 3.4 [46, 41], but the exact value is dependent on the polymer. k_1 and k_2 are the y-intercepts of each power law and are polymer-dependent.

M_w and η_0 span several orders of magnitude, so we use Equation 6 in the log-scale to get Equation 7,

$$\log \eta_{M_w} = \begin{cases} \log k_1 + \alpha_1 \log M_w & \text{if } M_w < M_{cr} \\ \log k_1 + (\alpha_1 - \alpha_2) \log M_{cr} + \alpha_2 \log M_w & \text{if } M_w \geq M_{cr} \end{cases} \tag{7}$$

Smoothing Equation 7 with $H_{\beta_{M_w}}$ gives Equation 8,

$$\begin{aligned} \log \eta_{M_w} =& [\log k_1 + \alpha_1 \log M_w] \cdot H_{\beta_{M_w}}(\log M_{cr} - \log M_w) \\ &+ [\log k_1 + (\alpha_1 - \alpha_2) \log M_{cr}] \cdot H_{\beta_{M_w}}(\log M_w - \log M_{cr}) \end{aligned} \tag{8}$$

where $H_{\beta_{M_w}}$ is the smoothened Heaviside step function using β_{M_w}, a parameter which dictates the rate of shift from α_1 to α_2.

Therefore, Equations 4, 5, and 8 determine the $\log \eta(M_w, T, \dot{\gamma})$. The predicted parameters n, $\dot{\gamma}_{cr}$, $\beta_{\dot{\gamma}}$ determine the $\dot{\gamma}$ dependence in $\log \eta(M_w, T, \dot{\gamma})$, which is also a function of $\eta_0(M_w, T)$. The predicted parameters C_1, C_2, and T_r determine the T dependence in $\eta_0(M_w, T)$, which is also a function of η_{M_w}. The predicted parameters α_1, α_2, M_{cr}, β_{M_w}, and k_1 determine the M_w dependence in η_{M_w}.

A.2 Bounding Ranges of Physical Parameters within the PENN Framework

The final constants output from the MLP use the Sigmoid function to hold them to physically meaningful ranges. The ranges also help reduce possible imbalances of gradients and/or exploding gradients that may occur during backpropogation, because of the complexity of the computation graph. In this work, we use a rudimentary approach to solve this problem, described in Table 3 and more complete solutions may be developed in future works.

A.3 PENN Training

This entire PENN architecture is trained, in part, to minimize the error of viscosity predictions. The sum of these errors across all n training points is called the viscosity loss $\mathcal{L}_\eta$, defined in Equation 9. Each data point is denoted by its index i.

$$\mathcal{L}_\eta = \frac{1}{n} \sum_{i=1}^{n} (\hat{\eta}_i - \eta_i)^2 \tag{9}$$

During training, we add loss terms (see Equation 10) to penalize the predicted α_1 and α_2 for the ith training point ($\hat{\alpha_{1,i}}$ and $\hat{\alpha_{2,i}}$, respectively) for deviating from their average values. The viscosity loss plus the penalty terms form the total loss $\mathcal{L}$.

$$\mathcal{L} = \mathcal{L}_\eta + \frac{1}{n} \sum_{i=1}^{n} w_\alpha [(\hat{\alpha_{1,i}} - 1)^2 + (\hat{\alpha_{2,i}} - 3.4)^2] \tag{10}$$

w_α is a hyperparameter that controls the impact that known values of the α_1 and α_2 parameters have on the final loss.

Table 3: Bounded Ranges and Justifications of Empirical within the PENN Framework

Parameter	Bounding Range	Justification
M_{cr}	(-1,1)	Keep critical value within M_w ranges of the dataset
α_1	(0,3)	Bound to practical value near 1
α_2	(0,6)	Bound to practical value near 3.4
k_1	(-1.5,0.5)	Keep viscosity value within η ranges of the dataset
β_M	(20, 50)	Appropriate range to control transition region, found through trial-and-error
C_1	(0,2)	Keep within practical ranges with regards to temperature scaling
C_2	(0,2)	Keep within practical ranges with regards to temperature scaling
T_r	(-1.5,1)	Keep reference temperature within T and just below T range of dataset
$\dot{\gamma}_{cr}$	(-1, 1)	Keep critical value within $\dot{\gamma}$ ranges of the dataset
n	(0,1)	Keep slope within range for general shear thinning fluids
$\beta_{\dot{\gamma}}$	(30)	Appropriate range to control transition region, found through trial-and-error

