

# Tackling Data Scarcity Challenge through Active Learning in Materials Processing with Electrospray

Fanjin Wang, Anthony Harker, Mohan Edirisinghe, and Maryam Parhizkar\*

Machine learning (ML) has been harnessed as a promising modelling tool for materials research. However, small data, or data scarcity, is a bottleneck when incorporating ML in studies involving experimentation. Current experiment planning methods show several disadvantages: one-factor-at-a-time (OFAT) experimentation became impractical due to limited laboratory resources; conventional design of experiments (DoE) failed to incorporate high-dimensional features in ML; Surrogate-based or Bayesian optimization (BO) shifted the goal to optimize material properties rather than guiding training data accumulation. The present research proposes leveraging active learning (AL) to strategically select critical data for experimentation. Two AL strategies, query-by-Committee (QBC) algorithm and Greedy method, are benchmarked against random query baseline on various materials datasets. AL is shown to efficiently reduce model prediction errors with minimal additional experiment data. Investigation of hyperparameters revealed benefits of applying AL at an early stage of experimental dataset construction. Moreover, AL is implemented and validated for an in-house materials development task - electrospray modelling. AL exploration as a paradigm is highlighted to guide experiment design for efficient data accumulation purposes, and its potential for further ML modelling. In doing so, the power of ML is expected to be fully unleashed to experimental researchers.

## 1. Introduction

Recent advancements in artificial intelligence (AI) have demonstrated its disruptive potential in knowledge extraction from data.<sup>[1,2]</sup> Spreading its influence beyond the realm of diffusion models and large language models, ML excels in modelling experimental research, where numerous variables influence the outcome. It has gained favor among researchers as a powerful tool for prediction as well as inference in a wide spectrum of areas, such as biology, pharmaceuticals, and materials science.<sup>[3-8]</sup> Regardless of field, the significant role of the training data is widely acknowledged, as it forms the foundation of any model and establishes an upper limit on model performance. Many current AI applications rely on extensive datasets, exemplified by repositories like ImageNet, housing over a million images, and terabytes of training data required for the GPT-4 model.<sup>[9,10]</sup> Adequate training data is crucial to achieve satisfactory prediction accuracy and to derive reliable insights.


In the area of experimental research, initiatives such as the Materials Project and Protein Data Bank (PDB) have been launched to facilitate data accumulation, yielding promising and fruitful results.<sup>[11,12]</sup> Nevertheless, due to the unique nature of experimental studies, researchers often encounter challenges in obtaining datasets tailored for their research questions. In such cases, datasets must be painstakingly curated through resource-demanding in-house laboratory experiments, which has deterred many attempts to implement data-driven modeling and gave rise to the data scarcity bottleneck.<sup>[13]</sup>

To address the issue of data scarcity in experiments, a methodological approach emerged as a potential solution. Traditional experiment planning often involved designating one factor as the independent variable while holding all others constant, a one-factor-at-a-time (OFAT) approach.<sup>[14]</sup> However, given the vast variable space, such OFAT experiments were impractical in providing sufficient information for data-driven modelling.<sup>[15]</sup> As an improvement, design of experiments (DoE) aimed to systematically plan experiments to obtain optimal information.<sup>[14,16]</sup> DoE has found extensive application in pharmaceuticals, materials science, and engineering research for comprehensively assessing the impact of multiple variables on responses.<sup>[15,17]</sup> Typically, DoE results were analyzed using techniques like analysis of variance (ANOVA) and response surface methods (RSM).

F. Wang, M. Edirisinghe  
Department of Mechanical Engineering  
University College London  
London WC1E 7JE, UK

A. Harker  
Department of Physics and Astronomy  
University College London  
London WC1E 6BT, UK

M. Parhizkar  
School of Pharmacy  
University College London  
London WC1N 1AX, UK  
E-mail: maryam.parhizkar@ucl.ac.uk

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/aisy.202300798>.

© 2024 The Authors. Advanced Intelligent Systems published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: 10.1002/aisy.202300798

More recently, ML methods have been introduced to the modeling of DOE experiment results.<sup>[18]</sup> Nevertheless, DoE methods demand well-structured experiment plans that must be strictly followed. This rigidity forbids any changes during the experiment process. Furthermore, current DOE methods, including Taguchi orthogonal design, Central Composite Designs (CCD), and Box-Behnken (BB) Designs, impose strict constraints on the number of variables and levels.<sup>[14,19]</sup> For example, CCD and BB were developed for evaluating three-level variables, while Taguchi orthogonal design struggles with experiments involving more than five variables, each with five levels.<sup>[19,20]</sup> These limitations have undermined the utility of DoE as a tool for experiment planning in the ML data accumulation phase.

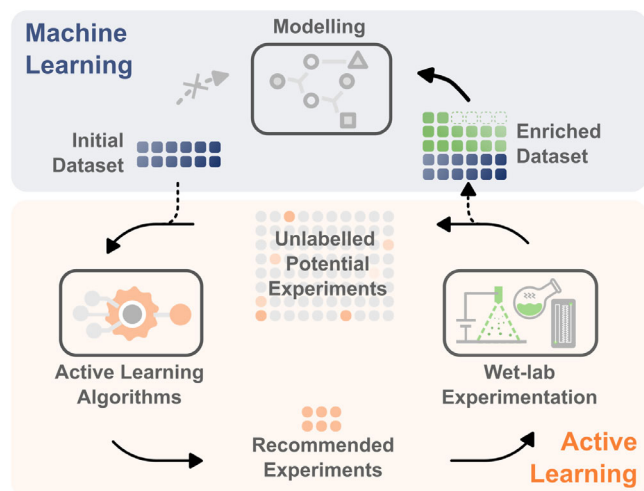
Therefore, a more advanced and flexible experiment design paradigm is highly desired to allow researchers to grasp the power of ML. Here, we introduce the concept of active learning (AL), a field in human-in-the-loop learning. Settle succinctly described AL as a recommendation system to propose unlabeled instances to be labeled by an oracle.<sup>[21]</sup> This approach is particularly valuable in situations where the cost of labeling data is significant, like hiring radiologists to identify cancer sites.<sup>[22,23]</sup> Recommendations given by AL are designed to prioritize critical instances and thus help maximize the information gained through labelling. From our viewpoint as researchers, such capacities of AL could assist researchers to allocate limited resources in laboratory. In such context, researchers become the oracle in AL, and the labelling process is essentially experimentation. As illustrated in **Figure 1**, the insufficient initial dataset led to poor modeling performance. AL's role was to strategically select potential experiments, followed by researchers performing laboratory experiments. With a few rounds of AL, the enriched dataset would allow better modeling results.

The majority of AL strategies proposed in the literature had a primary focus on the classification tasks. One prominent strategy, known as the query-by-committee (QBC) method, was put forward by Seung et al. in 1992 and further developed by Abe and Mamitsuka.<sup>[24–26]</sup> In QBC, each committee member votes for the potential class of an unlabeled instance. The

recommendation strategy is based on the maximum disagreement, which is calculated by the entropy between these votes. However, AL for classification is less applicable to experimental research, where most of variables are continuous (e.g., materials properties and processing parameters). Efforts to incorporate AL algorithms into regression expanded AL's potential for broader applications. Burbidge et al. derived the theoretical grounding for the regression version of QBC algorithm, where the disagreement of a committee was measured by the standard deviation of predictions.<sup>[27]</sup> Other algorithms like expected model change maximization (EMCM) were introduced later to use the derivation of the loss function during training as the selection criteria for recommendations.<sup>[28]</sup>

More recently, there is a growing interest in leveraging *adaptive learning* strategies to assist experiment planning in materials research. Under the umbrella of *adaptive learning*, AL and BO are both goal-driven learning strategies with different focuses.<sup>[29]</sup> BO, or more broadly surrogate-based optimization (SBO), placed emphasis on the identification of optimum candidates.<sup>[30]</sup> SBO algorithms seek balance of exploration and exploitation to avoid being trapped in local minimum. For example, Pandi et al. developed an SBO framework and tested it on various biological network optimization tasks.<sup>[31]</sup> Furthermore, in the field of materials research, Lookman et al. performed an extensive review on how these optimization strategies could facilitate novel material discovery.<sup>[32]</sup> In contrast, AL placed focus on exploration to minimize the uncertainty in the training dataset. Rodríguez-Pérez et al. used entropy-based AL strategy for improving classification accuracy of kinase inhibitor binding types.<sup>[33]</sup> Li et al. recently examined AL using prediction probability calculated by surrogates with chemistry datasets.<sup>[34]</sup> Both studies revealed the potential of AL to construct a refined dataset for training purposes. However, only limited studies investigated applications of AL for regression in materials discovery and development. Also, key hyperparameters in AL were not fully understood in a real-life experiment planning setting.

Current bottleneck in lacking enough training data made it extremely difficult to obtain ML models with satisfactory performance for prediction and inference. Such challenges are especially severe in experiment-based materials development, where simulation tools (e.g., *ab initio* calculations) are less available compared to materials discovery tasks. In the present study, we propose implementing AL as a generalizable tool for experiment designing that could strategically allocate limited experiment budget to critical datapoints. As an improvement on prior arts, we conducted a comprehensive evaluation, benchmarking the performance of two AL algorithms—namely, the QBC method and the Greedy method—against a baseline by random query. The evaluation was performed on a diverse range of datasets, spanning materials development (including processing and engineering tasks) as well as materials discovery (related to quantitative structure-property relationship, or QSPR, tasks) which involved complicated high-dimensional inputs. Furthermore, hyperparameters in AL algorithms were examined to shed light on their impact. In addition, we showcased AL on our in-house datasets, using them to recommend wet-lab experiments for data-driven modeling. We anticipate that this new experiment design paradigm to achieve previously unattainable tasks. For example, removing constraints of features in



**Figure 1.** A schematic illustration of the AL process to supply additional data to the original scarce dataset for ML training.

traditional DoE, receiving high-dimensional inputs like molecular descriptors, suggesting crucial data to tackle data scarcity in ML modeling, and eventually boosting ML applications for research in general.

