Highlights

A conceptual study of transfer learning with linear models for data-driven property prediction

Bowen Li, Srinivas Rangarajan

- A conceptual study of when transfer learning improves data-driven models.
- Using an interpretable model to infer how shared features affect transfer learning.
- Illustration of transfer learning in multi-fidelity predictive models.
- Illustration of transfer learning for molecular property models.
- Example of detrimental effect of transfer learning between disparate properties.

A conceptual study of transfer learning with linear models for data-driven property prediction

Bowen Li[a], Srinivas Rangarajan[a,*]

[a]*Department of Chemical and Biomolecular Engineering, Lehigh University, 111 Research Dr, Bethlehem, 18015, PA, USA*

Abstract

Transfer learning is a concept whereby data-driven models can be developed for tasks (e.g. molecular properties) with limited data availability (target task) by sharing information from a related task. In the context of chemical engineering, the two tasks can either pertain to related properties or to the same property calculated or measured in two different ways (with differing accuracies or resolution). Using an ensemble of linear and interpretable models, in this work, we present a conceptual study to explicate when transfer learning can be beneficial. We show that a large overlap of the underlying features of the two tasks (specifically greater than 50%) is required for transfer learning to improve the model for the target task. On the other hand, transferring information (in particular, information regarding salient features) from an uncorrelated task can be detrimental to train a model for the target task. Subsequently, we present three illustrative examples of transfer learning for molecular property prediction and rationalize the usefulness of transferred information based on the inferences from our conceptual studies. This work, thus, provides a simplified analysis of the concept of transfer learning for building molecular property models.

Keywords: Transfer learning, Machine learning, Molecular property

1. Introduction

Supervised learning has achieved great success in many machine learning applications, such as classification of images (Li et al., 2018; He et al., 2019; Raj et al., 2020), clustering of text (Yao et al., 2019; Yang et al., 2019b), materials discovery (Nikolaev et al., 2016; Xue et al., 2016; Kollmann et al., 2020; Ryan et al., 2018; Hegde, 2020; Jablonka et al., 2020; Cova and Pais, 2019; Meyer et al., 2018) and drug discovery (Korkmaz, 2020; Gentile et al., 2020; Janet and Kulik, 2017). In chemical engineering and related disciplines, supervised learning has been widely developed to map the molecular structure, such as atom coordinates or molecular fragments, to various property metrics (Mater and Coote, 2019; Hansen et al., 2015; Duvenaud et al., 2015; Gómez-Bombarelli et al., 2018; Coley et al., 2017; Yang et al.,

*Corresponding author

Email address: srr516@lehigh.edu (Srinivas Rangarajan)

2019a; Christensen et al., 2020; Zubatyuk et al., 2019; Westermayr and Marquetand, 2020). These models have been shown to achieve high levels of accuracy (Schutt et al., 2018; Bogojeski et al., 2020; Bartók et al., 2017; Christensen et al., 2020; Unke and Meuwly, 2019; Lubbers et al., 2018); however, they also require large amounts of data. Often, however, data are scarce and expensive to acquire, which then can limit the accuracy and generalizability of the models developed with traditional supervised learning methods.

One could look at human learning to address this challenge. Humans tend to apply previous experience and acquired knowledge to a new task, thereby "transferring" knowledge between related tasks. The concept of transfer learning works in a similar manner. In particular, if (1) data for a particular target task is limited, (2) there exists one or more related tasks whose domains or distributions are well correlated (but not necessarily identical), and (3) there is a sufficiently large amount data pertaining to these related tasks, then the performance of data-driven models of the target task can be improved by transferring information such as features, model structure, or parameters from related tasks. Examples of transfer learning can be found for image recognition (Bird et al., 2020; Kan et al., 2014), self-driving cars (Kim and Park, 2017; Xing et al., 2018), robotics (Rusu et al., 2017) and chemistry and drug design (Turki et al., 2017; Ward et al., 2019; Han and Choi, 2021; Peng et al., 2019; Cai et al., 2020).

Many properties, e.g. bulk molecular properties such as boiling and melting points or enthalpy and entropy of formation, likely have similar underlying features or data embeddings. Conceivably, transfer learning can work in such instances whereby models can be built for those properties for which data are scarce by leveraging information from related properties for which larger datasets are available (Iovanac and Savoie, 2020; Grambow et al., 2019). Further, since a molecular property can be estimated in multiple ways, e.g. via computations and experiments or by using two different levels of theory, multiple non-overlapping datasets of differing size and fidelity may be available for property of interest. Often, in these cases, the more accurate method is also more expensive and, consequently, the high fidelity property data set is significantly smaller in size than the relatively low fidelity one. In such instances, transfer learning can be employed to build models on the smaller (and relatively high fidelity) data set while concurrently harnessing information from the larger (low fidelity) dataset to ultimately build accurate multi-fidelity models.

For molecular and material property prediction, transfer learning is often applied in the context of training a neural network model (Smith et al., 2019). Typically, the first step is to pre-train a model with sufficient data in a related task. It is argued that, in these neural network models, the underlying features are learned in the first few layers (e.g. convolutional layers), therefore, to pass knowledge from the related task to the target task, the structure of these initial layers is transferred to the new model and the corresponding parameters are either frozen (or sometimes fine-tuned with some regularization). Subsequently, a few additional layers are added and the model is trained with the limited data of the target task. This way, the number of parameters that need to be tuned for the target model is minimized. For example, to develop a neural network for predicting the water solubility of molecules, Lentelink and Palkovits (2020) pre-trained a neural network for predicting molecular weight on a subset of the GDB17 database (Ruddigkeit et al., 2012) and transferred the first few

layers to the new model that was then trained on the ESOL dataset (Delaney, 2004). Lee and Asahi (2021) trained a graph convolutional neural network on the heat of formation to learn the representation of the crystal structures, and then used it as the transferred knowledge to predict multiple properties associated with crystals such as bulk moduli and dielectric constants. Grambow et al. (2019) showed that, by learning the embeddings and parameters with sufficient data from a lower level of theory (B3LYP) from the QM9 dataset (Ramakrishnan et al., 2014), the prediction of the heat of formation and other properties with the higher level of theory (CCSD(T)) could be improved with transfer learning. Further, Yamada et al. (2019) developed a library called XenonPy.MDL, which contains many pre-trained models including various properties for small molecules. They demonstrated that these models could improve the prediction of material properties of organic and inorganic chemistry.