B Training Approaches and Hyperparameter Tuning

The PENN and ANN models were implemented in PyTorch [47]. All models were trained on the same 9:1 (Train:Test) split. Before training, the features and η were scaled to a range of (-1,1). The polymer fingerprint, PDI, and temperature were scaled with the Scikit-Learn MinMaxScaler [48] to a range of (-1,1). The $\dot{\gamma}$ was scaled by first adding a small value of 10^{-5}, taking the $\log_{10}$, and then scaling to (-1,1). M_w was scaled by taking the $\log_{10}$ value and then scaling to (-1,1). For the PENN, $\log M_w$ and $\log \dot{\gamma}$ use the same scaling bounds as η.

Within the training set, a 10-fold cross-validation (CV) was used to ensure that the models did not overfit the training set. The ANN and PENN models also had separate models trained for each CV split. Hyperparameter optimization was performed using the Hyperband [49] optimization algorithm over each CV fold for both the ANN and the PENN models, with RayTune [50] implementations, respectively. The ANN and PENN models, both containing 4 layers (including 2 hidden layers), involved optimization of the same hyperparameters: layer 1 size (64, 128, 256, 512), layer 1 dropout (0,0.01, 0.015,0.02,0.025,0.03), layer 2 size (64, 128, 256, 512), layer 2 dropout (0,0.01,0.015,0.02,0.025,0.03), and weight decay (0.00001, 0.00005, 0.0001, 0.0005, 0.001). For the PENN, w_α (0.001, 0.005, 0.01, 0.03, 0.05) was also optimized. The value corresponding to the lowest $\mathcal{L}_\eta$ (Equation 9) of the CV test split was used.

The Adam optimizer was used to train the models with a learning rate (LR) reduction by a factor or 0.5 on the plateau of the validation loss given a patience of 20 epochs. An initial LR of 0.0001 was used for the PENN. Empirically, we found that the PENN tuning was sensitive to high LR. The initial LR for the ANN was 0.001. Training was stopped with an Early Stopping patience of no improvement in the validation loss after 25 epochs.

The GPR model was implemented using Scikitlearn [48] and trained using Bayesian optimization to tune key hyperparameters. The hyperparameters optimized include the noise level (`alpha`) with a range of $[10^{-2}, 10^{1}]$, the length scale of the RBF kernel (`length_scale`) with a range of $[10^{-2}, 10^{2}]$, and the constant value used in the kernel (`constant_value`) with a range of $[10^{-2}, 10^{2}]$, each with a logarithmic uniform prior. The optimization was performed over 50 iterations each over the 10-fold cross-validation, with the best-performing model parameters selected based on the results. The scaling for the inputs and outputs of the GPR were the same as the ANN.

C Obtaining Ground Values of Empirical Parameters from the Dataset

To establish a benchmark for comparing the three models, we obtained ground truth values of the parameters from the dataset. We did this by identifying subsets of our dataset involving the same

polymer with measured η of several T, M_w, or $\dot{\gamma}$. If a subset contained at least five points, we fitted the corresponding equation (Equation 5, 6, or 2) to obtain empirical parameters. The distributions of these ground truth parameter values are shown in the first row of Figure 4. There are a limited number of ground truth values because a small number of datapoints satisfy the above conditions. Nevertheless, this small sample set allowed us to make a few inferences about expected viscosity trends. The ground truth values of α_1 and α_2 are close to the theoretical values of 1 and 3.4, respectively[46]. α_2 values were occasionally less than the expected 3.4, possibly due to outliers or errors in fitting a small number of datapoints. The fitted $\log M_{cr}$ values fell within a range of $10^{2.5} - 10^5$ g/mol. For shear parameters, the majority of samples are found to have n in a range of $0.2 - 0.8$, which is typical for polymer melts [35]. The obtained $\dot{\gamma}_{cr}$ values were found in the range of $10^{-3} - 10^4$ 1/s. The fitted T_r values are mostly in a range of $T_r < 250K$. This is low when compared to T_g values found in thermal property datasets [12]. In our dataset, the datapoints that could be fitted to the η-T relationship were observed at $T < 475K$, so low T_r values could be overrepresented in the ground truth. The C_1 parameter average was 11.8 and the C_2 parameter average was 159.42 K. This analysis of the ground truth data suggests desired parameter values our models should predict.