## 2. Results

### 2.1. AL on Published Datasets

To evaluate the performance of AL on material datasets, we first benchmarked two AL strategies, the Greedy method and QBC algorithm, against a baseline method random query. The datasets were selected to represent various tasks in material research. For example, the concrete compressive strength dataset possessed 8 input variables detailing concrete preparation, with the compressive strength yielded from the corresponding parameters. Another example was the ESOL dataset where aqueous solubilities were correlated to molecular structures, comprising a QSPR materials dataset. In this retrospective setting of AL, each full dataset was randomly divided into two sets, a starting set and a pool set, with the aim to resemble the scenario of AL applications with limited amount of data as the starting set. The improvement in performance, represented by the reduction in root mean squared error (RMSE), was plotted for the five datasets (Figure 2a).

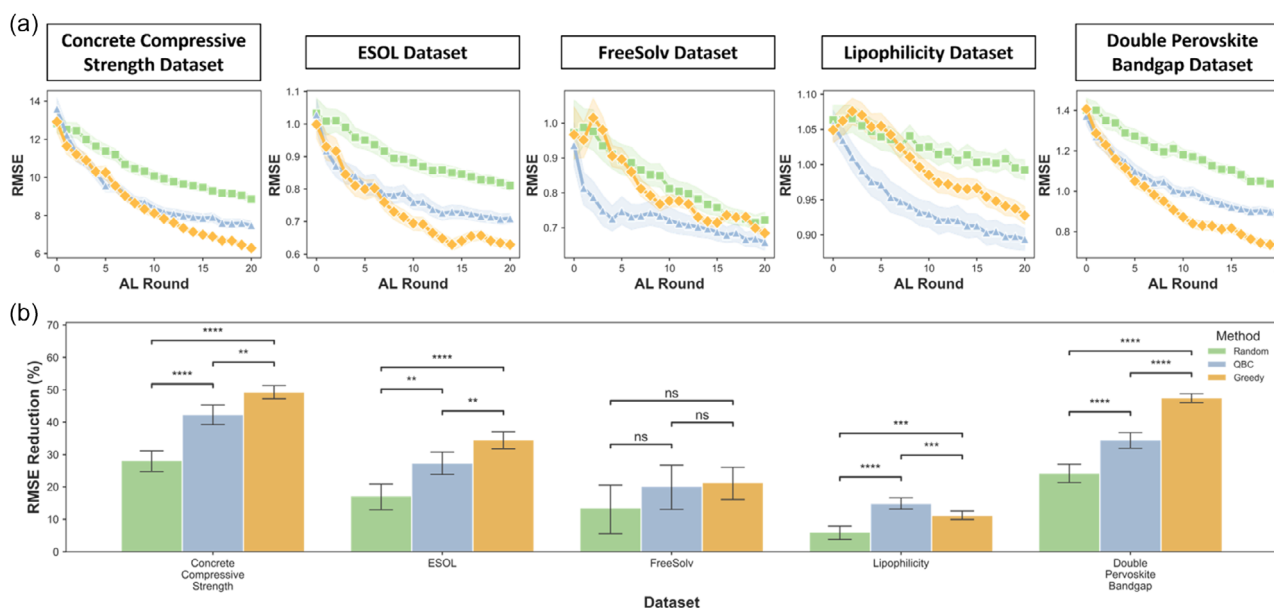
An instant observation was that, regardless of AL algorithms, model performance consistently improved as more training data became available, while some minor fluctuations were noted (e.g., the first few rounds with the FreeSolv and lipophilicity datasets). This coincided with the common observation that model performance would benefit from more data.<sup>[35]</sup> For the “bumping” jitters that occurred during AL, a similar observation

was reported by Faulds et al. where the authors explained it with local minimum during the training process.<sup>[36]</sup>

Furthermore, it was observed that the QBC and Greedy methods outperformed the random query baseline after 20 AL rounds in all datasets (Table S1, Supporting Information). The kinetics of RMSE reduction for both methods were considerably faster than the random baseline. The percentage reduction in RMSE before and after AL was used as an indicator of AL algorithm performance, as illustrated in the bar plots in Figure 2b. Both QBC and Greedy methods led to more significant ( $p < 0.01$ ) reduction of RMSE compared with random baseline except the FreeSolv dataset ( $p = 0.09$  for QBC and  $p = 0.21$  for Greedy). Taking a step further, such observations were still prominent even in the initial rounds of AL, as reflected by Figure 2a. The benchmarking results suggested that AL algorithms like QBC, or Greedy methods were able to suggest potential experiments that more effectively improve ML modeling than random baseline. This was in agreement with a previous benchmarking work on text labelling tasks.<sup>[37]</sup>

### 2.2. Impact of AL Parameters

The impact of parameters on AL performance was further explored, which included the hyperparameters in QBC algorithm and two global parameters in AL. In the benchmark experiment, hyperparameters for the QBC algorithm were selected based on a previous study.<sup>[28]</sup> Here, we examined the number of committee members and the type of committee members. It was observed that there was only a trivial effect on model performance when varying committee sizes (Figure S1a, Supporting Information). For the type of committee members, a limited influence could be seen based on results of three datasets (Figure S1b, Supporting Information). Only with ESOL dataset, committee



**Figure 2.** a) Model performance improvement by implementing AL with different strategies (green squares: Random, blue triangles: QBC, and orange diamonds: Greedy). Shaded areas show 95% confidence interval ( $N = 100$ ). b) Percentage reduction of RMSE after 20 rounds of AL. Error bars show 95% confidence interval ( $N = 100$ ). (Annotations: ns:  $p > 0.05$ , \*:  $p < 0.05$ , \*\*:  $p < 0.01$ , \*\*\*:  $p < 0.001$ , \*\*\*\*:  $p < 0.0001$ ).

D exhibited leading performance. These results suggested that even a small committee of two or three members in QBC algorithm exhibited a satisfactory performance. Some reports also support our observation.<sup>[26,37]</sup>

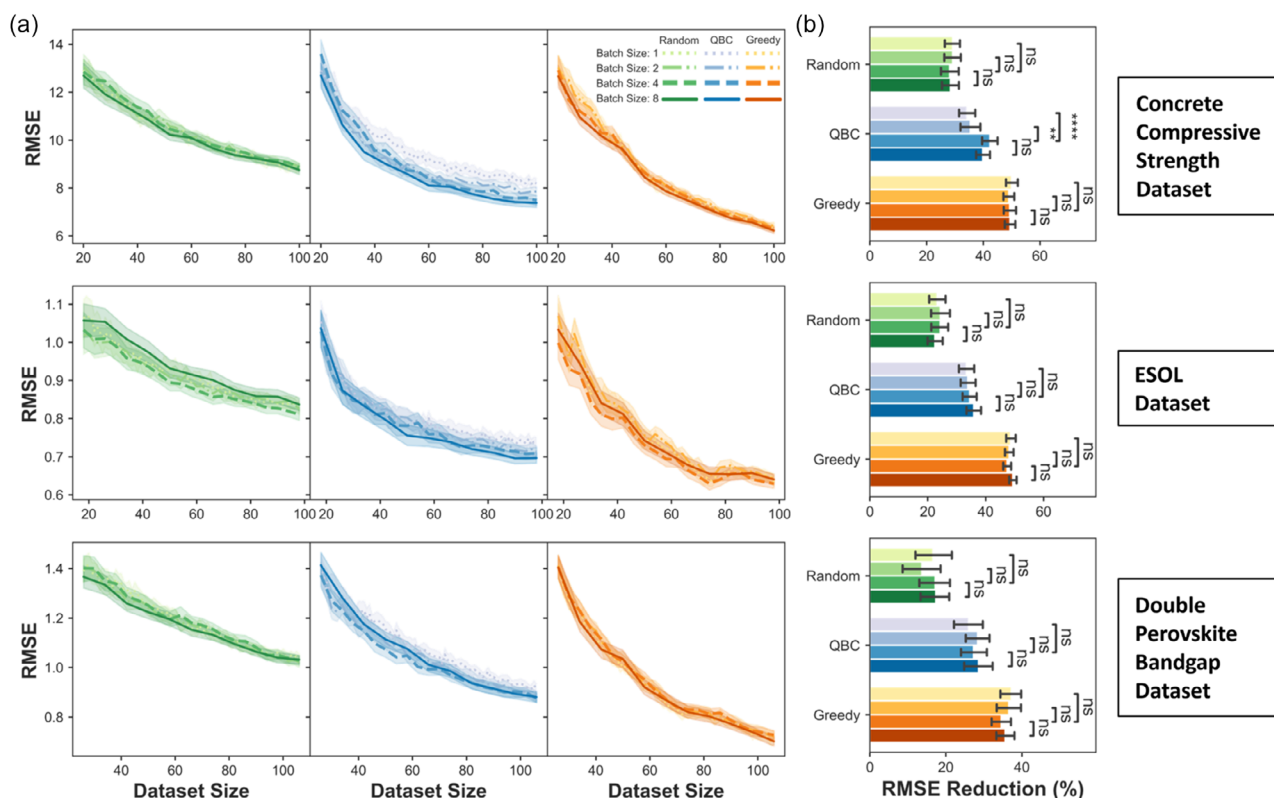
Furthermore, two global parameters in AL were inspected. One was the batch size, which determined the number of instances to be recommended in each round. The other was the starting ratio, which controlled the percentage of data allocated to the starting set as opposed to the pool set. As depicted in **Figure 3a**, varying batch size from the default 4 to 1, 2, or 8 did not result in significant differences in algorithm performance for the random and Greedy methods. However, it is worth noting that in the concrete dataset, increasing the batch size was beneficial to the performance of the QBC method. For other datasets, QBC's performance remained consistent regardless of batch size (**Figure 3b**).