The principal contribution of this work to provide a fundamental conceptual understanding of the requirements on the datasets for transfer learning to be beneficial. Specifically, using interpretable linear models, we first discuss the following conceptual question: consider two different properties, ρ_1 and ρ_2, which depend on a similar (partially overlapping) set of features, with the first having a large dataset while the other has a relatively small dataset for training data-driven models M_1 and M_2 respectively. Then, (1) to what extent can the model M_1 provide additional information (transferred knowledge) about important features to the training process for the second property so that a superior model (M_2') relative to M_2 can be trained? and (2) how does the extent of overlap in features between the two properties or the size of the training set for ρ_2 influence the performance of M_2' relative to M_2? We assume that the true relationships between the properties and the features are linear and the space of plausible features is much larger than the true number of features that the two properties depend on. While most properties are highly nonlinear functions of their features that are often best captured by neural networks, we reckon that these assumptions/simplifications are not too restrictive for the purpose of developing a conceptual understanding of transfer learning for molecular or material properties. Subsequently, we demonstrate three illustrative examples of transfer learning in molecular property prediction covering instances of both beneficial and detrimental effects of using "transferred information" and we rationalize these observations based on insights from the conceptual study.

2. Methods

In this section, we primarily discuss the methods employed in the conceptual case study; however, the description of building sparse linear models is also relevant for the three transfer learning examples.

Data generation:. For our conceptual study, our approach to address the problem defined above is to carefully synthesize the datasets D_1 and D_2 to allow for a systematic quantification of the difference between a transfer learned model M_2' and the original model M_2. The detailed steps for dataset creation are as follows. We first generate a random data matrix

X with the dimension of $(n \times p, n \approx p)$, where n is the number of data points and p is the cardinality of a parent set of features, P, i.e. $p = |P|$. Then two column-subsets of X, viz. X_1 and X_2, of dimensions $(n \times p_1)$ and $(n \times p_2)$ respectively are constructed, where we set $p_1 = p_2$ for simplicity. The two respective subsets of features, P_1 and P_2, have some overlap that is pre-determined, and is quantified by a ratio, α, defined as $\alpha = \frac{|P_1 \cap P_2|}{p_1}$. Two sets of coefficients, β_1 and β_2, are then sampled from a uniform distribution to generate the vectors of property values, $y_i = X_i\beta_i$, to which we add some uniform noise to ultimately obtain the synthetic datasets. In particular, $D_i = \{X_i, y_i\}$. It should be noted that all the data points in D_1 are used for training/validation for M_1 while a random subset of D_2, viz. D_2^{tr} containing m data points, where $m << n$, is chosen for training/validation of M_2. This ensures that the dataset used to train data-driven models of property ρ_2 is substantially smaller than D_1; the rest of the data points in D_2, $n - m$, are simply used for testing the predictive accuracy of both M_2 and M_2'.

Transfer learning analysis:. Once D_1 and D_2 are generated, the transfer learning study is formulated by comparing the performance of models trained on D_2^{tr} with different levels of knowledge about the potential features involved. We assume that the true features P_2 are unknown, therefore, we train M_2 directly on D_2^{tr} by identifying a sparse linear model (following the procedure described below) from the set of "plausible" features, P. For the transfer learned model M_2', we "estimate" P_2 using M_1, then a sparse model is built starting from this set of plausible features. If the estimated P_2 is accurate and smaller than P, the sparse model starts with fewer parameters and a small dataset D_2^{tr} is sufficient to train it accurately. In such a case, transfer learning is beneficial. If the estimated P_2 is poorly representative of the true features, then learning a sparse model starting from this incorrect set will expectedly lead to a poor model. We compare three cases of transfer learned models (M_2') with respect to a sparse model M_2 directly trained only on D_2^{tr}. These cases are: (1) M_2' is trained on D_2 but assuming that P_1 is known a priori and only these features matter for ρ_2 (i.e. assuming $P_1 \supseteq P_2$); (2)M_2' is trained on D_2 but assuming that, while $P_1 \supseteq P_2$, P_1 is also unknown and needs to be inferred by training a sparse model M_1 on D_1; and (3) a model trained on D_2, but assuming that, while $P_1 \not\supseteq P_2$, the features in P_1 have a higher likelihood of also being important for property ρ_2. It should be noted that sparsity theoretically enforces (as discussed below) the selection of only relevant features in M_1 or M_2, however, the noise in the data may result in some true features being missed or incorrect features being selected, which may in turn have implications on the effectiveness of transfer learning.

Parameter settings for the conceptual study:. In this work, the number of data points, n, and the cardinality of a parent set of features, p, of the parent data matrix X were set to be 1000. The dimension for features, p_1 and p_2, of the two constructed column-subsets X_1 and X_2 was set to be 200. After D_1 and D_2 are generated, all 1000 data points in D_1 are available for the model M_1; for D_2, unless otherwise specified, we set $m = 300$ for D_2^{tr} where 200 data points are used for training and the rest 100 are for validation to determine the optimal penalty λ during LASSO regression. The remaining 700 points in D_2 then are used as the testing set to compare the performance of the models M_2 and M_2'.

Building sparse linear models:. We build our data-driven models using the least absolute shrinkage and selection operator (LASSO), which extends standard linear least-squares regression by introducing a penalty value λ into the objective function of linear regression. For instance, the coefficients β_1 of M_1 are determined by:

$$\hat{\beta}_1 = arg \min_{\beta_1} \left\{ \frac{1}{n} \left\| y_1 - X_1 \beta_1 \right\|_2^2 + \lambda \left\| \beta_1 \right\|_1 \right\} \tag{1}$$

The penalty λ allows for pushing insignificant features to zero; thus, we assume all features of P are plausible to begin with and let the model determine the important features whose coefficient values remain non-zero. The value of λ determines the balance between model accuracy and model compactness. Larger λ leads to fewer features but the resulting model may not have a high level of accuracy; on the other hand, smaller values of λ might end up discarding many of the truly insignificant features in P. The optimal λ is found by performing a grid search on the validation set, where a range of λ values are tested and compared. While LASSO offers an automated way to balance the trade-off between model accuracy and compactness, there are ways to regularize or design neural networks to minimize the number of nodes and obtain a compact representation vector for the inputs which, in turn, can be transferred between models.

