Bridged Clustering in Scientific Research

Peixuan Ye[1[0009−0009−6955−2667]], Yingtong Wu[1[0000−0003−0140−2995]], and Ellen Vitercik[1[0000−0003−4891−1367]]

[1]Stanford University, Stanford CA 94305, USA pxye@stanford.edu

Abstract. We introduce Bridged Clustering, an algorithm that leverages existing unsupervised datasets to help achieve new supervised objectives in scientific research. Applying supervised learning to scientific research often poses the challenge of labeling enough samples to support scalable inference. As an alternative to excessive labeling, our algorithm leverages unlabeled data that is either already available in existing research or easier to collect in general. Bridged Clustering leverages two distinct sets of unlabeled data and a sparse supervised dataset to perform inference. The algorithm operates by independently clustering the input and output feature spaces, then learning a mapping between these clusters using the supervised set. This approach effectively bridges the gap between disparate data sources, enhancing predictive performance without needing extensive labeled data. We demonstrate the efficacy of Bridged Clustering in a biological context, where it successfully infers genetic information of leaf samples from their morphological traits. In general, Bridged Clustering offers a robust framework for utilizing available unlabeled data to support new inference objectives in scientific research, especially where labeled data is scarce.

Keywords: Clustering · Unsupervised Learning · Taxonomy

1 Introduction

In scientific research, it is crucial to leverage existing datasets. Through careful processing, data collected in past research for different objectives can be re-utilized to answer new research questions, allowing for compounded productivity in cross-disciplinary scientific research [3, 9].

This paper introduces a Machine Learning algorithm that creates a "bridge" between datasets. Specifically, in the scientific setting where researchers aim to relate a set of features $\mathcal{Y}$ to another set of features $\mathcal{X}$, the Bridged Clustering algorithm associates separately available $\mathcal{Y}$-specific data with $\mathcal{X}$-specific data to yield the prediction.

The $\mathcal{Y}$- and $\mathcal{X}$-specific data may come from separate sources of past studies where researchers are only interested in a subset of the feature-set. In addressing a new research problem that involves inferring $\mathcal{Y}$ from $\mathcal{X}$, the $\mathcal{X}$-specific data is seen as unlabeled input data, and the $\mathcal{Y}$-specific data is unsupervised output data disassociated with inputs.

Without leveraging unsupervised datasets, to learn a predictive function from $\mathcal{X}$ to $\mathcal{Y}$ would require a sizable supervised set containing fully-labeled samples. In many research contexts, labeling every sample of interest is expensive and unrealistic at scale [1]. The main contribution of Bridged Clustering is using unlabeled input and output data to augment the predictive ability of a small supervised set of examples, obtaining inferential accuracy on par with a larger labeled set.

Besides obtaining unsupervised data from existing research, the algorithm also facilitates research where data is newly collected. In many settings, unlabeled examples are significantly easier to obtain than labeled ones [6]. To study a population of interest, it is typically easier to first collect an unlabeled set of samples, and then separately collect a sample-independent set of labels. The collection of unsupervised sets can be completely independent of each other as long as they are sampled from the same population of interest, allowing flexibility in data sourcing.

To motivate this problem, our paper considers a setting where the algorithm can help biologists infer genetic information of samples from their morphological structures. Since researchers agree that it is cost-intensive to examine genetics of every available morphological sample, there is usually scarce fully-supervised data with both morphological and genetic information. Data of leaf morphology alone is often more available, as with general genetic data unspecific to morphological samples [4, 7]. Using these two sizable but unsupervised datasets, we can apply the Bridged Clustering algorithm to augment the predictive ability of the small supervised set.