We then considered the starting ratio, which referred to the proportion during the partitioning of the full dataset into two parts: the starting set and the pool set. Intuitively, the ratio determined the level of prior information that was possessed before the implementation of AL. Thus, it would be easy to imagine that the reduction of RMSE would be more prominent when supplying additional data to an extremely deficient dataset, compared with a "saturated" dataset. This assumption was confirmed in our experiments (**Figure 4**). The reduction of RMSE was more evident across all methods when the starting data size was the

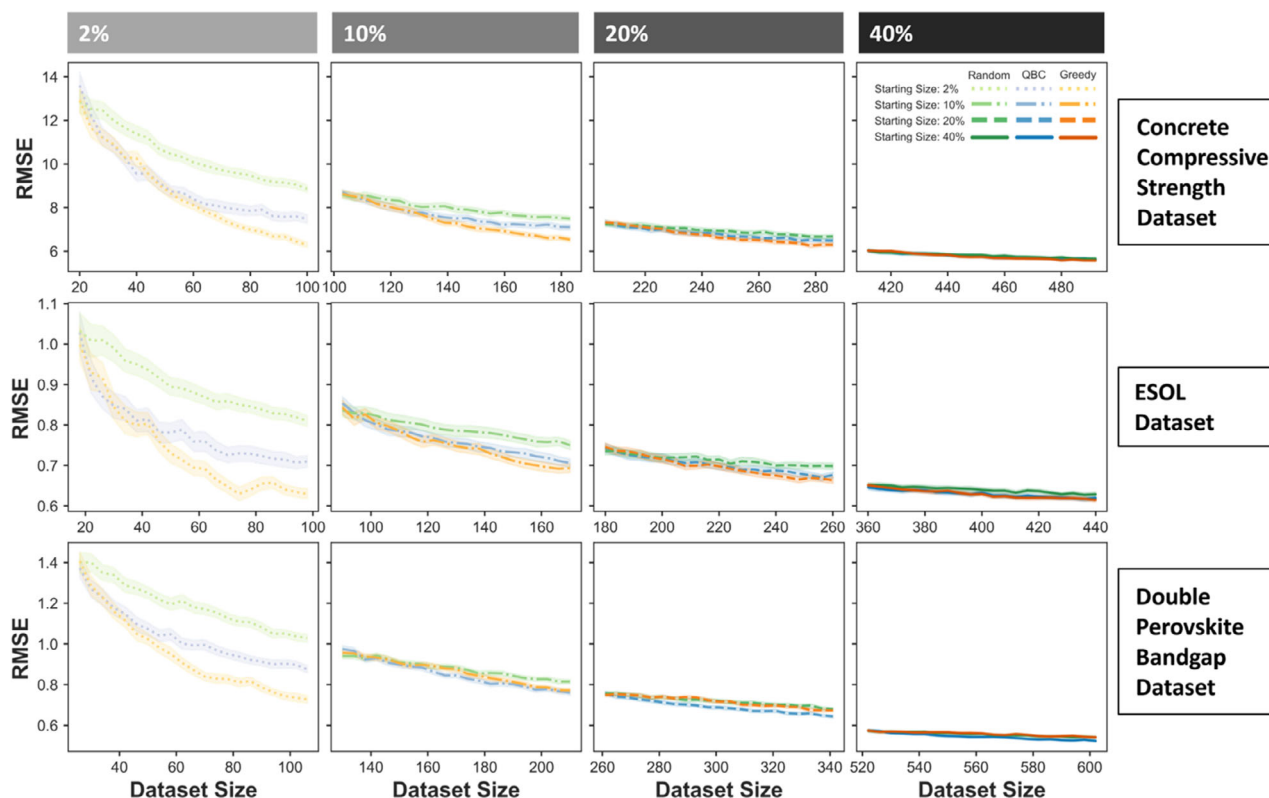
smallest (2%). Additionally, it showed that the QBC and greedy method consistently outperformed the baseline in the first two ratios 2% and 10%. This suggested that researchers would benefit more from implementing AL at an early stage.

### 2.3. AL on in-House Datasets

The following section will shift the focus toward evaluating AL within our in-house problem setting - electrospray particle generation. To provide some context, electrospray (also called electrohydrodynamic atomization) is a versatile micro-/nano-manufacturing technology employed for the controlled breakdown of a liquid jet into droplets utilizing an electric field.<sup>[38]</sup> When a volatile solvent is used during the process, it evaporates, resulting in the formation of solid fine particles (**Figure 5a**). Electrospray as a technology stands out for its ability to generate significantly smaller droplets with tightly controlled size distributions, making it an attractive candidate for fine particle preparation, particularly in pharmaceutical, analytical chemistry, and energy applications.<sup>[39,40]</sup> Nonetheless, harnessing the versatility offered by electrospray technology was far from straightforward. The characteristics of the generated particles, such as size distribution and morphology, were governed by a complex interplay of various parameters. Moreover, characterization presented a challenge because the sizes of electrospray-generated particles were typically below a few micrometers. In practice, the optimization



**Figure 3.** a) Model performance improvement with varying AL batch size settings and AL algorithms (green: Random; blue: QBC; and orange: Greedy) on three datasets. Shaded areas show 95% confidence interval ( $N = 100$ ). b) Percentage reduction of RMSE after querying 80 instances. Error bars show 95% confidence interval ( $N = 100$ ). Bar plots are color-shaded based on batch size (from light to dark: 1, 2, 4, and 8). (Statistical testing: Mann-Whitney-Wilcoxon two-tailed U test. Annotations: ns:  $p > 0.05$ , \*\*:  $p < 0.01$ , and \*\*\*\*:  $p < 0.0001$ ).



**Figure 4.** Model performance improvement with varying starting set size and AL algorithms (green: Random; blue: QBC; and orange: Greedy) on three datasets. Starting data size ranged from 2%, 10%, 20%, and 40% of the corresponding full dataset. Y-axes are shared between plots in the same row. Shaded areas show 95% confidence interval ( $N = 100$ ).

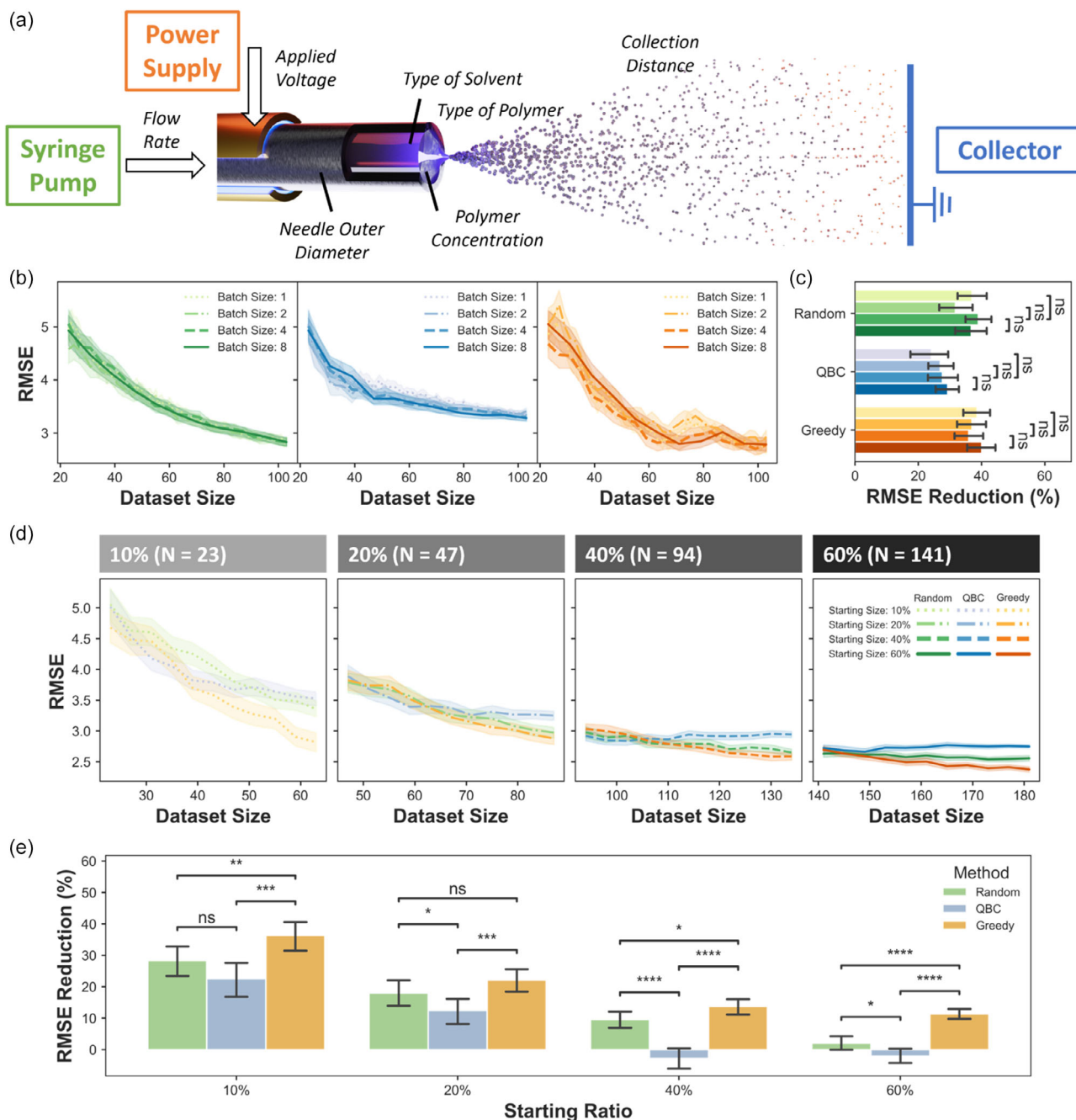
process required excessive trial-and-error and prior experience. Thus, it was of great interest to leverage data-driven modeling to understand the complicated relationships in electrospray.

Considering the similarity between the concrete dataset and the electrospray dataset, both being processing datasets, the experiment of batch size was repeated (Figure 5b,c). Unlike the case in the concrete dataset, here, no significant impact on AL performance could be seen. Furthermore, the starting dataset size was varied from 10%, 20%, 40%, and up to 60% of the full poly(lactic-co-glycolic acid (PLGA) dataset (Figure 5d,e). Results from the random method revealed the diminishing benefit of model performance with additional data. Starting with 23 instances, the RMSE reduced by 28.4% after 10 rounds of AL. In contrast, the reduction was 18.0% for the 47 instances (20%) group. For the remaining two groups, the reduction of RMSE after 10 rounds of AL dropped to 9.58% (the 40% group) and merely 2.00% (the 60% group).