D Computational Resources

We conducted our experiments using a Tesla V100-PCIE-32GB GPU. The GPU was utilized with CUDA version 12.3, ensuring compatibility with the latest libraries and frameworks and optimizing performance through advanced parallel computing capabilities.

The training of each PENN model and ANN model took 300MiB of GPU space. Each individual model took approximately 10 minutes to hyperparameter tune, about 35 minutes to train. The total training time for a PENN or ANN experiment across 10 CV folds took approximately 450-500 minutes. The total compute is estimated to be about 4500 minutes for the experiments in this manuscript. The full research project required many iterations of tuning to understand how to build the computational graph. Therefore we estimate the total compute to be about 10-15 times the compute used in the final experiments.

E Additional Parity Plots for Test Sets

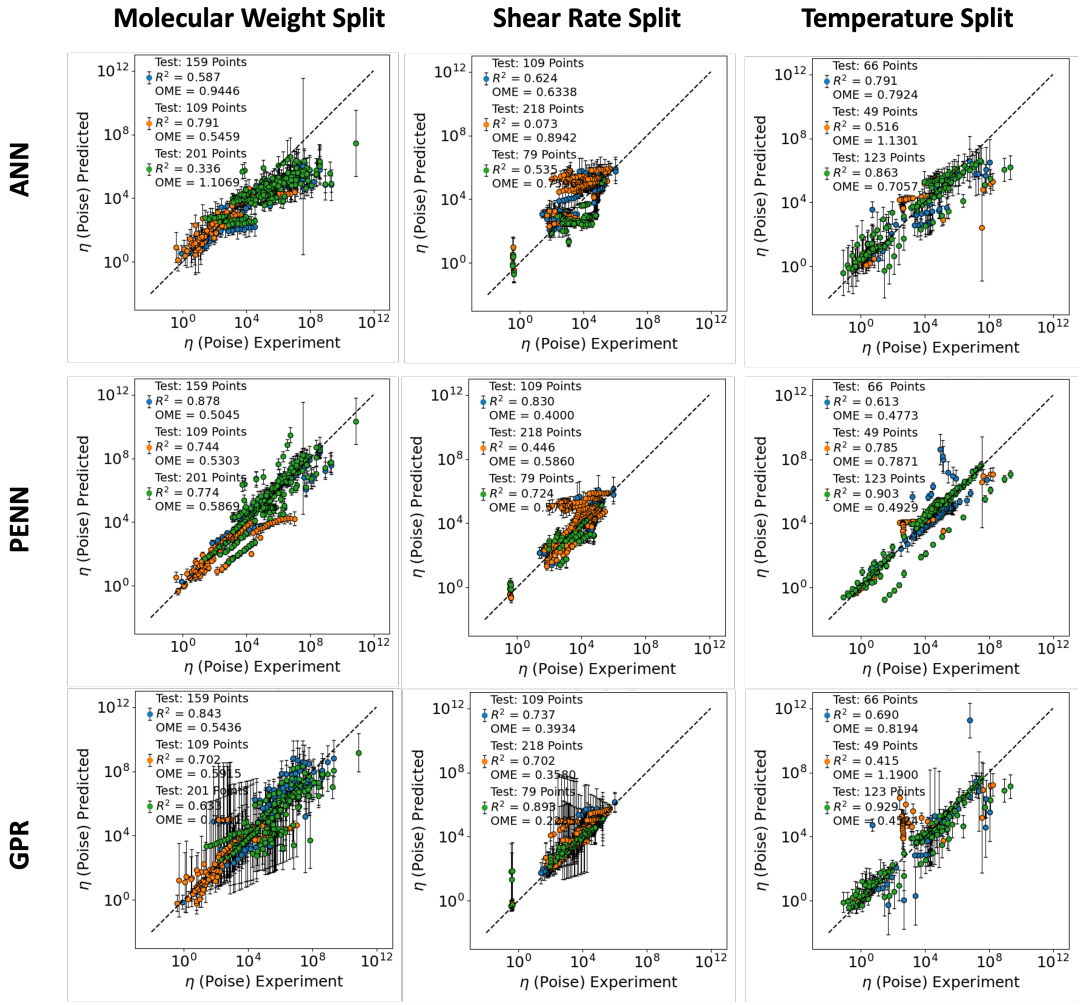

Figure 6: Parity plots containing trial information and the test sizes from each trial. Each plot compares experimental values for melt viscosity (η) to the predicted η. The dotted black lines represent perfect predictions. The coefficient of determination (R^2) and Order of Magnitude Error (OME) are reported over each test set trial.

F Additional Extrapolation Plots in Various Test Cases

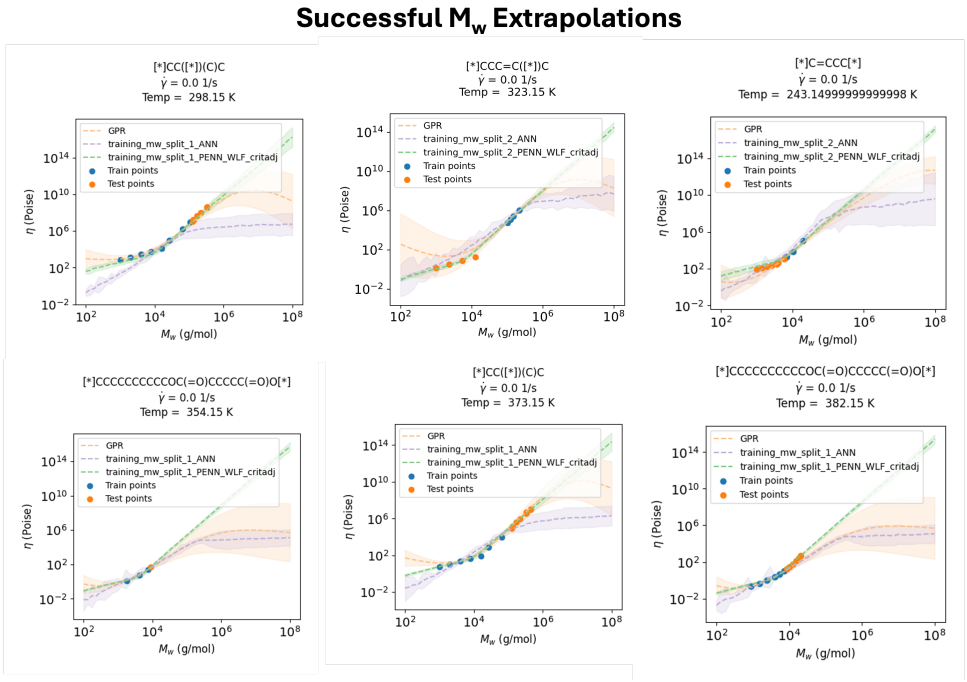

Figure 7: Examples successful of molecular weight extrapolations on partially seen and unseen monomers.