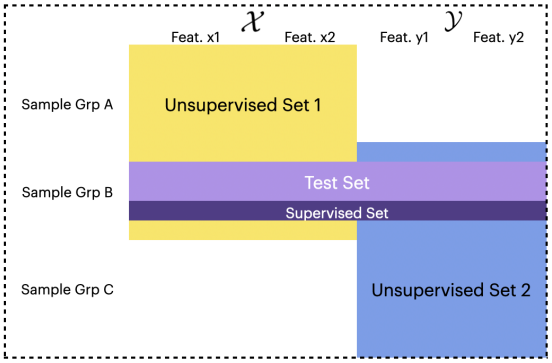


Fig. 1: General Dataset Structure for Bridged Clustering

The inferential function learned by the Bridged Clustering allows biologists to infer rough genetic information from quick examinations of leaf morphology. This is an important application because biologists often have to extrapolate the rough genetic composition of a sample to consider the conservational value

of the species in the area, and conducting iterative genetic testing would be cost-inefficient, making morphological examination a valuable alternative.

2 Related Work

2.1 Semi-supervised Learning

Our algorithm builds upon the semi-supervised learning (SSL) paradigm, which is to learn from both labeled and unlabeled data [11]. The fundamental assumption of SSL is that unlabeled data share an underlying structure with labeled data, enabling the model to infer meaningful patterns from unlabeled data that improve prediction accuracy. A consensus in the research community is that SSL-based methods is crucial for modern machine learning applications where labels are scarce or expensive to obtain [1].

While many SSL methods focus on enhancing features within the input space [10], Bridged Clustering extends this paradigm by clustering the output feature space as well. Similar to how clustering in the input space enables unlabeled data to meaningfully depict the feature space of the population, clustering the output labels gives a more robust representation of the output space.

2.2 Co-training

The idea of using separate unlabeled data to augment a small supervised set is reminiscent of the co-training model [2]. Co-training first learns a separate classifier for each conditionally independent feature space (corresponding to input space and output space in our model). The most confident predictions of each classifier on the unlabeled data are then used to iteratively construct additional labeled training data. In a graphical formulation, as long as every connected component contains a labeled point, the whole dataset can ultimately be labeled by the end of co-training.

Co-training reveals several important facts for our model.

1. Conditional independence of complementary feature spaces allows unlabeled data to provide inferential value.
2. Each connected component in the graph corresponds to one variable, which is the target label for co-training model and the latent variable for Bridged Clustering.

However, co-training and Bridged Clustering address fundamentally different problems. While Co-training aims to infer the target value from both feature spaces, Bridged Clustering seeks to predict the value of one feature (output) from the other feature (input), using the common variable as a latent "bridge" – not as a target.

Algorithm 1 Bridged Clustering Algorithm

Clustering in $\mathcal{X}$:
Apply a clustering algorithm to $\mathcal{X}$ to obtain cluster assignments $\mathcal{C}_{\mathcal{X}}$.
Clustering in $\mathcal{Y}$:
Apply a clustering algorithm to $\mathcal{Y}$ to obtain cluster assignments $\mathcal{C}_{\mathcal{Y}}$.
Bridge Learning:
Using the supervised set $\mathcal{S}$, learn the mapping $A_{x\to y}$ between clusters in $\mathcal{X}$ and clusters in $\mathcal{Y}$.
Prediction:
for each sample x_i in $\mathcal{X}$ **do**
    Assign x_i to a cluster $c_x = \mathcal{C}_{\mathcal{X}}(x_i)$.
    Find corresponding cluster in $\mathcal{Y}$: $c_y = A_{x\to y}(c_x)$.
    Predict $\hat{y}_i$ as the centroid of cluster c_y in $\mathcal{Y}$.
end for
return Predicted labels $\hat{\mathcal{Y}}$

3 Problem Formulation

Let $\mathcal{D}$ be the underlying population distribution over $\mathcal{X} \times \mathcal{Y}$. We draw the following three datasets independently from $\mathcal{D}$:

- **Input Features Set** $\mathcal{X} = \{x_1, x_2, \ldots, x_{|\mathcal{X}|}\} \sim \mathcal{D}|_X$
- **Output Labels Set** $\mathcal{Y} = \{y_1, y_2, \ldots, y_{|\mathcal{Y}|}\} \sim \mathcal{D}|_Y$.
- **Sparse Supervised Set** $\mathcal{S} = \{(x'_i, y'_i)\}_{i=1}^{k}\} \sim \mathcal{D}$

Using the three sets of samples, the Bridged Clustering algorithm aims to accurately estimate the missing labels $\hat{\mathcal{Y}} = \{\hat{y}_1, \ldots, \hat{y}_{|\mathcal{X}|}\}$ for samples in $\mathcal{X}$.

