Legg *et al. Hum. Cent. Comput. Inf. Sci.* (2019) 9:5
https://doi.org/10.1186/s13673-019-0167-8

Human-centric Computing and Information Sciences

RESEARCH Open Access

Visual analytics for collaborative human-machine confidence in human-centric active learning tasks

Phil Legg*, Jim Smith and Alexander Downing

*Correspondence:
Phil.Legg@uwe.ac.uk
Department of Computer Science and Creative Technologies, University of the West of England, Bristol, UK

Abstract

Active machine learning is a human-centric paradigm that leverages a small labelled dataset to build an initial weak classifier, that can then be improved over time through human-machine collaboration. As new unlabelled samples are observed, the machine can either provide a prediction, or query a human 'oracle' when the machine is not confident in its prediction. Of course, just as the machine may lack confidence, the same can also be true of a human 'oracle': humans are not all-knowing, untiring oracles. A human's ability to provide an accurate and confident response will often vary between queries, according to the duration of the current interaction, their level of engagement with the system, and the difficulty of the labelling task. This poses an important question of how uncertainty can be expressed and accounted for in a human-machine collaboration. In short, how can we facilitate a mutually-transparent collaboration between two uncertain actors—a person and a machine—that leads to an improved outcome? In this work, we demonstrate the benefit of human-machine collaboration within the process of active learning, where limited data samples are available or where labelling costs are high. To achieve this, we developed a visual analytics tool for active learning that promotes transparency, inspection, understanding and trust, of the learning process through human-machine collaboration. Fundamental to the notion of confidence, both parties can report their level of confidence during active learning tasks using the tool, such that this can be used to inform learning. Human confidence of labels can be accounted for by the machine, the machine can query for samples based on confidence measures, and the machine can report confidence of current predictions to the human, to further the trust and transparency between the collaborative parties. In particular, we find that this can improve the robustness of the classifier when incorrect sample labels are provided, due to unconfidence or fatigue. Reported confidences can also better inform human-machine sample selection in collaborative sampling. Our experimentation compares the impact of different selection strategies for acquiring samples: machine-driven, human-driven, and collaborative selection. We demonstrate how a collaborative approach can improve trust in the model robustness, achieving high accuracy and low user correction, with only limited data sample selections.

Keywords: Visual knowledge discovery, Data clustering, Active machine learning, Human-machine collaboration

Introduction

Machine learning involves the process of training a classifier to distinguish classes, using a large collection of high-quality accurately-labelled data samples. A variety of benchmark datasets exist online that include manually-crafted annotations and class labels, however for many real-world applications the process of collecting and annotating such large datasets by hand is neither practical nor feasible. As a motivating example, the popular MNIST digit classification dataset [1] consists of 60,000 training images and 10,000 test images of digits in the range 0 to 9. A human working at a response rate of 1 second/image would require 1000 min (over 16 h) to label this training data. Even a 'skilled' labeller with a response rate of 0.5 second/image would require over 8 h to perform this task, working continuously with no breaks. Clearly this is impractical for any human to perform, and the reliability of labels would most likely reduce over time. This example clearly highlights the issues of time and effort, and also human fatigue and reliability. Another factor to consider is understanding how the training samples contribute towards the accuracy of the classifier. Many batch learning systems take the approach of 'more data is better', without acknowledging that some samples that may be included could reduce performance (e.g., poorly-written digits in the MNIST example), or cases where the algorithm could be fooled by the input. Equally, since the learning features are typically just a subset of possible descriptors drawn from the real-world artefact, then it is vital to understand rapidly if the chosen features adequately represent those artefacts, to avoid extensive effort creating datasets that may never have sufficient discriminative power to construct a suitable classifier [2].

Active Learning is an area of research that aims to address human-machine collaboration in machine learning. Initially, the machine is trained on a small sample of labels by the human. As new unlabelled samples are observed, the machine can either provide a classification if it is suitable confident to do so, else it can query the human for a class label if it is unconfident [3]. Over time, the performance of the machine classifier improves as the human answers each query. Ideally, the active learning approach would perform as well as a batch learning approach, yet require significantly fewer training samples and therefore less human intervention. However, this poses several challenges. For example, how should the machine decide which samples to query with the user such that the best gain can be achieved? Also, how can the machine ensure that this selection does not introduce a bias in the training sample, such that a generalisable classifier fails to perform well? A crucial, but often overlooked assumption is that the user will be able to provide the correct label at all times. However, what if they can not, or what if they can provide a label but they are not fully confident in their decision—perhaps because the features stored are noisy or inadequately capture the artefact (for example, ambiguous hand-writing in the MNIST case).