Finally, we note that (1) the values of X are sampled uniformly between 0 and 1, (2) the coefficients β_1 and β_2 are sampled uniformly between 0.5 and 1, and (3) the noises are of the distribution $N \sim \{0, 1\}$. The range for the grid search for LASSO is defined as $\{(2^i) \, | i = i_{upper} : i_{low} : -1\}$, where different values of λ are tested on a validation set by decreasing from $2^{i_{upper}}$ to $2^{i_{low}}$ and divided by 2 iteratively. We further note that the accuracy of the models is determined using the root mean squared error (RMSE).

3. Results and discussion

3.1. Conceptual study for transfer learning

As discussed earlier, our goal is to investigate if the knowledge gained from a related task (i.e. building M_1) could help to learn the target task (i.e. building M_2) better. To this end, we here discuss systematic case studies to quantify the difference between a transfer learned model M_2' and the original model M_2, where the comparison involves different levels of transferred knowledge from D_1 as well as the different extent of overlap between P_1 and P_2, i.e. different values of α. We consider three case studies involving different assumptions about P_1 and P_2.

3.1.1. Case 1: Transfer learning assuming $P_1 \supseteq P_2$ and P_1 is known

We start the analysis with the most straightforward case, where we assume the set of features P_1 of ρ_1 is known a priori and is the same as P_2. Figure 1 shows the performance of M_2 and M_2' in terms of the root mean square error, or RMSE, on the testing set as the overlap between features of D_1 and D_2, determined by α, increases. Here, for simplicity, we fix D_2 and modify D_1 to modulate α; therefore, the performance of M_2 is independent of α. For small overlap of features (i.e. for small values of α), the transfer learned model performs

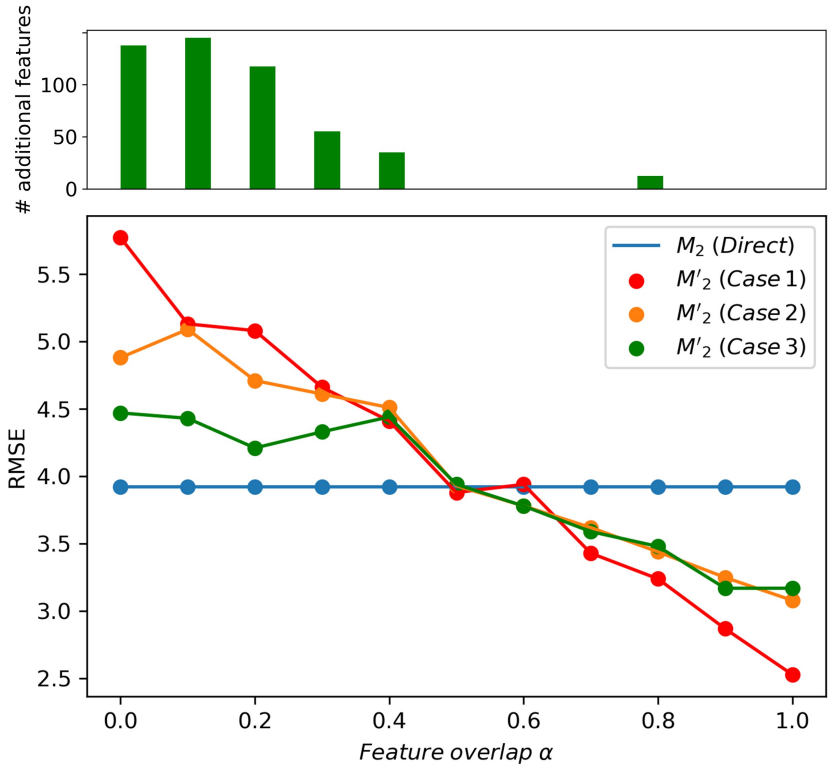

Figure 1: Comparison of the performance of the direct learning model M_2 ('Direct') with transfer learned models M_2' in 3 cases; Case 1: Transfer learning assuming $P_1 \supseteq P_2$ and P_1 is known. Case 2: Transfer learning assuming $P_1 \supseteq P_2$ but P_1 is unknown. Case 3: Transfer learning assuming $P_1 \not\supseteq P_2$ but P_1 is unknown. The feature overlap α reflects the percentage of the overlapped features in P_1 and P_2. The bar plot at the top shows the number of additional features in the model M_2' in Case 3 compared to that in Case 2.

poorly compared to the directly learned model as the features in P_1 are not sufficient to capture ρ_2. That is, the knowledge of P_1 misguides M_2' in this instance. On the other hand, for large values of α, indicative of substantial overlap of true features between the properties, the transfer learned model performs significantly better than M_2. That is, transfer learning of the model M_2' is beneficial in this instance. The crossover (or the intersection) between the two curves occurs roughly at $\alpha \sim 0.5$. This is a critical point because it essentially determines when transfer learning becomes useful.

This crossover point is dependent on the size of the training set available for M_2. Figure 2 shows that as the training set size increases, the crossover point shifts to a larger value of α. This is because, the performance of M_2 progressively becomes better and the line corresponding to M_2 in Figure 1 shifts down (thereby shifting the point of intersection further to the right). This effectively means that transferring knowledge from M_1 becomes progressively less effective in improving the model for the property ρ_2. Indeed, as the training set size increases to become comparable to D_1, the crossover value of α reaches 1.0 indicating that D_2 is large enough that unless $P_1 \equiv P_2$, there is no advantage gained from transfer learning.