3.1 Critical Assumptions

The algorithm relies on the critical assumption that there exists a latent space $\mathcal{T} = \{t_1, t_2, \ldots, t_j\}(j \ll |\mathcal{X}|, |\mathcal{Y}|)$, for which we can learn mappings $\mathcal{X} \to \mathcal{T}$ and $\mathcal{Y} \to \mathcal{T}$. In other words, as we run an unsupervised clustering algorithm over $\mathcal{X}$, we obtain labels in terms of $\mathcal{T}$, and running the clustering algorithm over $\mathcal{Y}$ yields labels in the same space $\mathcal{T}$.

To see how this assumption is useful, we return to the biological setting. Morphological data, encoded as feature vectors in $\mathcal{X}$, would form individual clusters in $\mathcal{X}$ that roughly correspond to individual leaf species, since same-cluster leaves that share similar morphological traits likely belong to the same species. The same applies to genetics data in $\mathcal{Y}$, where samples of the same species likely fall into the same genetic cluster. In this scenario, $\mathcal{T} : \{t_1, t_2, \ldots, t_j\}$ corresponds to the series of underlying species in the population.

Notice that this setting does not require researchers to have knowledge of different species, as the space $\mathcal{T}$ is a latent assumption that we make based on the clustering properties in $\mathcal{X}$ and $\mathcal{Y}$. Our algorithm is useful for settings where the $\mathcal{T}$ space is unknown, for instance when field researchers sample from a new

site, where there might be unknown species or otherwise uncertain delimitation of species.

Obtaining mappings $\mathcal{X} \leftrightarrow \mathcal{T}$ and $\mathcal{Y} \leftrightarrow \mathcal{T}$, we then use the small supervised set to associate clusters in $\mathcal{X}$ with clusters in $\mathcal{Y}$. The cluster centroids in $\mathcal{Y}$ are used as predictions for datapoints that fall into the corresponding clusters in $\mathcal{X}$.

Another useful assumption is that for any sample, its feature in $\mathcal{X}$ and label in $\mathcal{Y}$ are conditionally independent given the latent variable in $\mathcal{T}$. For a leaf sample, we can assume that its genetic components are mostly conditionally independent of its morphological features, given that the leaf belongs to a specific species. (That is, genetic variation between same-species samples is small, and is minimally correlated with the same-species morphological variation.)

Postulating conditional independence, we can see one possible advantage of Bridged Clustering over classical regression methods. Bridged Clustering is more robust to the sample variability introduced by conditional independence.

3.2 Optimization Objectives

In essence, the Bridged Clustering Algorithm transforms the basic predictive objective (given input x_i, minimize $|\hat{y}_i - y_i|$) into two clustering objectives (ensure effective clustering in $\mathcal{X} \leftrightarrow \mathcal{T}$ and $\mathcal{Y} \leftrightarrow \mathcal{T}$) and an mapping objective (ensure correct $\mathcal{X} \leftrightarrow \mathcal{Y}$ cluster association in $\mathcal{T}$ space).

Clustering Objectives Let $\mathcal{C}_{\mathcal{X}}$ be the function that assigns each feature vector $x_i \in \mathcal{X}$ to one of j clusters, and let $\mathcal{C}_{\mathcal{X}}^{(k)}$ denote the set of feature vectors assigned to cluster k.