Notably, the QBC algorithm yielded unsatisfactory results on the PLGA dataset. The reduction in RMSE with the baseline method was significantly more than that with QBC in most starting size ratios, including 20%, 40%, and 60%. In Figure 5d, it appeared that the QBC method reached a plateau and was unable to further reduce the RMSE after reaching around 80 datapoints. It was assumed that the poor performance of the QBC algorithm could be related to the nature of this literature-originated dataset. Experiments extracted from various papers, as opposed to a

well-designed experiment space (e.g., in the concrete strength dataset), may deteriorate QBC algorithm's performance. This issue will be further elaborated in the next section with PCL dataset. In summary, datapoints collected from different papers would form clusters instead of an evenly distributed space. In a retrospective setting, a portion of these clusters would be masked out to simulate a potential experiment space. Furthermore, it was limiting what could be "queried" from the pool set. In published datasets tested above, the QBC algorithm had a better performance than the baseline, which may be related to their larger and more evenly distributed pool sets. This problem, however, was less of a concern for implementing AL prospectively, where the potential experiments were usually generated from a set of variables and levels. An encouraging observation was that the Greedy method outperformed the baseline, prompting us to evaluate these three methods in wet-experiment scenarios.

After completing retrospective AL on our in-house PLGA dataset, we attempted putting AL algorithms in practice (prospectively) to guide laboratory experiments. The PCL dataset had a much smaller dataset with 113 data points, giving a poor performance in preliminary modelling attempts (Figure S2, Supporting Information). An XGBoost model with default hyperparameters yielded an RMSE of 3.47, a mean absolute percentage error (MAPE) of 21% and an  $R^2$  of 0.55 under 6-fold cross-validation. It was evident that there was room for



**Figure 5.** a) A close-up illustration of an electrospaying needle. The outer electrode (gold) is connected to a high-voltage power supply. The inner metal needle runs the polymer solution fed from a syringe pump. A grounded metal board is placed in a distance as the collector for particles. Processing parameters are listed in italic. b) Model performance improvement with varying AL batch size settings and AL algorithms on PLGA dataset. Shaded areas show 95% confidence interval ( $N = 100$ ). c) Percentage reduction of RMSE after 80 instances recommended. Error bars show 95% confidence interval ( $N = 100$ ). Bar plots are color-shaded based on batch size (from light to dark: 1, 2, 4, and 8). d) Model performance improvement with varying starting set size and AL algorithms. e) Percentage reduction of RMSE after 20 rounds of AL with different starting dataset size. Error bars show 95% confidence interval ( $N = 100$ ). (Annotations: ns:  $p > 0.05$ , \*:  $p < 0.05$ , \*\*:  $p < 0.01$ , and \*\*\*\*:  $p < 0.0001$ ).

improvement compared to the ML model's performance on the PLGA dataset.

To perform AL, the available 113 instances were treated as the starting set and proceeded with defining the pool set. Potential values of the corresponding parameters were set based on the

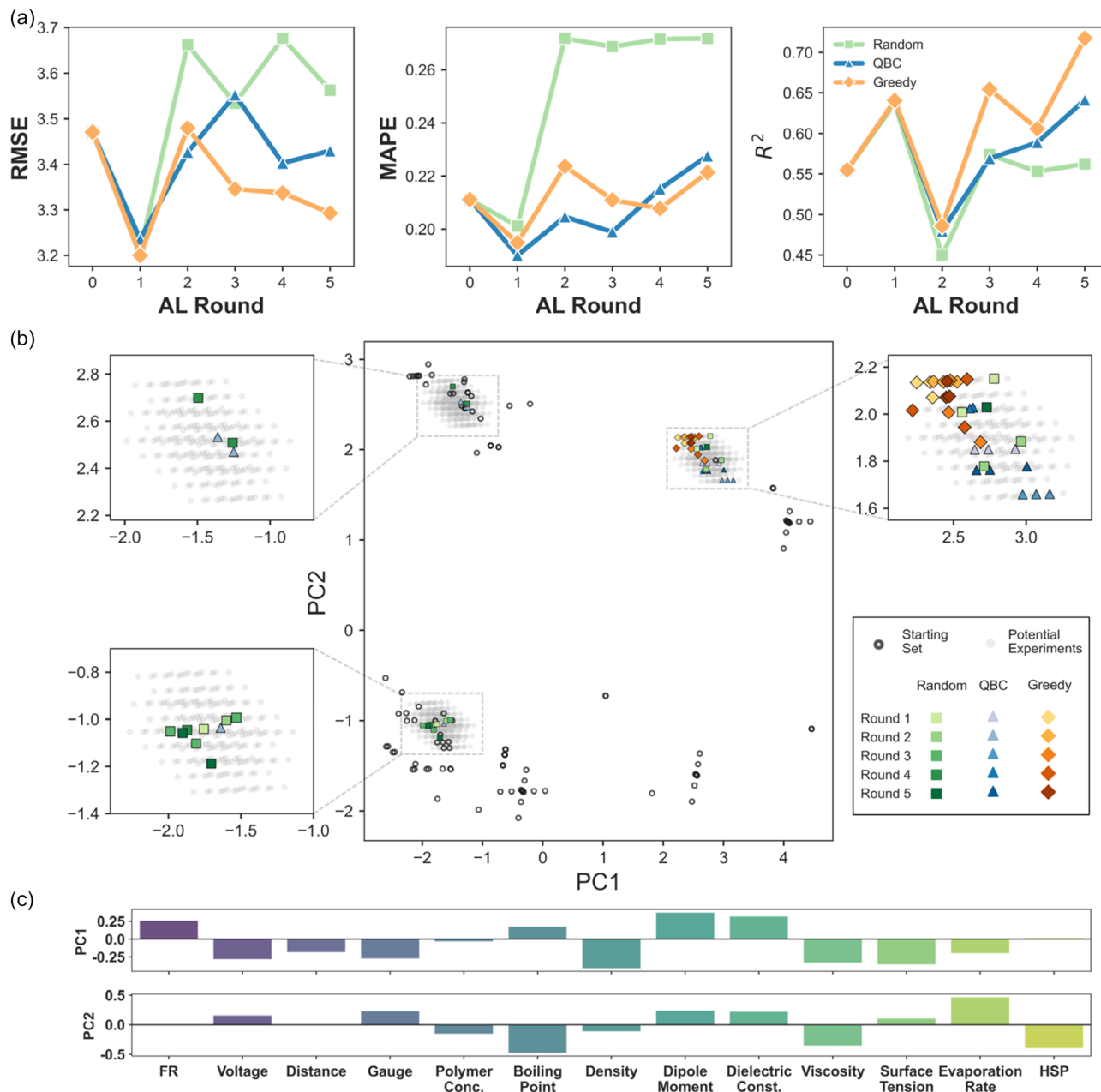
domain knowledge about the electrospay process, forming 720 possible combinations. It is important to note that conducting all 720 experiments would be an extremely challenging, if not impossible, task due to resource constraints. Furthermore, to the best of our knowledge, there was no conventional DoE method

**Table 1.** Details of experiments recommended with AL algorithms and the results obtained. The needle's outer diameter was kept constant at 0.71 mm. Each experiment was repeated in triplicates ( $N = 3$  independent experiments). Particle mean diameters and standard deviations were obtained by counting at least 100 particles from three SEM images taken at different locations.

AL Round	Method	Sample	Flow Rate [mL h <sup>-1</sup> ]	Voltage [kV]	Collection Distance [mm]	Solvent	Polymer [w/v%]	Particle Diameter Mean [μm]	Particle Diameter Standard Deviation [μm]
Round 1	QBC	Q-1-1	3.60	14.0	160	Acetone	5.0	3.49	0.87
		Q-1-2	3.60	14.0	220	Acetone	5.0	3.60	0.93
		Q-1-3	3.60	14.0	200	Acetone	5.0	3.30	0.95
	Random	R-1-1	0.30	12.0	200	Acetone	1.0	1.90	0.43
		R-1-2	2.50	16.0	160	Chloroform	5.0	9.07	1.17
		R-1-3	3.60	16.0	180	Acetone	1.0	2.99	0.64
	Greedy	G-1-1	0.30	16.0	220	Acetone	1.0	0.97	0.22
		G-1-2	0.30	14.0	220	Acetone	1.0	1.35	0.30
		G-1-3	0.30	16.0	200	Acetone	1.0	1.06	0.22
Round 2	QBC	Q-2-1	3.60	16.0	160	DCM	5.0	9.42	2.00
		Q-2-2	3.60	14.0	160	DCM	5.0	12.13	1.25
		Q-2-3	3.60	16.0	160	Chloroform	5.0	10.75	1.47
	Random	R-2-1	1.40	12.0	180	Acetone	5.0	3.23	0.85
		R-2-2	0.30	10.0	160	Acetone	1.0	1.32	0.33
		R-2-3	0.30	8.0	160	DCM	1.0	6.75	1.28
	Greedy	G-2-1	1.40	16.0	220	Acetone	1.0	1.86	0.40
		G-2-2	0.30	16.0	160	Acetone	1.0	1.24	0.34
		G-2-3	0.30	16.0	180	Acetone	1.0	1.12	0.32
Round 3	QBC	Q-3-1	3.60	8.0	200	Acetone	5.0	5.15	2.09
		Q-3-2	3.60	8.0	180	Acetone	5.0	4.88	1.80
		Q-3-3	3.60	8.0	220	Acetone	5.0	4.98	2.02
	Random	R-3-1	0.30	16.0	160	Chloroform	5.0	9.14	2.66
		R-3-2	3.60	14.0	220	Chloroform	5.0	15.32	1.98
		R-3-3	3.60	10.0	220	Chloroform	1.0	11.03	1.41
	Greedy	G-3-1	0.30	12.0	220	Acetone	1.0	1.06	0.22
		G-3-2	2.50	16.0	220	Acetone	1.0	2.03	0.49
		G-3-3	0.30	8.0	220	Acetone	1.0	1.16	0.30
Round 4	QBC	Q-4-1	2.50	16.0	160	Acetone	3.0	2.93	0.73
		Q-4-2	2.50	16.0	180	Acetone	3.0	2.82	0.79
		Q-4-3	1.40	16.0	160	Acetone	3.0	3.00	0.75
	Random	R-4-1	2.50	14.0	200	DCM	1.0	9.55	1.38
		R-4-2	2.50	8.0	220	DCM	1.0	10.93	1.80
		R-4-3	1.40	16.0	160	Chloroform	5.0	14.14	2.29
	Greedy	G-4-1	3.60	16.0	220	Acetone	1.0	2.17	0.56
		G-4-2	0.30	10.0	220	Acetone	1.0	1.09	0.31
		G-4-3	0.30	16.0	220	Acetone	3.0	2.37	0.68
Round 5	QBC	Q-5-1	0.30	8.0	220	Acetone	3.0	3.64	0.63
		Q-5-2	3.60	8.0	220	Acetone	3.0	3.23	0.57
		Q-5-3	0.30	8.0	200	Acetone	3.0	2.20	0.47
	Random	R-5-1	0.30	8.0	200	Chloroform	3.0	11.83	2.19
		R-5-2	1.40	12.0	220	Chloroform	3.0	11.76	1.68
		R-5-3	2.50	16.0	160	Acetone	3.0	2.75	0.53
	Greedy	G-5-1	0.30	14.0	200	Acetone	1.0	1.03	0.17
		G-5-2	1.40	14.0	220	Acetone	1.0	2.02	0.44
		G-5-3	1.40	16.0	200	Acetone	1.0	1.99	0.40