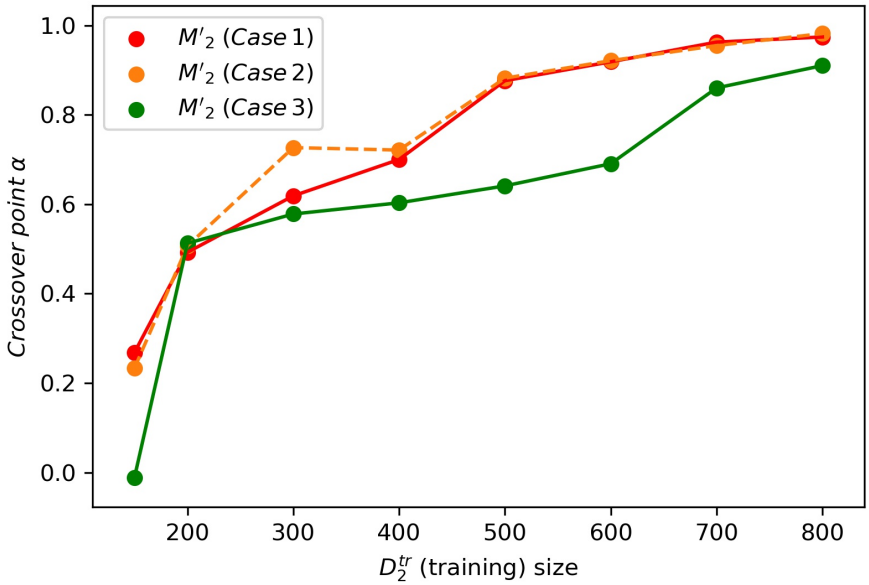

Figure 2: Crossover value of the feature overlap α corresponding to different size of the training set of D_2^{tr} for the three cases. Each point represents the minimum value of α at which the direct learned model M_2 starts performing worse than the transfer learned model M_2' for a given training set size. Note that only the training portion of D_2^{tr} is plotted on the x-axis, the validation set is kept constant at 100 and the remaining points in D_2 are used for testing.

3.1.2. Case 2: Transfer learning assuming $P_1 \supseteq P_2$ but P_1 is unknown

We demonstrated the effectiveness of transfer learning where we know the important features a priori. In reality, such information may not be readily available and P_1 needs to be estimated via unsupervised or supervised learning. Here, we use LASSO on ρ_1 to generate a model M_1 and thereby identify the important features P_1. The LASSO feature selection is carried out by randomly splitting D_1 into the 0.8/0.2 ratio of train/validation set, and a grid search is performed on the validation set to find the optimal λ. Given the noise in the training set, we expect that the identified set of features, $\hat{P}_1$, is not the true P_1; the primary question then is to understand the impact of this noise in transfer learning.

Figure 1 shows the performance of M_2 and M_2' for this case, assuming $\hat{P}_1 \supseteq P_2$. We can observe that M_2' in Case 2 outperforms the corresponding model in Case 1 for small α, while it underperforms relative to Case 1 for larger values of α, especially after the crossover point. Interestingly, in this case, we found that the optimal LASSO model selected ~ 400 features, i.e. $|\hat{P}_1| \sim 400$, while the true number of features for ρ_1 is 200, i.e., $|P_1| = 200$; however, all features in P_1 were selected by LASSO despite the noise in the data. Clearly, this explains why the qualitative behavior is similar to the first case. However, the relative performance of the transfer learned models of Case 1 and Case 2 suggests that the extra features identified for M_1, viz. $\hat{P}_1 - P_1$, aids transfer learning at low feature overlap in Case 2 while a smaller set of features (viz. P_1) used for Case 1 aids in transfer learning

at large values of α. This is because, at low overlap between P_1 and P_2, there is likely a larger overlap of features between $\hat{P}_1$ and P_2; therefore, transfer learning from $\hat{P}_1$ provides a larger set of feature choices for developing M_2'. On the other hand, this extra set of features is detrimental at larger overlap because the noise in the data results in the LASSO model picking incorrect features from $\hat{P}_1$ to build M_2'.

3.1.3. Case 3: Transfer learning assuming $P_1 \not\supseteq P_2$ but P_1 is unknown

Here, we further extend the transfer learning investigation by allowing for the more realistic situation that there are features that influence ρ_2 which are not important for ρ_1. We accomplish this by training M_2' on all features (similar to M_2) but prioritizing the features in $\hat{P}_1$ (found by LASSO) by assigning a relatively higher penalty (λ) for the unselected features (i.e. those in $P - \hat{P}_1$) than for those in $\hat{P}_1$. This forces LASSO to favor those features in $\hat{P}_1$ but allows for selecting other features if they are critical. The optimal values of the two penalties, say λ_1 for those in $\hat{P}_1$ and λ_2 for the rest, are computed by a grid search to minimize the errors in the validation set but we constrain the ratio of the penalties, i.e. $\lambda_2 : \lambda_1 \geq k$. Here, we set $k = 2$.