The clustering objective for the feature space is given by:

$$\min_{\mathcal{C}_{\mathcal{X}}} \sum_{k=1}^{j} \sum_{x_i \in \mathcal{C}_{\mathcal{X}}^{(k)}} \|x_i - \frac{1}{|\mathcal{C}_{\mathcal{X}}^{(k)}|} \sum_{x_\ell \in \mathcal{C}_{\mathcal{X}}^{(k)}} x_\ell\|^2.$$

Let $\mathcal{C}_{\mathcal{Y}}$ be the function that assigns each output vector $y_i \in \mathcal{Y}$ to one of j clusters, and let $\mathcal{C}_{\mathcal{Y}}^{(k)}$ denote the set of output vectors assigned to cluster k.

The clustering objective for the output space is given by:

$$\min_{\mathcal{C}_{\mathcal{Y}}} \sum_{k=1}^{j} \sum_{y_i \in \mathcal{C}_{\mathcal{Y}}^{(k)}} \|y_i - \frac{1}{|\mathcal{C}_{\mathcal{Y}}^{(k)}|} \sum_{y_\ell \in \mathcal{C}_{\mathcal{Y}}^{(k)}} y_\ell\|^2$$

.

Mapping Objective With supervised set $\mathcal{S}$, we determine the most probable correspondence between clusters in $\mathcal{X}$ and $\mathcal{Y}$ that maximizes the cluster-to-cluster association accuracy. We seek the association function $A_{x \to y}$ such that:

$$\max_{A_{x \to y}} \sum_{(x_i, y_i) \in S} \mathbb{1}\{A_{x \to y}(\mathcal{C}_{\mathcal{X}}(x_i)) = \mathcal{C}_{\mathcal{Y}}(y_i)\}$$

.

3.3 Graphical Formulation

Now we will formulate the Bridged Clustering model graphically. Given nodes for sample inputs $x_1, x_2, \ldots, x_{|\mathcal{X}|}$, and nodes for sample outputs $y_1, y_2, \ldots, y_{|\mathcal{Y}|}$, perform clustering for the input and output vectors respectively. Instantiate j nodes for cluster labels $\mathcal{C}_{\mathcal{X}}$, and each edge $(x_a, \mathcal{C}_{\mathcal{X}b})$ indicates that the ath input vector is assigned to the bth input cluster. Instantiate j nodes for cluster labels $\mathcal{C}_{\mathcal{Y}}$, with edges $(y_a, \mathcal{C}_{\mathcal{Y}b})$ representing output cluster assignments.

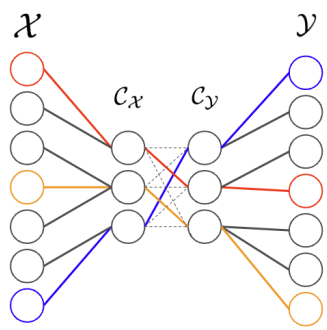


Fig. 2: Building connected components through supervised (colored) examples, simplified for illustration.

Following the two clustering procedures, we should have $2j$ connected components, each representing an input or output cluster. Using the small supervised set, We build j edges $(\mathcal{C}_{\mathcal{X}a}, \mathcal{C}_{\mathcal{Y}b})$ in a way such that the greatest number of supervised samples $(x_i, y_i) \in \mathcal{S}$ could have x_i and y_i in the same connected component, halving the number of connected components, as shown in Figure 2.

4 Experiments

In the experimental setting, we use the Bridged Clustering algorithm to predict genetic features of Quercus oak leaf samples from the sample's morphological traits. More specifically, we test the accuracy in which our model predicts the key dimensions of genetic variation of biallelic single nucleotide polymorphisms (SNPs) in these leaf samples.

4.1 Datasets and Feature Selection

In the general setting for Bridged Clustering, we are given two unsupervised training sets with an overlapping subset, from which we sample a supervised set and a test set.