that supports the planning for factors and levels outlined. In total, five rounds of AL were performed with laboratory-based experimentation to compare the performance of three strategies. Wet-lab electrospray experiments were conducted following the recommended parameters from AL. Particles obtained from experiments were further characterized by SEM to understand the particle size (Figure S3, Supporting Information). The results were summarized in Table 1.

We first explain the rationale for choosing hyperparameters in prospective AL. The batch size of three was selected based on the consideration of practical capacity limits in the lab, knowing that the batch size showed limited influence on AL performance (Figure 5a). A batch size of one candidate per round could be performed but would result in the waste of laboratory reagents and preparation time. This will be further elaborated in the discussion section. Furthermore, since AL was prospectively



**Figure 6.** a) ML model performance, as evaluated by 6-fold cross-validation with RMSE, MAPE, and  $R^2$ , with prospective AL on PCL dataset. b) Visualization of the starting PCL dataset, potential experiments space, and AL selected instances with principal component analysis (PCA). Zoom-in plots are three areas where potential experiments were located in. Each point represents an instance in the dataset and is color-coded (black circles: starting set, grey dots: potential experiments, and colored markers: instances selected by AL algorithm). AL strategies adopted are represented with different hues and markers (green squares: Random, blue triangles: QBC, and orange diamonds: Greedy). Selected instances with AL are shaded to indicate in which round they were selected (from light to dark: 1<sup>st</sup> to 5<sup>th</sup> Round). c) contributions of features to the two principal components.



implemented on the PCL dataset, the pool dataset defined here is  $\approx 6$  times larger than the initial dataset. Thus, it was roughly equivalent to 10% partition in a retrospective scenario. We expected, under this setting, the difference of AL strategies would be more prominent. The hyperparameters in QBC algorithm remained the same as in the benchmarking setting, since the size and type of committee showed no significant impact on AL performance. For the evaluation model, XGBoost with default hyperparameters were used due to its superior performance compared with other algorithms, as reported in our previous publication.<sup>[41]</sup> In addition, the comparative nature of the prospective AL experiment did not pose a strict requirement on hyperparameter optimization.

The performance after each round was recorded and plotted in **Figure 6a**. The Greedy method managed to reduce the RMSE from 3.47 to 3.29, whereas the random baseline increased it to 3.56. The QBC method experienced some fluctuations and slightly reduced the RMSE to 3.43. In terms of MAPE, neither of these three methods contributed to reducing MAPE. Both QBC and Greedy methods maintained it around 23% and 22%, respectively, while the random baseline leveled up the MAPE from the starting 21% to 27%. Regarding the  $R^2$ , starting from 0.55, the Greedy method achieved the highest  $R^2$  of 0.71, surpassing the QBC (0.64) and random (0.56) methods. It was unsurprising that the Greedy method performed better than QBC, as it was consistent with the PLGA dataset and some published datasets.

To further shed light on the details of AL, we visualized it through principal component analysis (PCA) in **Figure 6b**. The black circles were datapoints from the starting set, some of which followed specific patterns, like on crosses or straight lines (**Figure S4**, Supporting Information). Upon examining the raw data, it became evident that these patterns originated from a well-planned investigation of PCL micro/nanoparticle production with electrospray.<sup>[42,43]</sup> Another finding of this plot was about the relationship between the location of datapoints and the type of solvent used. As mentioned previously, the type of solvent was represented by the physical properties. These properties had different contributions to principal components as indicated in **Figure 6c**. Four distinct clusters were highlighted in **Figure S4** (Supporting Information), corresponding to the solvents used. Specifically, DCM cluster was on the top left, and chloroform cluster was on bottom left. It could be seen that these two clusters on the left already had a few datapoints situated in-between, meaning that prior knowledge (instances) existed before AL. Sitting on the top right corner, the acetone cluster owned only a few neighboring datapoints. Furthermore, like the patterns created by the existing data, the potential experiments (represented by grey dots) were organized into unique lattice-like patterns.

Having interpreted the “background” datapoints of the PCA plot, the focus was then turned to the colored markers which represented the in-house experimental data. Green squares were points selected randomly and were therefore spreading throughout three clusters. QBC method queried a few points within the DCM and chloroform clusters, while most queried points stayed in the acetone cluster. Interestingly, the Greedy method heavily sampled the data scarce acetone cluster. Such trends were associated with the underlining query mechanism. As outlined in

algorithm 2, the Greedy algorithm would evaluate the distance between a potential instance and all existing data and select deliberately the ones that had the longest distance. The preference of the acetone cluster was justified from the PCA plot, which showed that this cluster was the furthest from the existing knowledge. It is worth noting that both AL strategies, the Greedy approach and QBC method, were superior to the random baseline.

### 3. Discussions

Our study demonstrated the potential application of AL as an experiment planning tool, with a focus on materials datasets. The key parameters in AL were examined to shed light on their impact. In addition, we have demonstrated the usage of AL on our in-house modeling attempt for electrospraying.

Before any discussion, we believed that it would be beneficial to clarify the relationship between AL and SBO/BO. This was because optimization algorithms was highlighted in many publications while sharing the name of AL.<sup>[13,18,32,44]</sup> Indeed, both methods are used for recommending experiments. However, BO has the goal of searching for desired target, which in materials discovery context is an optimized materials property. After multiple rounds of BO and experiments, researchers will be able to find the experiment conditions that yield, or at least close to, the designed target.<sup>[30]</sup> If SBO/BO were to be used in our electrospray dataset, the task would have been shifted to identifying the electrospray condition that generates the smallest, or the largest, PCL particles. On the contrary, rounds of AL will explore an enriched dataset which leads to the best ML modeling. Moreover, the algorithm of SBO/BO and AL can be different. Typically, the strategy used in SBO/BO balances exploration (high uncertainty) and exploitation (high expectation), whereas AL solely focuses on exploration. In this regard, SBO/BO was more often considered as an optimization tool and AL was considered closer to DoE. Notably, performing SBO/BO will still accumulate new data points and benefit the training of surrogates. For example, Borkowski et al. utilized SBO to identify and understand the interactions between buffer composition and the protein production in cell-free systems.<sup>[45]</sup> They implemented a modified QBC method that accounted for both exploration and exploitation. After 10 rounds, the model’s coefficient of determination ( $R^2$ ) increased significantly. Also, Montoya et al. leveraged SBO to identify stable Fe-X binary compounds and reported the reduction of prediction error.<sup>[46]</sup> However, the improvement of prediction performance is rather a by-product of the global searching process. As discussed by Lookman et al. empirical observations have shown that high model accuracy is not necessary in SBO/BO processes.<sup>[32]</sup> Thus, we would like to highlight again that AL has a different purpose, when compared with optimization strategies.

Focusing on our results, the effectiveness of AL was confirmed through benchmarking on datasets and the results were in well agreement with literature. For example, we validated the conclusion that even a small committee in QBC algorithm yielded similar performance as the one with a large committee. Nevertheless, some previously under-examined topics showed surprising results. Notably, the batch size in AL only had trivial

effect on AL performance in our experiment (Figure 3). When scrutinizing this seemingly simple parameter, the impact turned out to be highly sophisticated. From a theoretical standpoint, most AL algorithms were designed for sequential AL (SAL), where only one instance was selected at a time (i.e., batch size = 1). As summarized by Settle, implementing the strategy of myopically choosing the top  $k$ -best instances from the pool set would fail to consider the overlapped information between instances.<sup>[47]</sup> Thus, having a large batch size was believed to be detrimental to AL performance. However, from a practical point of view, conducting just one experiment to label one instance could lead to the waste of consumables, testing materials, and characterization devices in the laboratory. Although it was widely assumed that the performance of AL algorithms would decline with larger batch sizes, limited evidence was provided with regression datasets. In this study, we conducted an evaluation of the impact of varying the batch size, and our results did not entirely align with previous assumptions. In one dataset, the QBC method with a larger batch size of 4 or 8 outperformed versions with a batch size of 1 or 2 (Figure 3). One possible explanation for our observation was that intrinsic noises in real-world data and the high dimensionality of data require multiple datapoints for modeling. In other words, the information in a noisy and scarce data source may not be as 'overlapped' in the SAL settings. Our observation agreed with another previous work where a marginal reduction in performance was observed when increasing the batch size.<sup>[48]</sup>