In this work, we demonstrate the benefit of human-machine collaboration within the process of active learning, where limited data samples are available or where labelling costs are high. To achieve this, we develop a visual analytics tool called *Acti*VAte (Visual Analytics in *Acti*ve Machine Learning) that promotes transparency, inspection, understanding and trust, of the learning process through human-machine collaboration. Fundamental to the notion of confidence, both parties can report their level of confidence during active learning tasks using the tool, such that this can be used to inform learning.

Human confidence of labels can be accounted for by the machine, the machine can query for samples based on confidence measures, and the machine can report confidence of current predictions to the human, to further the trust and transparency between the collaborative parties. This conception of human-machine collaboration makes the conventional notion of a 'user' or a 'supervisor' somewhat redundant and outdated, as these terms are suggestive of a one-way flow of information rather than a dialogue between collaborative partners. Within this work, when we refer to a 'user' this could, and should, be considered more precisely as a 'collaborator' that engages in a two-way dialogue with the system. Using the *Acti*VAte tool, we present an empirical study to investigate classifier performance under different sample selection strategies (machine-driven, user-driven, and collaborative), to assess classifier accuracy, and human re-labelling (which can be considered a form of intervention or correction). We also investigate different labelling techniques, and show how confidence can inform data augmentation techniques to improve classifier robustness. We show that the collaborative selection yields both high accuracy and low user correction, emphasising the effectiveness of the collaborative visual analytic workflow. The contributions of this work are summarised as follows:

- We demonstrate the benefit of human-machine collaboration using a novel visual analytics tool called *Acti*VAte that facilitates transparency and trust in active machine learning via two-way collaborative dialogue on data attributions between human and machine.
- We integrate the notion of confidence within an active learning framework, for both human and machine to express personal confidence in data attributions, providing mutual transparency for other parties in further decision-making processes.
- We incorporate human confidence weighting as part of a data augmentation labelling scheme, which we find can improve the robustness of the classifier in the presence of ambiguous or mis-classified data samples compared to traditional data augmentation techniques.
- We conduct an empirical study using the *Acti*VAte system, to compare the effectiveness of machine, human, and collaborative selection strategies. We measure performance against four different labelling schemes, and show how collaborative selection can achieve high classifier accuracy whilst also minimising the corrective effort required by the human's collaborative interactions.

Related work

To address related works for our study, we review the use of active machine learning techniques and we also explore how visual analytics has been used for improved machine learning applications.

Attenberg and Provost [4] look at the difficulties of applying active learning in practice, suggesting that the adoption of the technique is still relatively low despite the promises of reducing the cost and effort of machine learning development. Raghavan et al. [5] extend traditional active learning to include feedback on features in addition to labelling instances, suggesting that the human can be better utilised as a feature detector for re-weighting important features. Donmez et al. [6] discuss *Pro-active learning*, recognising that the human is not an all-knowing oracle and suggesting that the machine should be able to learn when the user is imperfect, not able to answer, or when multiple users provide conflicting opinions. Lin et al. [7] describe *Re-active learning* as an extension of active learning where relabelling is allowed. This may enable multiple users to refine or correct a previous label, which may be suitable in cases where labels are crowdsourced, or where users may have low confidence in their initial response. This presents another benefit for active learning, since the human performing the task of labelling may be incorrect (either knowingly or unknowingly). Lughofer et al. [8] show how taking account of users' labelling confidence can reduce prediction errors by up to 50% in an online learning task. Smith et al. [9] study how user confidence could be predicted from how users perform visual decision-making tasks, which could further human and machine collaboration, either to only make use of training labels that a human user is confident are correct, or as a means of identifying unconfident cases that need further investigation.