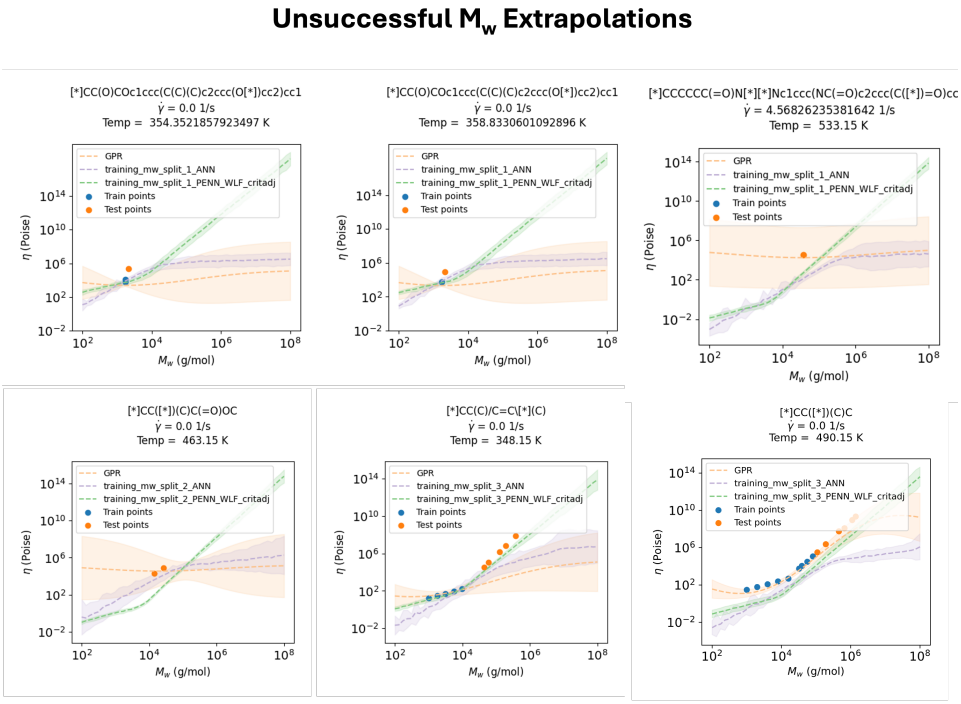

Figure 8: Examples of unsuccessful molecular weight extrapolations on partially seen and unseen monomers. PENN extrapolations consist of data points that lie outside of the uncertainty bounds of the predictions.

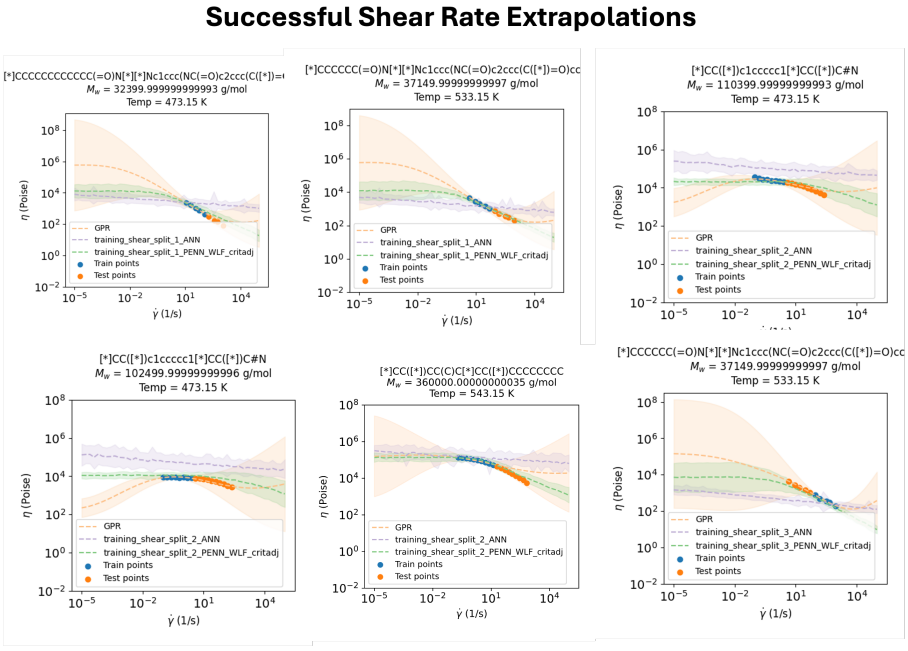

Figure 9: Examples of successful shear rate extrapolations on partially seen and unseen monomers.