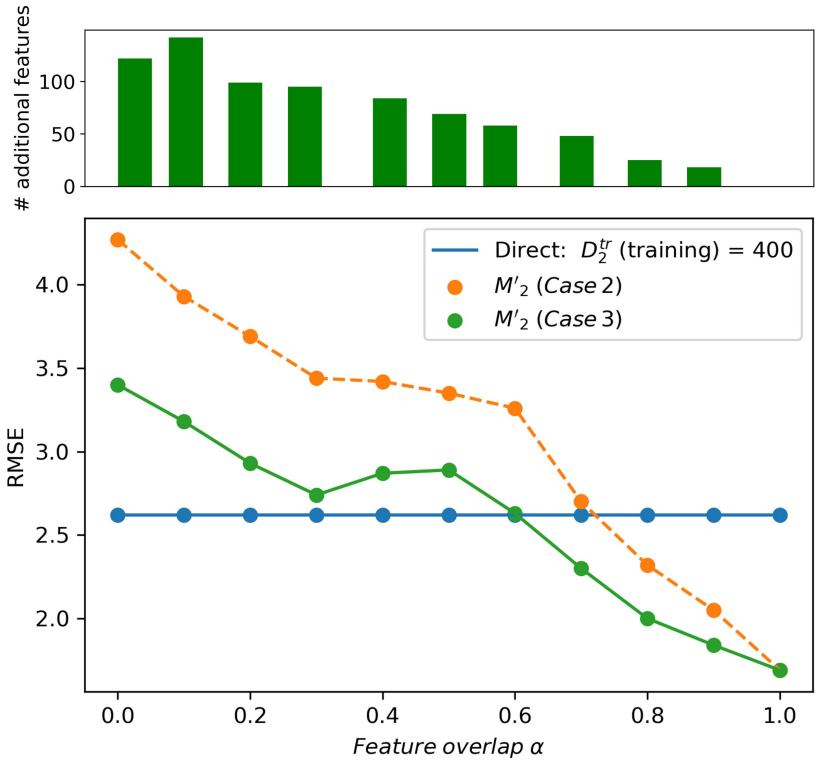

Figure 3: Comparison of the performance of the transfer learned model M_2' in Case 2 and Case 3 by increasing D_2^{tr} (training set) size from 200 to 400. The RMSE of the M_2' prediction on the remaining set of D_2 are reported.

Ideally, such a procedure of prioritization should produce models that are at least as good as the direct model (M_2) for small values of α while retaining or surpassing the performance

of M_2' from Case 2 for larger values of α. Figure 1 shows the results for this case; we can note that transfer learning with such prioritization of features leads to improved performance compared to the transfer learned model of Case 2. Beyond the crossover point, the transfer learned models for Cases 2 and 3 are nearly identical. This is also seen from the bar graph in Figure 1 which shows that no (or very few) additional unselected features had to be included in the new model (compared to M_2' from Case 2). For smaller values of α, on the other hand, the new M_2' performs better than in Case 2 but it is still not as good as the direct model M_2. Furthermore, LASSO adds almost fifty new features from the unselected set, which along with the features in $\hat{P}_1$ is able to partially overcome the low overlap between the true features in P_1 and P_2. Decreasing the ratio k may move the transfer learned model closer to M_2 for smaller values of α but may make it worse for larger values of α. Finally, the transfer learned model in Case 1 still outperforms Case 3 for larger values of α indicating the importance of having a smaller set of features to choose from when the overlap of features between P_1 and P_2 is large. Nevertheless, in the absence of information regarding feature overlap or about P_1, case 3 offers the best overall performance.

Figure 2 shows that crossovers occur earlier in Case 3 compared to Case 2 regardless of the training set size. Since the $m = 300$ (200 for training, 100 for validation) setting is the only instance where the crossover points of Case 2 and Case 3 are nearly the same, we further examined these two cases for $m = 500$ (400 for training and 100 for validation). The results are in Figure 3. Unlike in Figure 1, for most values of α, additional features (over and above that in Case 2) were needed for Case 3, and the performance of the model from Case 3 surpasses the performance from Case 2 even beyond the crossover point. Thus, crossover points occur earlier for Case 3.

Since the three cases were evaluated using a single randomly generated pair of datasets (D_1 and D_2), we evaluated the generalizability of these results by considering an ensemble of 100 different synthesized datasets. The results, discussed in detail in the supporting information (S1), clearly shows that $\alpha = 0.5$ is the typical crossover point for Case 1, while lower values of α are sufficient for the other two cases, consistent with the results presented above. This shows that our results should generalize well as evident from the illustrative examples considered next.

3.2. Illustrative examples of transfer learning

Table 1: Comparison of direct and transfer learning for three illustrative examples

examples	datasets	**RMSE (direct)**	**RMSE (transfer)**	D1 LASSO features	D2 LASSO features	common features (α)
1	$D_2 : H_f$(G4MP2) $D_1 : H_f$(B3LYP)	**5.61 (kcal/mol)**	**4.81**	169	164	147 (0.90)
2	D_2: BP D_1: MP	**33.57 (C)**	**31.36**	181	209	94 (0.45)
3	D_2: BP D_1: H_f(G4MP2)	**33.57 (C)**	**34.20**	110	209	61 (0.29)

Using the insights from the conceptual study, we present and discuss three examples of transfer learning as given below.

- *Example 1:* Developing a model for the heat of formation (H_f) using data at the G4MP2 level of theory by transfer learning from a larger (and less accurate) B3LYP dataset. We use the first thousand (1000) smallest C, O, and H containing molecules from the QM9 dataset whose heats of formation values are available at the B3LYP level of theory. We use the G4MP2 values for these molecules from a recently reported work (Narayanan et al., 2019).

- *Example 2:* Developing a model for the boiling point (BP) of C, N, O, and H containing molecules by transfer learning from a larger (and related) melting point (MP) dataset. We use the first thousand (1000) small molecules (heavy atoms ≤ 9) from the PHYSPROP database (Mansouri et al., 2018; Agency, 2014) for BP and MP properties separately.

- *Example 3:* Developing a model for the boiling point (as in the previous example) while transfer learning from a larger heats of formation dataset (at the G4MP2 level). In this case, the BP data set was the same as in the previous example while the G4MP2 data set is the first thousand (1000) smallest C, O, N and H containing molecules from Narayanan et al. (2019).

In each case, the transfer learning model M_2' is built similar to Case 3 considered earlier, i.e. (1) assuming $P_1 \not\supseteq P_2$ and P_1 needs to be inferred from the sparse model M_1 trained on D_1, and (2) unselected features $P - \hat{P}_1$ have a relatively higher penalty λ to be included in training M_2'. Note that in Example 1, D_1 and D_2 share the same molecules with different level of properties, while in Example 2 and 3, only part of molecules in D_1 and D_2 are the same. The parent features, P are obtained by generating the pathway fingerprints of each of the molecules in both datasets using Open Babel (O'Boyle et al., 2011) and collecting all the unique paths. For a given molecule, the pathway fingerprints enumerate the atoms and the linear substructure of path length ranging from 1 to 7. For example, given the molecule "OC=N", i.e., its pathway fingerprints would be O, C, N atoms (corresponding to the path length of 1), OC, C=N groups for the path of length 2, and OC=N fragments for the path of length 3. The maximum length of pathway fingerprints is set to 7 in this work, which is the default length in most cases and works well from our experience (Li and Rangarajan, 2019). This is sufficient to allow for a rigorous identification of the potential fragments, instead of manually picking fragments as has been done in the past. The performance of M_2' is compared to model M_2 that is directly built from the parent set of features P.