In addition, it is essential to emphasize some potential applications of AL in experimental studies. One was inspired by the benchmarking results of ESOL, FreeSolv, and lipophilicity datasets, where Morgan fingerprints were used as the input. Morgan fingerprints was one of the most popular chemical descriptors used in computational modelling.<sup>[49]</sup> Essentially, chemical descriptors use a set of numbers to represent molecular structures. When it comes to the Morgan fingerprint, it uses 2048 features to describe a molecule. Many other featurization methods, like the Mol2Vec or the Mordred fingerprint, were also associated with this high dimensionality, making it difficult to design experiments systematically in this feature space.<sup>[50,51]</sup> The promising performance of AL in these datasets suggested its potential in material and drug discovery applications, where preliminary laboratory results could be highly costly and scarce. Furthermore, the state-of-the-art molecular featurization and prediction using graph neural networks also require a large amount of training data.<sup>[52,53]</sup> Although our wet experiment was carried out on a material development task, it is highlighted that, based on the benchmarking evaluation, AL could also benefit material and drug discovery research. For example, assisting in the construction of more reliable and accurate QSPR models. Moreover, AL strategies could also be applied in transfer learning or model fine-tuning scenarios for experimental studies. Taking our application as an example, if a researcher would like to investigate a new or greener solvent that had never been documented in the dataset, they could implement AL to recommend a few datapoints and perform the experiments. With these supplementary data, errors caused by extrapolation may decrease, leading to a more robust ML model. Li et al. recently presented a study on how AL and yoked learning could benefit the training of deep learning models with retrospective applications on materials datasets.<sup>[34]</sup> Another potential application was to work

synergistically with other technologies to tackle the data scarcity issue. This includes miniaturization technologies like microfluidics to cut down the usage of reagents and further reduce the expense to an acceptable range. The use of microfluidics and high-throughput experiments were demonstrated as routes to efficiently collect data in biology and materials research.<sup>[54–58]</sup> Lab robots would also greatly alleviate the time constraints in lab experiments.<sup>[59,60]</sup>

## 4. Conclusion

Insufficient data became a significant issue that troubled many endeavors to leverage ML to model and analyze data from experimental research. The current paradigm of data accumulation through performing full factorial wet-lab experiments was extremely resource-demanding and time-consuming. In the present study, we proposed and evaluated AL as a potential new paradigm of experiment planning. The QBC and Greedy methods demonstrated superior capacity in improving ML model performance with significantly smaller amounts of data. Explorations on parameters in AL suggested trivial impact of batch size on the performance of AL methods on materials datasets. Further examination confirmed the limited influence of QBC committee size and type on its performance. Finally, wet-lab experiments of electrospray preparation of PCL microparticles showcased AL in assisting experiment planning. It was thus concluded that AL could serve as an effective and versatile tool to strategically address the current bottleneck of data scarcity in experimental studies, enabling researchers to fully harness the power of ML.

## 5. Experimental Section

*Data Extraction and Processing for in-House Dataset:* The in-house dataset was extracted from previous publications with detailed procedures described in our previous publication.<sup>[41]</sup> In brief, seven key parameters in electrospraying were manually collected into a spreadsheet. These parameters included polymer type, polymer concentration, solvent type, flow rate, applied voltage, needle outer diameter, and collection distance (Table S2, Supporting Information). Subsequently, the type of solvent in the original dataset was substituted by their respective physical properties. Notably, eight solvent physical properties were carefully selected to best characterize solvent behavior during electrospray. These properties included boiling point, density, dipole moment, dielectric constant, viscosity, surface tension, relative evaporation rate (where butyl acetate = 1), and the Hansen solubility distance calculated with respect to the polymer. Then, the data was further divided based on the type of polymer, resulting in two different datasets: the PLGA dataset and the PCL dataset. In these datasets, the remaining 13 features were subjected to preprocessing. They were first standardized by subtracting the mean and dividing by the standard deviation. To address the missing values, a  $k$ -nearest neighbor (kNN) imputation was performed. Finally, the data underwent a shuffling process to ensure random arrangements. All data processing was performed using sklearn ver. 1.1.3 in a Conda environment (Python version 3.9.17).

*AL:* Essentially, AL algorithms evaluate the existing data and give recommendations by selecting from a potential experiment pool (the pool set). In the present study, three AL strategies were chosen and evaluated. The simplest strategy was the random query method where recommendations were randomly picked from the pool set (**Algorithm 1**). The second strategy was the Greedy method. To perform the Greedy method, Euclidean squared distances from a specific sample in the pool set to

**Algorithm 1.** Random query method for AL in regression.

**Input:** the potential experiment pool  $\mathcal{P} = \{(\mathbf{x}_i)\}_{i=1}^n$ , the batch size  $k$  per round of query.

- 1: Randomly sample  $k$  times without replacement from the pool set  $\mathcal{P}$
- 2: Construct recommendation set  $\mathcal{R}_R = \{(\mathbf{x}_i^*)\}_{i=1}^k$

**Output:** the recommendation set  $\mathcal{R}_R$ .

all existing training data were first calculated. The total distance of that specific experiment to the training set was obtained by summing up the Euclidean squared distances. This step was then repeated on all samples in the pool set to understand which samples were the furthest from the training data. Finally, depending on the batch size, the top  $k$  potential experiments in the pool set with the largest total Euclidean squared distances were recommended (**Algorithm 2**). The third method implemented in the current paper was the QBC method. The concept of the QBC method was to select the most “uncertain” samples in the pool set judged by the committee, which was constructed from models trained with existing data. The uncertainty of a specific sample in the QBC method was characterized by disagreement, or more specifically the standard deviation of the predictions given by the committee members in regression.<sup>[26,27]</sup> Thus, procedures to conduct QBC strategy were: 1) bootstrapping (resampling with replacement)  $c$  times from the training data to construct  $c$  new groups of training data, 2) train a group of regression models respectively with the bootstrapped data, 3) for each sample in the pool set, make predictions with models in the committee and calculate the standard deviation of  $c$  predictions as the measurement of uncertainty, and 4) repeat the previous step to all samples in the pool set and recommend the top  $k$  samples with the largest uncertainty (**Algorithm 3**). The performance of models was evaluated RMSE, which was calculated by:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N}} \quad (1)$$

where  $y_i$  is the ground truth value,  $\hat{y}_i$  is the predicted value, and  $N$  is the total number of samples.

We further introduced two other metrics to evaluate model performance in prospective AL, namely MAPE and  $R^2$ . They were calculated by:

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{|y_i|} \quad (2)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (\bar{y} - y_i)^2} \quad (3)$$

**Algorithm 2.** Greedy method for AL in regression.

**Input:** the potential experiment pool  $\mathcal{P} = \{\mathbf{x}_i\}_{i=1}^n$ , the training dataset with known labels  $\mathcal{L} = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^m$ , the batch size  $k$  per round of query.

- 1: **for** each  $\mathbf{x}$  the pool set  $\mathcal{P}$  **do**
- 2: Calculate Euclidean squared distances  $\{d_j^2 = (\mathbf{x} - \mathbf{x}_j)^2\}_{j=1}^m$  between  $\mathbf{x}$  and each sample  $\mathbf{x}_j$  in the training dataset (labels not included)  $\mathcal{L}_x = \{\mathbf{x}_j\}_{j=1}^m$
- 3: Calculate the total Euclidean square distance  $d$  by summing up the distances with respect to all samples in the training dataset  $d^2 = \sum_{j=1}^m d_j^2$
- 4: **end for**
- 5: Construct recommendation set  $\mathcal{R}_G = \{\mathbf{x}_i^*\}_{i=1}^k$  by selecting the top  $k$  samples with the largest total Euclidean square distance  $d^2$

**Output:** the recommendation set  $\mathcal{R}_G$ .

**Algorithm 3.** QBC method for AL in regression.

**Input:** the potential experiment pool  $\mathcal{P} = \{\mathbf{x}_i\}_{i=1}^n$ , the training dataset with known labels  $\mathcal{L} = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^m$ , the batch size  $k$  per round of query, the number of committee members  $c$ , the learning algorithm  $f$  used for the committee.

- 1: Bootstrap  $c$  times from training set  $\mathcal{L} = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^m$  to construct  $\mathcal{B} = \{\mathcal{L}^1, \mathcal{L}^2, \dots, \mathcal{L}^c\}$
- 2: Train a committee of regressors  $\mathcal{C} = \{f^1, f^2, \dots, f^c\}$  with resampled sets in  $\mathcal{B}$  respectively
- 3: **for** each  $\mathbf{x}$  in the pool set  $\mathcal{P}$  **do**
- 4: Calculate the prediction result  $\hat{y}^i$  of the given  $\mathbf{x}$  with each committee member  $f^i$  to obtain the committee results  $\{\hat{y}^1, \hat{y}^2, \dots, \hat{y}^c\}$  where  $\hat{y}^i = f^i(\mathbf{x})$
- 5: Calculate the standard deviation  $\sigma_x$  of the committee results as the measurement of disagreement
- 6: **end for**
- 7: Construct recommendation set  $\mathcal{R}_Q = \{\mathbf{x}_i^*\}_{i=1}^k$  by selecting the top  $k$  samples with the largest disagreement, as characterized by  $\sigma_x$

**Output:** the recommendation set  $\mathcal{R}_Q$ .

where  $y_i$  is the ground truth value,  $\hat{y}_i$  is the predicted value,  $\bar{y}$  is the mean of all ground truth values, and  $N$  is the total number of samples.

**AL: Retrospective:** In this section we describe the process to conduct AL on readily available data for the purpose of validating the performance of AL algorithms and exploring the effect of various settings in AL. Furthermore, the algorithm was implemented on the PLGA dataset to verify the effectiveness prior to utilizing them for prospective AL.