For our experiments, we are given a morphological dataset containing 572 samples, each with 19 morphological features, and another genetics dataset containing 179 samples, each with 6 PC features, PCs representing a genetic variation of biallelic single nucleotide polymorphisms (SNPs). There are 111 oak leaf samples that are present in both datasets, complete with both morphological and genetic features. Of these samples, we randomly sample a isolated test subset, and another small supervised subset. Every sample that is not in the test set is assigned to the training group, including supervised and unsupervised. All of these samples belong to one of 3 members of the Quercus family: Quercus acerifolia (QA), Quercus shumardii (QS), or Quercus rubra (QR).

To enable efficient clustering, we should reduce the dimensions of both feature sets. Methods vary for different applications of the algorithm, but here we select features based on how well they form clusters corresponding to the latent variable of interest: species. Biologists have already hand-labeled the species categorization information for the samples in the dataset (3 possible species: QA, QR, QS). Through iterative testing, we sampled the best set of features to include into the feature vectors – the criteria is that, with this set of features, our algorithm has the highest probability of assigning samples hand-labeled as the same species to the same cluster. This external criteria of clustering quality can be measured in terms of cluster purity or Normalized Mutual Information [8].

Posing the ground-truth species information against the empirical cluster assignment, we measure how well differently-configured clustering algorithm recognizes species as a latent factor. With cluster quality as a standard, we converge on the best configuration of the clustering algorithm: the best subset of input/output features. We ended up selecting 5 morphological features (Terminal extension length, Lateral sinus radius, Lateral lobe distal width, Degree of axillary pubescence, Tree height) and 3 genetics features (PC 1, 2, 3) to be included in our model.

4.2 Clustering Algorithm

Using the same standard of cluster quality, we measured the NMI of different clustering algorithms, and opted the K-means algorithm [5], which achieved NMI = 0.619 for Morphology, and NMI = 0.539 for Genetics.

We apply the K-means clustering algorithm to the 5-feature morphological vectors in the inputs space, and then apply it to the 3-feature genetic vectors in the output space. The results of clustering is shown in Figure 3.

Upon examining the results of clustering, we found a good alignment in cluster and species delineation. Clusters almost exclusively contain samples from one species, and same-species samples are almost always assigned to the same cluster. That means we have established a reasonably reliable $\mathcal{X} \leftrightarrow \mathcal{T}$ and $\mathcal{Y} \leftrightarrow \mathcal{T}$ as we treat species classification as the latent variable $\mathcal{T}$.

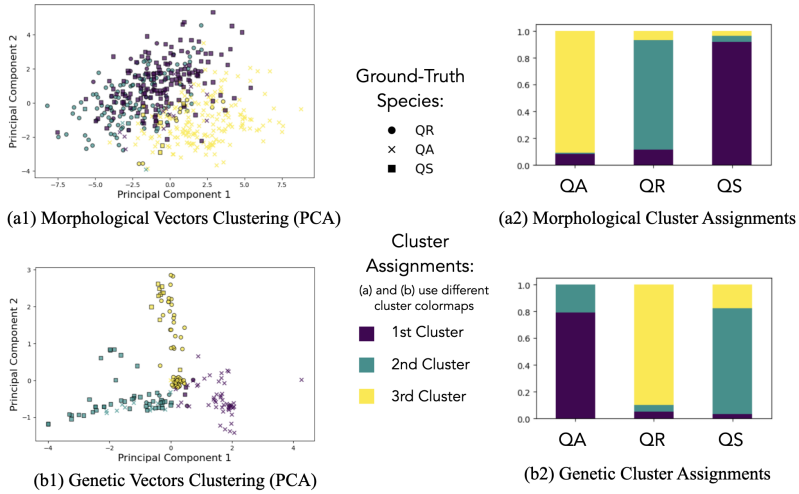


Fig. 3: Using Principle Component Analysis, (a1) and (b1) visualize the clustering results in the morphological input feature space and the genetics output feature space respectively. The clustering quality for the morphological space (a2) and the genetics space (b2) is measured in terms of the cluster assignment consistency for different species.