The results for these examples are tabulated in Table 1, where the root mean square error (RMSE) is used to compare M_2' (transfer) and M_2 (direct). Consistent with the conceptual study, we set $m = 300$ for D_2^{tr} (200 for training, 100 for validation) in the examples and use the remainder of the data (i.e. 700 points for testing). Given that these models are linear, we note that they are not as accurate as the state-of-the-art neural network-based models (Ward et al., 2019; Yang et al., 2019a; Schütt et al., 2018). Nevertheless, being

interpretable and additive, these models offer the necessary insights about transfer learning. We can observe that in the first two examples, M_2' has a lower RMSE than M_2 indicating that transfer learning is beneficial. The better performance is intuitive given that the two tasks are closely related in both examples, i.e. properties from different levels of theories (H_f from B3LYP and G4MP2) are intrinsically correlated, while MP and BP are both thermodynamic properties involving phase change. The first example is an instance of multi-fidelity modeling via data fusion, whereby two datasets containing the same property but at two different levels of accuracy (or resolution) are combined (or fused) to build a predictive model. Typically, the dataset with a higher accuracy (or greater resolution) is smaller (i.e., it is the "D_2" set) than that with the relatively lower "fidelity" data (which is then the set "D_1"). The second example is the more typical instance of transfer learning between two related properties. In the third example, transferring information between the two properties, H_f and BP, is actually worse than directly learning a model, likely because these two properties are fundamentally different (BP is related to phase change while H_f is essentially an enthalpy).

In the conceptual study, we noticed that transfer learned model M_2' behaves better than directly learned model M_2 given a high value of ratio α between the true features P_1 and P_2, i.e., when the target task D_2 is closely related to the data-rich task D_1. We can use this finding to further understand the results of these illustrative examples.

While α could be easily computed for the conceptual study, it is hard to do so for the real datasets used in the illustrative examples as the true features are unknown. Therefore, we estimate $\hat{P}_1$ and $\hat{P}_2$ for the two datasets by developing a LASSO model on the entire dataset (i.e. splitting the 1000 data points in the ratio 0.8:0.2 for training and validation) in a manner identical to how we estimate $\hat{P}_1$ in Case 2 of the conceptual study.

The results of feature selection for the datasets is shown in Table1. Not surprisingly, the features identified for H_f from D_1 (B3LYP) and D_2 (G4MP2) have 147 out of 164 ($\hat{P}_2$) in common, which suggests a high value of α (~ 0.90). In the third example, α is 0.29, quantitatively indicating that the properties depend on fundamentally different features. These two examples clearly show the extremes of transfer learning situations. In the second example, the overlap of features is somewhere in the middle, and is likely to be a common scenario. Indeed, the value of α for this example is 0.45; while this is not high, we still see that transfer learning leads to better performance. To understand this better, we identified the most important features in $\hat{P}_2$ (by selecting only those features whose absolute weights are larger than the average of all features in the set). There were 58 features in this subset, of which 29 also exist in $\hat{P}_1$. By only considering the important features, $\alpha = 0.50$ which plausibly explains the observed positive effect of transfer learning.

The correlation between estimated α and the benefit from transfer learning in the three examples suggests a simple empirical way to determine if two properties are related, in the absence of prior domain knowledge. In particular, simple linear sparse models, e.g. using LASSO, can be easily developed using available data to determine if there is sufficient overlap of features, even if subsequently data-driven features are learnt (e.g. from convolutional neural networks). The feature overlap can, in turn, provide a qualitative determination, *a priori*, about whether or not transfer learning will work.

In summary, transfer learning requires a large overlap of the underlying features; in particular, we require that $\alpha \geq 0.5$. Since knowing α *a priori* is hard, one may have to rely on domain expertise to judge if two tasks are related. This is easy to judge in the context of multi-fidelity modeling where the comparison is between two different ways of measuring the same property but is expected to be harder while comparing physical, chemical, or biological properties. We, therefore, propose that approaches similar to Case 3 are the most efficient. In general, transfer learning should allow for the consideration of a larger space of features and model structures (especially for more sophisticated nonlinear models) but concurrently place higher emphasis (e.g. through larger weights) on information that is being transferred from the related task. In principle, for other problems, especially nonlinear examples, a similar conclusion could be inferred where there is a certain value/range of α that makes features in two tasks related enough for applying transfer learning. However, a similar study is required to determine the value of α and our quantitative result (i.e. the overlap of 50%) cannot be generalized to them without a systematic analysis. We finally Note that transferred information could potentially be used in active learning, especially in a diversity-maximizing exploratory step to identify a broad yet compact collection of molecules to be included in the training set (Li and Rangarajan, 2019).

3.3. Conceptual study of transfer learning for nonlinear models

We further extend our conceptual study to a nonlinear model built with a neural network. We create a synthetic dataset with a nonlinear relationship between inputs and outputs, then follow a similar process to the conceptual case studies for the linear models to evaluate the effect of transfer learning. Basically, we apply a polynomial expansion on the original data matrix to introduce nonlinearity into the dataset, while using a neural network to train the model and transfer the feature knowledge from M_1 to M_2' in the form of weights of M_1's first layer. We assume a complete overlap of features to investigate the extreme scenario of whether M_2' by learning the important features from M_1. Our results show that compared to the directly learned model, M_2, whose weights were initialized randomly, transfer learning the first layer weights (parameters) from M_1 to M_2' leads to better performance both on the training set as well as the test set. The details of this exercise can be found in the supporting information (SI).