To perform retrospective AL, a full dataset was first split into the starting set and the pool set at a predefined ratio. Here, the starting set was treated as prior knowledge for the user, whereas the ground truth labels in the pool set were hidden to simulate ‘potential experiments’. Before starting, an initial model performance was obtained by 6-fold cross validated training and evaluation on the starting set. XGBoost with default hyperparameters (py-XGBoost ver. 1.5.0) was selected as the ML model in our case.<sup>[53,61,62]</sup> Then, retrospective AL was conducted by repeating the following three steps. Firstly, three AL algorithms were implemented on the pool set to select ‘potential experiments’ respectively. The number of experiments was controlled by the batch size of AL. Furthermore, ground truth values of these suggested experiments were revealed and put back to the starting set as if the wet lab experiments were performed to obtain results. Finally, model performance was evaluated with the same 6-fold cross validation process for three updated starting sets (with respect to the three AL algorithms). These three steps together were referred to as one round of AL process. Three AL algorithms were first implemented retrospectively on several published datasets (**Table 2**) to evaluate performance on engineering and materials tasks. Specifically, the concrete compressive strength dataset was obtained from UCI.<sup>[63]</sup> The ESOL, FreeSolv, and Lipophilicity datasets were retrieved from the corresponding implementations in DeepChem library (ver. 2.7.1) with the default molecular featurizer (extended-connectivity fingerprint (ECFP4, 2048 bits)).<sup>[35,64–67]</sup> The double perovskite bandgap dataset was sourced from Matminer library (ver. 0.9.0) with the default Magpie elemental descriptor and oxidation states as the featurization.<sup>[68,69]</sup> The ratio to split starting sets and pool sets was set at 2:98 for all datasets and data random shuffling was performed prior to partitioning. In the benchmarking study, the number of committee members was set at 4. Gradient boost decision tree (GBDT, with default hyperparameters in the implementation by sklearn ver. 1.1.3) was selected as algorithms for models that constructed the QBC committee. The batch size for one round of AL query was initially set at 4. And 20 rounds of AL were performed on each dataset to observe the efficiency to improve model performance. The percentage reduction of RMSE was calculated by subtracting the RMSE at the 20<sup>th</sup> round from the

**Table 2.** Datasets used to evaluate AL algorithms.

Dataset Type	Dataset Name	Feature Type	Prediction Target	Number of Features	Number of Instances	Data Source
Processing	Concrete Compressive Strength Dataset	Processing parameters	Compressive strength	8	1030	UCI <sup>[63]</sup>
Physical Chemistry	ESOL Dataset	Molecular structure	Aqueous solubility	1024	902	Delaney, DeepChem <sup>[64]</sup>
Physical Chemistry	FreeSolv Dataset	Molecular structure	Hydration free energy	1024	513	Mobley and Guthrie, DeepChem <sup>[65]</sup>
Physical Chemistry	Lipophilicity Dataset	Molecular structure	Lipophilicity	1024	3360	Hersey, DeepChem <sup>[66]</sup>
Materials Design	Double Perovskite Bandgap Dataset	Composition	Bandgap of double perovskites	136	1306	Pilania et al. Matminer <sup>[68]</sup>
Processing	PLGA Electrospray	Processing parameters	Particle diameter	13	235	Wang et al. in-house curated <sup>[41]</sup>

starting RMSE, and then divided by the starting RMSE. To better characterize the results, the mean RMSE and its 95% confidence intervals were obtained based on 100 times repetition of AL processes. In each repetition, the starting set and pool set were randomly partitioned.

To investigate the influence of hyperparameters in the QBC algorithm, additional experiments were performed on published datasets. This included observing QBC algorithm performance under different committee sizes and committee compositions. More specifically, the effect of committee size was examined through benchmarking with committees constructed from 2, 4, and 8 GBDT models. The effect of committee composition was evaluated by selecting other ML algorithms other than GBDT. Committee A was composed of the default 4 GBDT models. Committee B had 4 random forest (RF) models. Committee C had a wide selection of one GBDT model, one RF model, one XGBoost model, and one multi-layer perceptron (MLP) model. For committee D, 4 MLP models were used. All models used the default hyperparameter settings implemented in sklearn ver. 1.1.3. The maximum iteration for MLP was set at 10 000.

Two parameters in retrospective AL, namely the batch size and the starting ratio, was studied. A batch size of 4 instances per round was chosen empirically in the previous benchmarking experiment. With 20 rounds of AL performed, it added up to a total number of 80 instances. Considering that running 20 rounds of AL with different batch sizes would not provide comparable results, the budget of 80 instances was fixed throughout this experiment. For example, in the batch size setting with 1 instance per query, 80 rounds of AL were performed, whereas 10 rounds of AL queries were conducted when the size was chosen as 8. For the starting ratio, the starting set was partitioned from the full dataset with a 2:98 ratio to the pool set. This 2% ratio was further altered to 10%, 20% and 40% in the AL process (20 rounds with a batch size of 4). All experiments were repeated 100 times, and the dataset was randomly shuffled each time.

Finally, AL strategies were benchmarked retrospectively on our manually curated PLGA dataset. The batch size and starting ratio experiments were performed on this dataset. The batch sizes were similarly changed from 1, 2, 4 to 8. Due to the small dataset size, 10%, 20%, 40% and 60% were tested for the starting ratio.

**AL: Prospective:** Prospective AL was implemented on the data scarce PCL dataset. The PCL dataset was constructed by 113 instances of experiment records describing the electrospray results of PCL polymer. The same feature engineering process as the PLGA dataset was performed on this dataset. All instances in the PCL dataset were treated as the 'starting set' and the potential experimental parameters were defined based on prior experiment experience (Table 3). In total, the number of potential experiments (the pool set) was 720. In a prospective setting for AL, three different algorithms, random query, Greedy, and QBC, were implemented in parallel. In total, five rounds of AL were performed, and the batch size of 3 experiments was chosen per round. Experiments recommended by AL algorithms were performed in the laboratory in triplicates. Results after characterization were put back to the starting dataset and evaluated with an XGBoost model (similarly, with default hyperparameters).

**Table 3.** Potential experiment parameters to be selected by AL algorithms.

Parameters	Unit	Potential Values
Type of Polymer	–	PCL
Polymer Concentration	%(w/v)	1.0, 3.0, 5.0
Type of Solvent	–	Dichloromethane (DCM), Acetone, Chloroform
Flow Rate	mL h <sup>-1</sup>	0.3, 1.4, 2.5, 3.6
Applied Voltage	kV	8.0, 10.0, 12.0, 14.0, 16.0
Needle Outer Diameter	mm	0.71
Collection Distance	mm	160, 180, 200, 220

The evaluation was implemented through a 6-fold cross validation with RMSE, MAPE, and  $R^2$  used as metrics for model performance. To further illustrate the process of AL, all datapoints in PCL, including starting data, potential experiment space, and additional experiments, were visualized through PCA (sklearn ver. 1.1.3).

**Wet-Lab Experiments: Materials:** The polymer PCL with an average molecular weight of 80 000 g mol<sup>-1</sup> was purchased from Sigma-Aldrich (Gillingham, UK). For solvents, chloroform (99% purity) and DCM (99.8% purity) were purchased from Sigma-Aldrich (Gillingham, UK). Acetone (99% purity) was obtained from LP Chemicals Limited (Cheshire, UK).

**Wet-Lab Experiments: Electrospray of PCL Particles:** Polymer solutions were prepared by mixing the PCL pellets in the corresponding solvent at ambient temperature with magnetic stirring overnight. For PCL with acetone as the solvent, the solutions were kept to 50 °C overnight to allow complete dissolution. The solutions were then fed through a syringe pump (Harvard PHD 4400, Edenbridge, UK) which was connected to a 22-gauge needle (outer diameter 0.71 mm) through a capillary. The positive output of a high voltage power supply (Glassman High Voltage Inc., NJ, United States) was connected to the needle through a crocodile clamp and the collection plate was connected to the ground. Before electrospray, the distance between the needle and collection plate, flow rate, and voltage were adjusted to the suggested value according to AL algorithms. Experiments were conducted at atmospheric pressure. The temperature and humidity in the room were controlled to be 23–25 °C and 40–50%. Particles were collected with a glass slide placed on the collection plate for further characterization. Each experimental condition recommended by AL was repeated three times on different days.

**Wet-Lab Experiments: Characterization of PCL Particles:** Scanning electron microscopy (SEM) was used as the main characterization method of particle size. The glass slides were observed with a Zeiss Gemini 360 SEM (Germany) under an acceleration voltage of 1.00 kV through an In-Lense detector. Three images were taken randomly on different locations of the glass slide for each sample. Images were further analyzed using

Image) (National Institute of Health, USA). To obtain the particle size distribution, diameters of particles were measured randomly on the images and this value for 100 particles was recorded. The mean size of a specific sample was calculated by taking the mean of these 100 measurements.

**Statistics:** All statistics were performed with Mann-Whitney-Wilcoxon two-tailed U tests. The tests were implemented through Scipy (version 1.10.1) in a Conda environment (Python version 3.9.17).

## Supporting Information

Supporting Information is available from the Wiley Online Library or from the author.

## Acknowledgements

The author Fanjin Wang would like to thank the Engineering and Physical Sciences Research Council (EPSRC) for supporting his PhD research (grant nos. EP/R513143/1 and EP/W524335/1).

## Conflict of Interest

The authors declare no conflict of interest.

## Data Availability Statement

The datasets used in benchmarking are available online at their corresponding sources. The benchmarking results can be reproduced with the framework provided in GitHub code repository of this project: <https://github.com/FrankWanger/ExpAL>. Any other data that support the findings of this study are available from the corresponding author upon reasonable request.