4.3 Bridging Clusters

To learn the mapping between input clusters and output clusters, we review the supervised samples that have been assigned to an input and output cluster each. For every supervised sample and its two cluster affiliations, we increase the confidence that these two clusters are mutually associated. In the end, we take the most probable 1-to-1 mapping between the input and output clusters.

Since we have the ground truth for the latent variable $\mathcal{T}$, we can assess the reliability of our algorithm in retrieving the correct cluster associations. The rate of success is a function of the number of supervised samples. As shown in Figure 4, if we collect more than 10% of the 111 fully-labeled examples into our supervised set, the algorithm retrieves the correct cluster associations with probability $> 97\%$, and collecting 20% ensures almost prefect accuracy.

4.4 Inference Results

The algorithm is finally assessed by its accuracy of inference. We run the Bridged Clustering algorithm, and for random leaf samples, we predict the 3 genetic features from their morphological features. For every random sample, we measure the distance between its algorithm-predicted genetic features and the actual feature value – the smaller the distance, the more accurate our algorithm. We run this experiment for different sizes of the supervised set. For each experiment, we

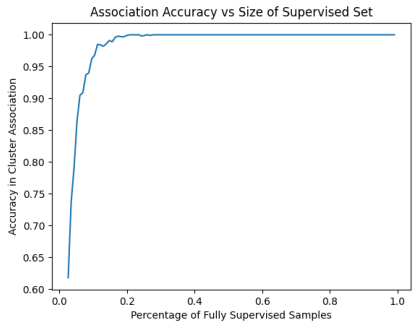


Fig. 4: Accuracy for cluster-cluster mapping, defined as the percentage of successful randomized trials

Table 1: Mean of euclidean distances between predicted and true genetic coordinates.

Supervised%	**Bridged Clustering**	KNN(k=1)	KNN(k=2)	KNN(k=3)	Linear Reg
5%	**11.77**	13.43	12.45	12.36	14.16
10%	**10.71**	12.53	11.58	11.30	17.27
15%	**10.37**	11.88	11.19	10.86	42.27
20%	**10.35**	11.71	10.90	10.61	28.48
25%	**10.34**	11.38	10.73	10.48	16.86
30%	10.22	10.89	10.24	**10.05**	14.46

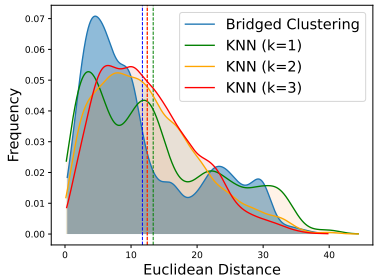
accumulate the test results from 500 random trials, and observe the distribution of euclidean distances between the predicted and true genetic coordinates.

For our primary baseline, we use K-Nearest-Neighbor Regression, a method that solely relies on the fully-supervised portion of the data. For every test sample, the KNN model searches for its K closest euclidean neighbors in the input space (in morphological coordinates), and returns the average of the genetic coordinates of these neighbors as prediction value. We tested KNN on the full feature set, instead of the preprocessed feature set that we curated for Bridged Clustering, addressing the concern of unfairly tuning the data in favor for the Bridged Clustering method at the expense of baselines.

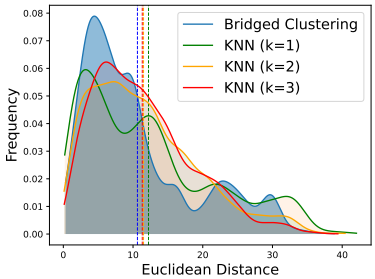
As a supplement to the KNN baseline, we also experimented with Linear Regression. But as discovered, there is very limited linear correlation between the morphological and genetic features of samples in our datasets, indicating conditional independence given species.

The experiments yield results as shown in Table 1. On average, the genetics predicted by the Bridged Clustering algorithm are the closest to the ground-truth values, as compared to the KNN predictions. Our predictions also has smaller variances, suggesting that our algorithm returns a more consistent estimate (See Figure 5). These observations hold while the number of supervised examples is scare, which in this case is less than 25% of all labeled samples.