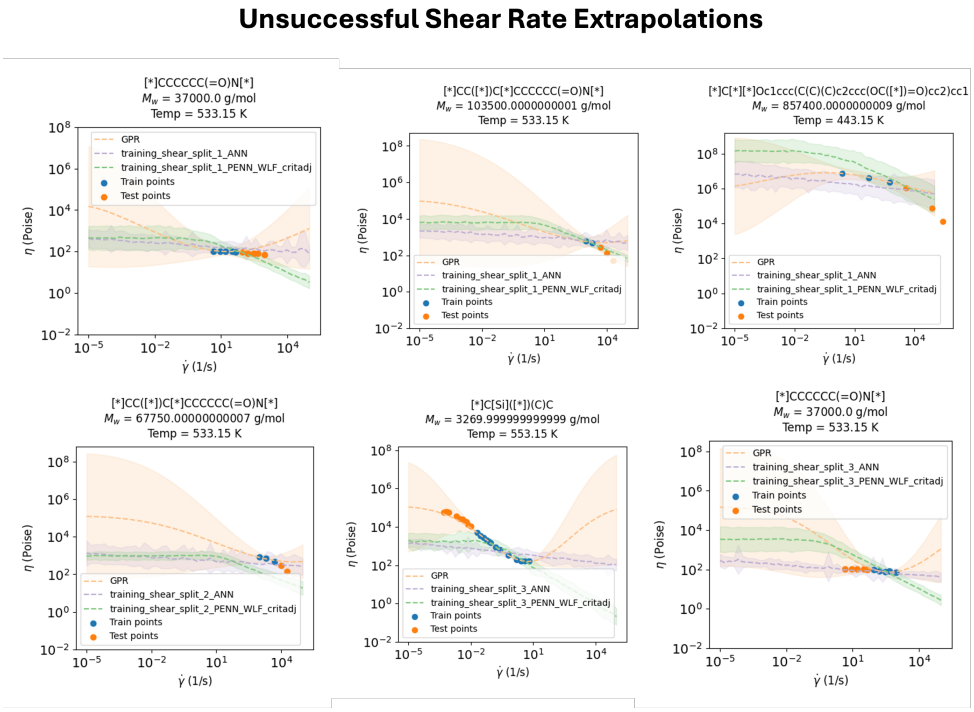

Figure 10: Examples of unsuccessful shear rate extrapolations on partially seen and unseen monomers. PENN extrapolations consist of data points that lie outside of the uncertainty bounds of the predictions and/or incorrect predictions of the $\dot{\gamma}_{cr}$ transition region.

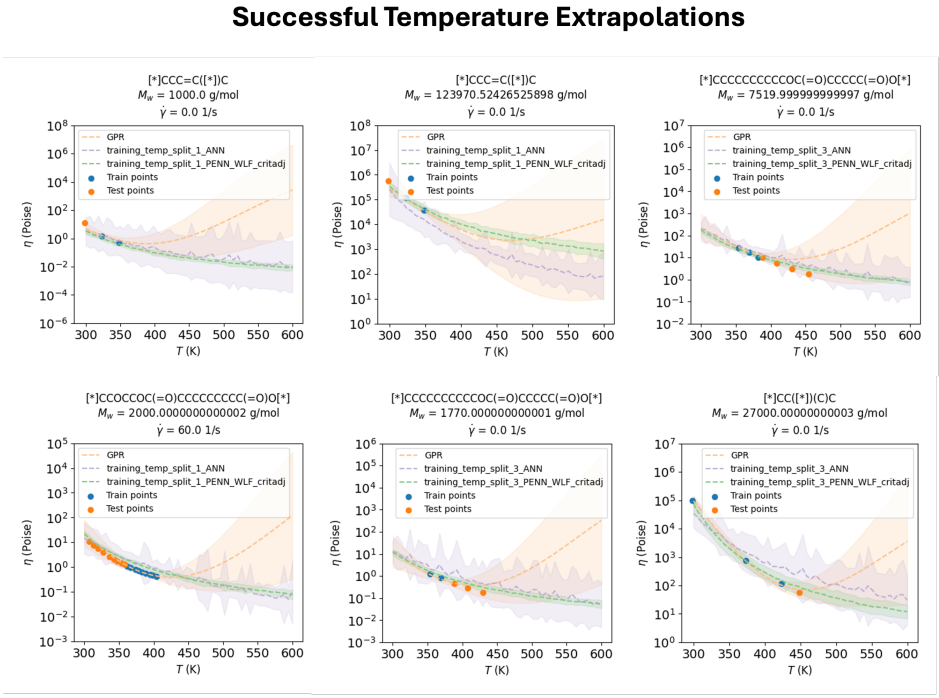

Figure 11: Examples of successful temperature extrapolations on partially seen and unseen monomers.