Cook et al. [10] consider the use of machine learning and visual analytics for business decision-support. They look at how profitability and uncertainty can be analysed and predicted based on a collection of features that are hand-crafted from company records. Choo et al. [11] describe a visual testbed for dimensionality reduction and clustering, demonstrating how high-dimensional datasets can be explored using visual analytics tools. Their tool shows how different classes of data (e.g., digit datasets) may vary in terms of the clustering region and spread of the data in the reduced space. Their tools rely on knowing the class labels for the underlying datasets. Krause et al. [12] propose a visual analytics system for understanding feature selection in machine learning tasks. In particular, they look at interactive partial dependence diagnostics to study how individual features affect the overall prediction, as well as understanding how and why specific data points are predicted, and support for tweaking feature values to observe how the prediction responds. They demonstrate their approach for detecting the onset of diabetes from electronic medical records. Legg et al. [13] propose the use of active learning to facilitate the understanding and analysis of video search, and as a means of refining search results based on uncertain sketch-based queries. Legg et al. [14, 15] also demonstrate how user feedback can be incorporated within an active learning approach for insider threat detection. Endert et al. [16] looked at the integration of machine learning and visual analytics, illustrating how visual analytics provides a practical medium for supporting interactive and iterative machine learning applications that can offer benefits over traditional batch learning tasks.

Liu et al. [17] present a survey of how visual analytics can be used for better analysis of machine learning models. They focus on three categories of application: understanding, diagnosis, and refinement. Examples of work covered in the survey include Ren et al. [18], who use a parallel coordinates view coupled with a pixel-based visualisation to support performance diagnosis of multi-class classifiers within a single visualisation, with an aim to reduce cognitive load during analysis.

There has been much interest recently in how visual analytics can be used for interpretation and understanding of machine learning. Phillips et al. [19] describe interpretable active learning, working with the Local Interpretable Model-agnostic Explanations framework (LIME) to provide explanations for active learning recommendations. They demonstrate how LIME can be used to generate locally faithful explanations for an active learning strategy, and how these explanations can be used to understand how different models and datasets explore a problem space over time. Rubens and Sugiyama [20] study influence-based collaborative active learning, where multiple users may work together to provide a result, such as a group-based recommendation. Yang and Loog [21] also study active learning using uncertain information. With regards to understanding deep neural network structures, Liu et al. [22] look at understanding the training process of deep generative models, Ming et al. [23] focus on understanding the hidden memories of recurrent neural networks, and Kahng et al. [24] propose ActiVis as a visual analytics tool for exploring large deep neural networks. Olah et al. [25] discuss the building blocks of interpretability within convolutional neural networks. Within the Computer Vision community there is much interest in visualising neural network structures to improve understanding, given the image-based nature of inputs and the semantic representation of hidden nodes. As noted by the increase of works combining visualisation and machine learning, there is much scope for how visualisation can improve the understanding of machine learning models [24, 26].

There are a number of different approaches for how users may interact with a learning process, beyond simply providing labels to a classifier. Brezeal and Thomaz [27] demonstrated that providing more transparency and multi-channel communication increased user engagement and reduced learning times in a robotic Reinforcement Learning Task. Within the fields of design and optimisation via techniques such as Interactive Evolutionary Algorithms [28], the importance of interfaces that support people to take a more active role in guiding search, and the effects on facilitating and prolonging engagement has been extensively studied e.g. [29–31]. Amershi et al. survey the role of humans in interactive machine learning [32], looking at different forms of user interaction, and also considering how people can provide more than just data labels. Sacha et al. analyse how visual interaction can help in dimensionality reduction for understanding characteristics of the data [33]. Aung et al. study labelling uncertainty based on student engagement, and discuss the notion of soft and hard labels based on individual and summary statistics of the labeller's responses [34]. Smith et al. study how user confidence in visual decision making tasks can be assessed, based on eye gaze fixations and interactions [35]. Bernard et al. propose a visual interactive labeling process called VIAL [36]. They demonstrate VIAL on image recognition in CCTV footage, however with a much more complex and configurable interface. Our approach addresses different objectives, such as illustrating the selection of samples from a pool of unlabelled cases, the spatial drag-and-drop labelling of samples to extend the physical analogy of classifying objects into groups, and the positioning of objects to convey a level of confidence or association within the groups.