4. Conclusions

Using simplified, interpretable linear models, we carried out a conceptual study to understand the requirements for transfer learning to be favorable while building accurate and generalizable molecular property models with scarce data. Through three case studies, we find that a substantial overlap of features (i.e. $\geq 50\%$ of features being common) is essential for transfer learning to be beneficial. We subsequently considered three illustrative examples wherein we showed that (i) heats of formation at a higher level of theory (G4MP2) can be better learned using transferred information (salient features) from a relatively low level of theory (B3LYP) because the estimated feature overlap was 90%; (ii) learning a boiling point

model will be negatively impacted if information is transferred from a model of heats of formation model because their estimated feature overlap was less than 30%; and (iii) melting point data can be used to learn a better boiling point model because the estimated overlap was 45 - 50%. In the absence of information regarding the salient features (P_1) of the related (data-rich) task, it appears that the best approach to train a model for the data-poor target task is the third case considered in our study. By doing this, we allow for all plausible features P to be included in the machine-learned model but place a greater emphasis on those features that were identified (via LASSO) to be important for the related task. These insights are also useful while developing neural network models; indeed, the third case study suggests some form of mild or regularized fine-tuning will be beneficial while reusing the transferred neural network layers of the model of a related task.

5. Supplementary material

The python codes for the three cases of the conceptual study and the three illustrative examples, as well as the nonlinear model study are provided in the supplementary material.

6. Declaration of Competing Interest

The authors have no conflict of interest.

7. Author statement

Bowen Li: Data curation, Formal analysis, Methodology, Writing - original draft.

Srinivas Rangarajan: Conceptualization, Methodology, Funding acquisition, Supervision, Writing - review & editing.

8. Acknowledgement

SR acknowledges partial financial support from Lehigh University and the National Science Foundation (CBET-1953245). Portions of this research were conducted on Lehigh University's Research Computing infrastructure partially supported by the NSF Award 2019035.

References

Agency, U.S.E.P., 2014. Us epa(2014) epi suite data. URL: https://www.epa.gov/tsca-screening-tools/epi-suitetm-estimation-program-interface.

Bartók, A.P., De, S., Poelking, C., Bernstein, N., Kermode, J.R., Csányi, G., Ceriotti, M., 2017. Machine learning unifies the modeling of materials and molecules. Science advances 3, e1701816.

Bird, J.J., Faria, D.R., Ekárt, A., Ayrosa, P.P., 2020. From simulation to reality: Cnn transfer learning for scene classification, in: 2020 IEEE 10th International Conference on Intelligent Systems (IS), IEEE. pp. 619–625.

Bogojeski, M., Vogt-Maranto, L., Tuckerman, M.E., Müller, K.R., Burke, K., 2020. Quantum chemical accuracy from density functional approximations via machine learning. Nature communications 11, 1–11.

Cai, C., Wang, S., Xu, Y., Zhang, W., Tang, K., Ouyang, Q., Lai, L., Pei, J., 2020. Transfer learning for drug discovery. Journal of Medicinal Chemistry 63, 8683–8694.

Christensen, A.S., Bratholm, L.A., Faber, F.A., Anatole von Lilienfeld, O., 2020. Fchl revisited: Faster and more accurate quantum machine learning. The Journal of chemical physics 152, 044107.

Coley, C.W., Barzilay, R., Green, W.H., Jaakkola, T.S., Jensen, K.F., 2017. Convolutional embedding of attributed molecular graphs for physical property prediction. Journal of chemical information and modeling 57, 1757–1772.

Cova, T.F., Pais, A.A., 2019. Deep learning for deep chemistry: optimizing the prediction of chemical patterns. Frontiers in chemistry 7, 809.

Delaney, J.S., 2004. Esol: estimating aqueous solubility directly from molecular structure. Journal of chemical information and computer sciences 44, 1000–1005.

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.P., 2015. Convolutional networks on graphs for learning molecular fingerprints. arXiv preprint arXiv:1509.09292 .

Gentile, F., Agrawal, V., Hsing, M., Ton, A.T., Ban, F., Norinder, U., Gleave, M.E., Cherkasov, A., 2020. Deep docking: a deep learning platform for augmentation of structure based drug discovery. ACS central science 6, 939–949.

Gómez-Bombarelli, R., Wei, J.N., Duvenaud, D., Hernández-Lobato, J.M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T.D., Adams, R.P., Aspuru-Guzik, A., 2018. Automatic chemical design using a data-driven continuous representation of molecules. ACS central science 4, 268–276.

Grambow, C.A., Li, Y.P., Green, W.H., 2019. Accurate thermochemistry with small data sets: a bond additivity correction and transfer learning approach. The Journal of Physical Chemistry A 123, 5826–5835.

Han, H., Choi, S., 2021. Transfer learning from simulation to experimental data: Nmr chemical shift predictions. The Journal of Physical Chemistry Letters 12, 3662–3668.

Hansen, K., Biegler, F., Ramakrishnan, R., Pronobis, W., Von Lilienfeld, O.A., Muller, K.R., Tkatchenko, A., 2015. Machine learning predictions of molecular properties: Accurate many-body potentials and nonlocality in chemical space. The journal of physical chemistry letters 6, 2326–2331.

He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., Li, M., 2019. Bag of tricks for image classification with convolutional neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 558–567.

Hegde, R.S., 2020. Deep learning: a new tool for photonic nanostructure design. Nanoscale Advances 2, 1007–1023.

Iovanac, N.C., Savoie, B.M., 2020. Simpler is better: How linear prediction tasks improve transfer learning in chemical autoencoders. The Journal of Physical Chemistry A 124, 3679–3685.

Jablonka, K.M., Ongari, D., Moosavi, S.M., Smit, B., 2020. Big-data science in porous materials: materials genomics and machine learning. Chemical reviews 120, 8066–8129.

Janet, J.P., Kulik, H.J., 2017. Predicting electronic structure properties of transition metal complexes with neural networks. Chemical science 8, 5137–5152.

Kan, M., Wu, J., Shan, S., Chen, X., 2014. Domain adaptation for face recognition: Targetize source domain bridged by common subspace. International journal of computer vision 109, 94–109.

Kim, J., Park, C., 2017. End-to-end ego lane estimation based on sequential transfer learning for self-driving cars, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 30–38.

Kollmann, H.T., Abueidda, D.W., Koric, S., Guleryuz, E., Sobh, N.A., 2020. Deep learning for topology optimization of 2d metamaterials. Materials & Design 196, 109098.