Unsuccessful Temperature Extrapolations

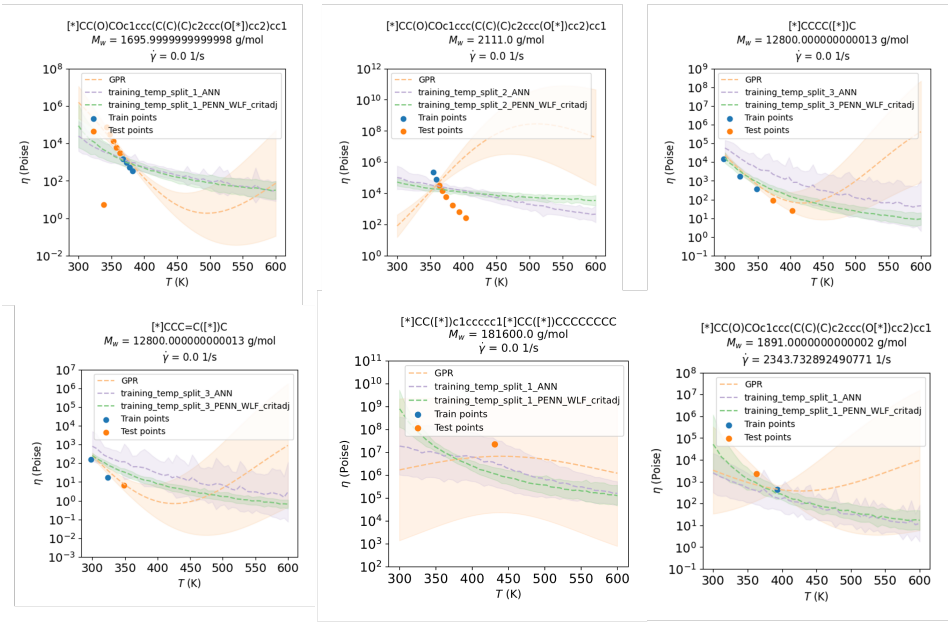

Figure 12: Examples of unsuccessful temperature extrapolations on partially seen and unseen monomers. PENN extrapolations consist of data points that lie outside of the uncertainty bounds of the predictions.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction discuss the scope of the work applied to polymer melt viscosity and the ability to extrapolate to unseen physical settings. This is reflected in the results.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the shortcomings of the PENN approach and the necessary considerations when applying the method. Examples demonstrating limitations are given in Figure 5.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The assumptions and theory in this paper are addressed. The theory is in the physical characteristics of polymer melts and assumptions made from these are given throughout the paper and in Appendix A.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The details of the dataset, splitting, architecture details, and training information/hyperparameter tuning are all provided.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The authors have released the dataset on the open-source section of our research group github. The code will be released upon this work's publication in a journal. Enough information to reproduce the PENN and the experiments are provided otherwise.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: This information is provided in the Section 2.2, Section 2.3 and Appendix B.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We quantify uncertainties and report error bars where necessary, shown in Figures 3 and 5.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The authors explain the computational usage in Appendix D.

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Upon review, the authors believe that this follows the code of ethics.

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer:[NA] .

    Justification: The authors believe that there is not enough societal impact from this work to be discussed because it is in the materials science field.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA] .

    Justification: The authors believe that there is no major risk from this work because it is in the materials science field.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The credit and refer to the creators of Polyinfo [32] where the dataset is from and Polymer Genome [6, 5].

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: The authors do not provide any assets in this paper.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: There are no human subjects in this work.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA] .

    Justification: The paper does not involve crowdsourcing or research with human subjects.