## Keywords

active learning, machine learning, materials development, materials discovery, small data

Received: November 22, 2023

Revised: April 2, 2024

Published online:

- [1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, in *Int. Conf. on Learning Representations*, online, May 2021.
- [2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei et al., in *Advances on Neural Information Processing Systems*, Vancouver, Canada, December 2020, pp. 1877–1901.
- [3] A. Suwardi, F. Wang, K. Xue, M.-Y. Han, P. Teo, P. Wang, S. Wang, Y. Liu, E. Ye, Z. Li, X. J. Loh, *Adv. Mater.* **2022**, *34*, 2102703.
- [4] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, A. Walsh, *Nature* **2018**, *559*, 547.
- [5] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, S. Zhao, *Nat. Rev. Drug Discovery* **2019**, *18*, 463.
- [6] Z. Liu, D. Zhu, L. Raju, W. Cai, *Adv. Sci.* **2021**, *8*, 2002923.
- [7] C. Lv, X. Zhou, L. Zhong, C. Yan, M. Srinivasan, Z. W. Seh, C. Liu, H. Pan, S. Li, Y. Wen, Q. Yan, *Adv. Mater.* **2022**, *34*, 2101474.
- [8] V. L. Deringer, M. A. Caro, G. Csányi, *Adv. Mater.* **2019**, *31*, 1902765.
- [9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, in *IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, May 2009, pp. 248–255.
- [10] OpenAI (Preprint), arXiv:2303.08774, v1, submitted: Mar. 2023.
- [11] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, K. A. Persson, *APL Mater.* **2013**, *1*, 011002.
- [12] F. C. Bernstein, T. F. Koetzle, G. J. B. Williams, E. F. Meyer, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, M. Tasumi, *J. Mol. Biol.* **1977**, *112*, 535.
- [13] R. Chang, Y.-X. Wang, E. Ertekin, *npj Comput. Mater.* **2022**, *8*, 1.
- [14] J. Antony, *Design of Experiments for Engineers and Scientists*, Elsevier, London 2014.
- [15] J. Gilman, L. Walls, L. Bandiera, F. Menolascina, *ACS Synth. Biol.* **2021**, *10*, 1.
- [16] R. Arboretti, R. Ceccato, L. Pegoraro, L. Salmaso, *Qual. Reliab. Eng. Int.* **2022**, *38*, 1131.
- [17] S. N. Politis, P. Colombo, G. Colombo, D. M. Rekkas, *Drug Delivery Ind. Pharm.* **2017**, *43*, 889.
- [18] B. Cao, L. A. Adutwum, A. O. Oliynyk, E. J. Lubner, B. C. Olsen, A. Mar, J. M. Buriak, *ACS Nano* **2018**, *12*, 7434.
- [19] G. E. P. Box, D. W. Behnken, *Technometrics* **1960**, *2*, 455.
- [20] A. S. Hedayat, N. J. A. Sloane, J. Stufken, *Orthogonal Arrays*, Springer, New York, NY 1999.
- [21] B. Settles, in *Active Learning and Experimental Design Workshop in Conjunction with AISTATS 2010, JMLR Workshop and Conf. Proc.*, Sardinia, Italy, May 2010, pp. 1–18.
- [22] L. Yang, Y. Zhang, J. Chen, S. Zhang, D. Z. Chen, in *Medical Image Computing and Computer Assisted Intervention - MICCAI 2017* (Eds: M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, S. Duchesne), Springer International Publishing, Cham 2017, pp. 399–407.
- [23] S. Budd, E. C. Robinson, B. Kainz, *Med. Image Anal.* **2021**, *71*, 102062.
- [24] Y. Freund, H. S. Seung, E. Shamir, N. Tishby, *Mach. Learn.* **1997**, *28*, 133.
- [25] N. Abe, H. Mamitsuka, in *Proc. of the Fifteenth Int. Conf. on Machine Learning*, Morgan Kaufmann Publishers Inc, San Francisco, CA 1998, pp. 1–9.
- [26] H. S. Seung, M. Opper, H. Sompolinsky, in *Proc. of the Fifth Annual Workshop on Computational Learning Theory*, Association For Computing Machinery, New York, NY 1992, pp. 287–294.
- [27] R. Burbidge, J. J. Rowland, R. D. King, in *Intelligent Data Engineering and Automated Learning - IDEAL 2007* (Eds: H. Yin, P. Tino, E. Corchado, W. Byrne, X. Yao), Springer Berlin Heidelberg, Berlin, Heidelberg 2007, pp. 209–218.
- [28] W. Cai, Y. Zhang, J. Zhou, in *2013 IEEE 13th Int. Conf. on Data Mining*, IEEE, Dallas, TX 2013, pp. 51–60.
- [29] F. Di Fiore, M. Nardelli, L. Mainini (Preprint), arXiv:2303.01560, v3, submitted: Mar. 2023.
- [30] S. Greenhill, S. Rana, S. Gupta, P. Vellanki, S. Venkatesh, *IEEE Access* **2020**, *8*, 13937.
- [31] A. Pandi, C. Diehl, A. Yazdizadeh Kharrazi, S. A. Scholz, E. Bobkova, L. Faure, M. Nattermann, D. Adam, N. Chapin, Y. Foroughijabbari, C. Moritz, N. Paczia, N. S. Cortina, J.-L. Faulon, T. J. Erb, *Nat. Commun.* **2022**, *13*, 3876.
- [32] T. Lookman, P. V. Balachandran, D. Xue, R. Yuan, *npj Comput. Mater.* **2019**, *5*, 1.
- [33] R. Rodríguez-Pérez, F. Miljković, J. Bajorath, *J. Cheminf.* **2020**, *12*, 36.
- [34] Z. Li, Y. Xiang, Y. Wen, D. Reker, *Artif. Intell. Life Sci.* **2024**, *5*, 100089.

- [35] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, V. Pande, *Chem. Sci.* **2018**, 9, 513.
- [36] A. Faulds, *Math. Comput. Sci.* **2022**, 4, 1.
- [37] B. Settles, M. Craven, in *Proc. of the 2008 Conf. on Empirical Methods in Natural Language Processing*, Association For Computational Linguistics, Honolulu, Hawaii **2008**, pp. 1070–1079.
- [38] J. Xie, J. Jiang, P. Davoodi, M. P. Srinivasan, C.-H. Wang, *Chem. Eng. Sci.* **2015**, 125, 32.
- [39] M. E. Cam, Y. Zhang, M. Edirisinghe, *Expert Opin. Drug Delivery* **2019**, 16, 895.
- [40] J. B. Fenn, M. Mann, C. K. Meng, S. F. Wong, C. M. Whitehouse, *Science* **1989**, 246, 64.
- [41] F. Wang, M. Elbadawi, S. L. Tsilova, S. Gaisford, A. W. Basit, M. Parhizkar, *Mater. Des.* **2022**, 219, 110735.
- [42] S. Zhang, C. Campagne, F. Salaün, *Coatings* **2019**, 9, 84.
- [43] S. Zhang, C. Campagne, F. Salaün, *Appl. Sci.* **2019**, 9, 402.
- [44] J. Schmidt, M. R. G. Marques, S. Botti, M. A. L. Marques, *npj Comput. Mater.* **2019**, 5, 1.
- [45] O. Borkowski, M. Koch, A. Zettor, A. Pandi, A. C. Batista, P. Soudier, J.-L. Faulon, *Nat. Commun.* **2020**, 11, 1872.
- [46] J. H. Montoya, K. T. Winther, R. A. Flores, T. Bligaard, J. S. Hummelshøj, M. Aykol, *Chem. Sci.* **2020**, 11, 8517.
- [47] B. Settles, *Active Learning Literature Survey*, University of Wisconsin–Madison, Madison, WI **2009**.
- [48] W. Cai, M. Zhang, Y. Zhang, *IEEE Trans. Neural Networks Learn. Syst.* **2017**, 28, 1668.
- [49] H. L. Morgan, *J. Chem. Doc.* **1965**, 5, 107.
- [50] S. Jaeger, S. Fulle, S. Turk, *J. Chem. Inf. Model.* **2018**, 58, 27.
- [51] H. Moriwaki, Y.-S. Tian, N. Kawashita, T. Takagi, *J. Cheminf.* **2018**, 10, 4.
- [52] P. Reiser, M. Neubert, A. Eberhard, L. Torresi, C. Zhou, C. Shao, H. Metni, C. van Hoesel, H. Schopmans, T. Sommer, P. Friederich, *Commun. Mater.* **2022**, 3, 1.
- [53] P. Xu, X. Ji, M. Li, W. Lu, *npj Comput. Mater.* **2023**, 9, 1.
- [54] M. Shevlin, *ACS Med. Chem. Lett.* **2017**, 8, 601.
- [55] A. O. Oliynyk, E. Antono, T. D. Sparks, L. Ghadbeigi, M. W. Gaultois, B. Meredig, A. Mar, *Chem. Mater.* **2016**, 28, 7324.
- [56] N. S. Eyke, B. A. Koscher, K. F. Jensen, *Trends Chem.* **2021**, 3, 120.
- [57] F. Ren, L. Ward, T. Williams, K. J. Laws, C. Wolverton, J. Hatrick-Simpers, A. Mehta, *Sci. Adv.* **2018**, 4, eaaq1566.
- [58] W. Li, T. Yang, C. Liu, Y. Huang, C. Chen, H. Pan, G. Xie, H. Tai, Y. Jiang, Y. Wu, Z. Kang, L.-Q. Chen, Y. Su, Z. Hong, *Adv. Sci.* **2022**, 9, 2105550.
- [59] B. Burger, P. M. Maffettone, V. V. Gusev, C. M. Aitchison, Y. Bai, X. Wang, X. Li, B. M. Alston, B. Li, R. Clowes, N. Rankin, B. Harris, R. S. Sprick, A. I. Cooper, *Nature* **2020**, 583, 237.
- [60] M. Abolhasani, E. Kumacheva, *Nat. Synth.* **2023**, 2, 483.
- [61] T. Chen, C. Guestrin, *Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, San Francisco, CA, August **2016**, p. 785.
- [62] K. A. Brown, S. Brittman, N. Maccaferri, D. Jariwala, U. Celano, *Nano Lett.* **2020**, 20, 2.
- [63] I.-C. Yeh, *Concrete Compressive Strength*, UCI Machine Learning Repository **1998**, <https://doi.org/10.24432/C5PK67>.
- [64] J. S. Delaney, *J. Chem. Inf. Comput. Sci.* **2004**, 44, 1000.
- [65] D. L. Mobley, J. P. Guthrie, *J. Comput. Aided Mol. Des.* **2014**, 28, 711.
- [66] A. Hersey, *ChEMBL Deposited Data Set - AZ\_dataset* EMBL-EBI, **2015**.
- [67] B. Ramsundar, P. Eastman, P. Walters, V. Pande, *Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery and More*, O'Reilly Media, Sebastopol, CA **2019**.
- [68] G. Pilania, A. Mannodi-Kanakithodi, B. P. Uberuaga, R. Ramprasad, J. E. Gubernatis, T. Lookman, *Sci. Rep.* **2016**, 6, 19375.
- [69] L. Ward, A. Dunn, A. Faghaninia, N. E. R. Zimmermann, S. Bajaj, Q. Wang, J. Montoya, J. Chen, K. Bystrom, M. Dylla, K. Chard, M. Asta, K. A. Persson, G. J. Snyder, I. Foster, A. Jain, *Comput. Mater. Sci.* **2018**, 152, 60.