Korkmaz, S., 2020. Deep learning-based imbalanced data classification for drug discovery. Journal of Chemical Information and Modeling 60, 4180–4190.

Lee, J., Asahi, R., 2021. Transfer learning for materials informatics using crystal graph convolutional neural network. Computational Materials Science 190, 110314.

Lentelink, N.J., Palkovits, S., 2020. Transfer learning as tool to enhance predictions of molecular properties based on 2d projections. Advanced Theory and Simulations 3, 2000148.

Li, B., Rangarajan, S., 2019. Designing compact training sets for data-driven molecular property prediction

through optimal exploitation and exploration. Molecular Systems Design & Engineering 4, 1048–1057.
Li, F., Qiao, H., Zhang, B., 2018. Discriminatively boosted image clustering with fully convolutional auto-encoders. Pattern Recognition 83, 161–173.
Lubbers, N., Smith, J.S., Barros, K., 2018. Hierarchical modeling of molecular energies using a deep neural network. The Journal of chemical physics 148, 241715.
Mansouri, K., Grulke, C.M., Judson, R.S., Williams, A.J., 2018. Opera models for predicting physicochemical properties and environmental fate endpoints. Journal of cheminformatics 10, 1–19.
Mater, A.C., Coote, M.L., 2019. Deep learning in chemistry. Journal of chemical information and modeling 59, 2545–2559.
Meyer, B., Sawatlon, B., Heinen, S., von Lilienfeld, O.A., Corminboeuf, C., 2018. Machine learning meets volcano plots: computational discovery of cross-coupling catalysts. Chemical science 9, 7069–7077.
Narayanan, B., Redfern, P.C., Assary, R.S., Curtiss, L.A., 2019. Accurate quantum chemical energies for 133000 organic molecules. Chemical science 10, 7449–7455.
Nikolaev, P., Hooper, D., Webber, F., Rao, R., Decker, K., Krein, M., Poleski, J., Barto, R., Maruyama, B., 2016. Autonomy in materials research: a case study in carbon nanotube growth. npj Computational Materials 2, 1–6.
O'Boyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T., Hutchison, G.R., 2011. Open babel: An open chemical toolbox. Journal of cheminformatics 3, 1–14.
Peng, Y., Yan, S., Lu, Z., 2019. Transfer learning in biomedical natural language processing: an evaluation of bert and elmo on ten benchmarking datasets. arXiv preprint arXiv:1906.05474 .
Raj, R.J.S., Shobana, S.J., Pustokhina, I.V., Pustokhin, D.A., Gupta, D., Shankar, K., 2020. Optimal feature selection-based medical image classification using deep learning model in internet of medical things. IEEE Access 8, 58006–58017.
Ramakrishnan, R., Dral, P.O., Rupp, M., Von Lilienfeld, O.A., 2014. Quantum chemistry structures and properties of 134 kilo molecules. Scientific data 1, 1–7.
Ruddigkeit, L., Van Deursen, R., Blum, L.C., Reymond, J.L., 2012. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. Journal of chemical information and modeling 52, 2864–2875.
Rusu, A.A., Večerík, M., Rothörl, T., Heess, N., Pascanu, R., Hadsell, R., 2017. Sim-to-real robot learning from pixels with progressive nets, in: Conference on Robot Learning, PMLR. pp. 262–270.
Ryan, K., Lengyel, J., Shatruk, M., 2018. Crystal structure prediction via deep learning. Journal of the American Chemical Society 140, 10158–10168.
Schutt, K., Kessel, P., Gastegger, M., Nicoli, K., Tkatchenko, A., Muller, K.R., 2018. Schnetpack: A deep learning toolbox for atomistic systems. Journal of chemical theory and computation 15, 448–455.
Schütt, K.T., Sauceda, H.E., Kindermans, P.J., Tkatchenko, A., Müller, K.R., 2018. Schnet–a deep learning architecture for molecules and materials. The Journal of Chemical Physics 148, 241722.
Smith, J.S., Nebgen, B.T., Zubatyuk, R., Lubbers, N., Devereux, C., Barros, K., Tretiak, S., Isayev, O., Roitberg, A.E., 2019. Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning. Nature communications 10, 1–8.
Turki, T., Wei, Z., Wang, J.T., 2017. Transfer learning approaches to improve drug sensitivity prediction in multiple myeloma patients. IEEE Access 5, 7381–7393.
Unke, O.T., Meuwly, M., 2019. Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges. Journal of chemical theory and computation 15, 3678–3693.
Ward, L., Blaiszik, B., Foster, I., Assary, R.S., Narayanan, B., Curtiss, L., 2019. Machine learning prediction of accurate atomization energies of organic molecules from low-fidelity quantum chemical calculations. MRS Communications 9, 891–899.
Westermayr, J., Marquetand, P., 2020. Machine learning for electronically excited states of molecules. Chemical Reviews .
Xing, Y., Tang, J., Liu, H., Lv, C., Cao, D., Velenis, E., Wang, F.Y., 2018. End-to-end driving activities and secondary tasks recognition using deep convolutional neural network and transfer learning, in: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 1626–1631.

Xue, D., Balachandran, P.V., Hogden, J., Theiler, J., Xue, D., Lookman, T., 2016. Accelerated search for materials with targeted properties by adaptive design. Nature communications 7, 1–9.

Yamada, H., Liu, C., Wu, S., Koyama, Y., Ju, S., Shiomi, J., Morikawa, J., Yoshida, R., 2019. Predicting materials properties with little data using shotgun transfer learning. ACS central science 5, 1717–1730.

Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., et al., 2019a. Analyzing learned molecular representations for property prediction. Journal of chemical information and modeling 59, 3370–3388.

Yang, S., Huang, G., Cai, B., 2019b. Discovering topic representative terms for short text clustering. IEEE Access 7, 92037–92047.

Yao, L., Mao, C., Luo, Y., 2019. Graph convolutional networks for text classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 7370–7377.

Zubatyuk, R., Smith, J.S., Leszczynski, J., Isayev, O., 2019. Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network. Science advances 5, eaav6490.