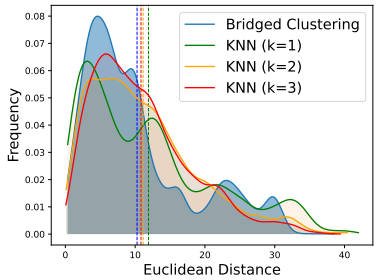
As the size of the supervised set grows large relative to the entire dataset, the unsupervised samples provide less inference power and Bridged Clustering does not outperform all baseline methods, suggesting a limitation of our algorithm.



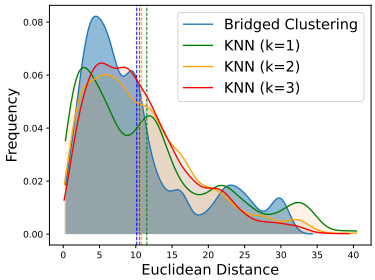
(a) 5% Supervision



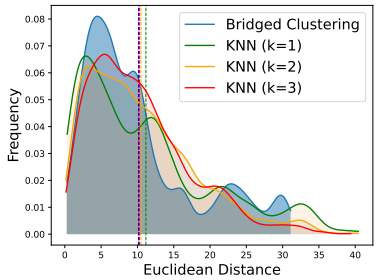
(b) 10% Supervision



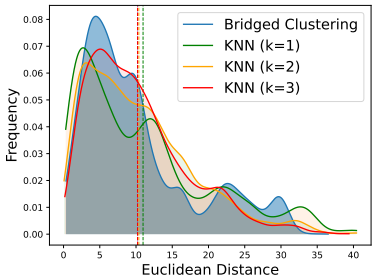
(c) 15% Supervision



(d) 20% Supervision



(e) 25% Supervision



(f) 30% Supervision

Fig. 5: Distribution of euclidean distances between predicted and true gene coordinates: a direct comparison between Bridged Clustering and KNN methods.

References

1. Balcan, M.F.F., Sharma, D.: Data driven semi-supervised learning. Advances in Neural Information Processing Systems **34**, 14782–14794 (2021)
2. Bartlett, P., Mansour, Y., Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory. pp. 92–100 (1998). https://doi.org/10.1145/279943.279962
3. Castaneda, D.I., Cuellar, S.: Knowledge sharing and innovation: A systematic review. Knowledge and Process Management **27**(3), 159–173 (2020)
4. Lexer, C., Widmer, A.: The genic view of plant speciation: recent progress and emerging questions. Philosophical Transactions of the Royal Society B: Biological Sciences **363**(1506), 3023–3036 (2008)
5. Lloyd, S.: Least squares quantization in pcm. IEEE Transactions on Information Theory **28**(2), 129–137 (1982)
6. Ratner, A., Bach, S.H., Ehrenberg, H., Fries, J., Wu, S., Ré, C.: Snorkel: Rapid training data creation with weak supervision. In: Proceedings of the VLDB endowment. International conference on very large data bases. vol. 11, p. 269. NIH Public Access (2017)
7. Stein, E.D., Martinez, M.C., Stiles, S., Miller, P.E., Zakharov, E.V.: Is dna barcoding actually cheaper and faster than traditional morphological methods: results from a survey of freshwater bioassessment efforts in the united states? PloS one **9**(4), e95525 (2014)
8. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research **3**, 583–617 (01 2002). https://doi.org/10.1162/153244303321897735
9. Tenopir, C., Allard, S., Douglass, K., Aydinoglu, A.U., Wu, L., Read, E., Manoff, M., Frame, M.: Data sharing by scientists: practices and perceptions. PloS one **6**(6), e21101 (2011)
10. Van Engelen, J.E., Hoos, H.H.: A survey on semi-supervised learning. Machine learning **109**(2), 373–440 (2020)
11. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. Springer Nature (2022)