Confidence-based active learning framework

Traditional batch learning systems that are pre-trained on large datasets have shortcomings for practical usage. Firstly, end-users are not engaged in the learning process, and so can not study how they operate, and whether to trust the reliability of the system (e.g., understanding why the machine made a particular decision, and whether it was

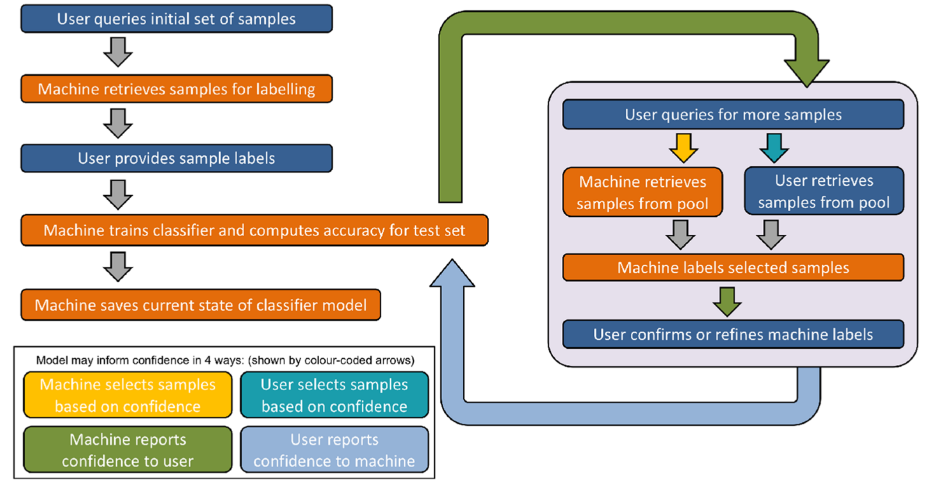

User queries initial set of samples
Machine retrieves samples for labelling
User provides sample labels
Machine trains classifier and computes accuracy for test set
Machine saves current state of classifier model
User queries for more samples
Machine retrieves samples from pool
User retrieves samples from pool
Machine labels selected samples
User confirms or refines machine labels
Model may inform confidence in 4 ways: (shown by colour-coded arrows)
Machine selects samples based on confidence
User selects samples based on confidence
Machine reports confidence to user
User reports confidence to machine

Fig. 1 Confidence-based visual analytic loop for active machine learning. User actions are shown in blue, and machine actions are shown in orange. The outer stages show the initial phase of operation, then the inner loop shows the subsequent actions for each iteration of training. Coloured arrows correspond to 4 possible forms of confidence that can be utilised from the current classifier model: selection based on confidence, and reporting of confidence, for both user and machine. Central to the process, as the user queries or selects further samples, and also confirms or refines machine-generated labels, this contributes towards their knowledge generation of the system process

the correct decision). Secondly, if the distribution of observations changes over time then there may well be need for re-training of the system (e.g., malware detection, where an adversary would purposely adapt the distribution of new malware types). Finally, obtaining large representative datasets may be challenging and so a continual learning framework that is initialised on a limited data sample may be of more practical benefit. Here, we propose that an interactive learning system can offer significant improvement to address these shortcomings. In particular, through human-machine collaboration, both parties can inform each other of their levels of confidence during the task, further enhancing the notion of trust in data input quality and classifier performance.

Figure 1 presents a confidence-based visual analytic loop for active machine learning. Initially, the system has no labelled examples to construct a classifier. The user initiates the process and queries the machine to retrieve an initial set of samples from the pool of unlabelled instances. Samples can be selected based on a variety of schemes: randomly, based on a measure of dissimilarity or distance (such that the selected samples exhibit variety and coverage of the overall distribution), or by user-selection from the sample pool. The user can then provide labels for these samples (whilst reporting how confident they are in the assigned label), and the machine can then train the classifier on this sample set to obtain an accuracy score against a test set that is not incorporated in training. Once the machine has developed an initial classifier, the visual analytics loop can be utilised, and confidence of the current model can be exploited further at this stage.

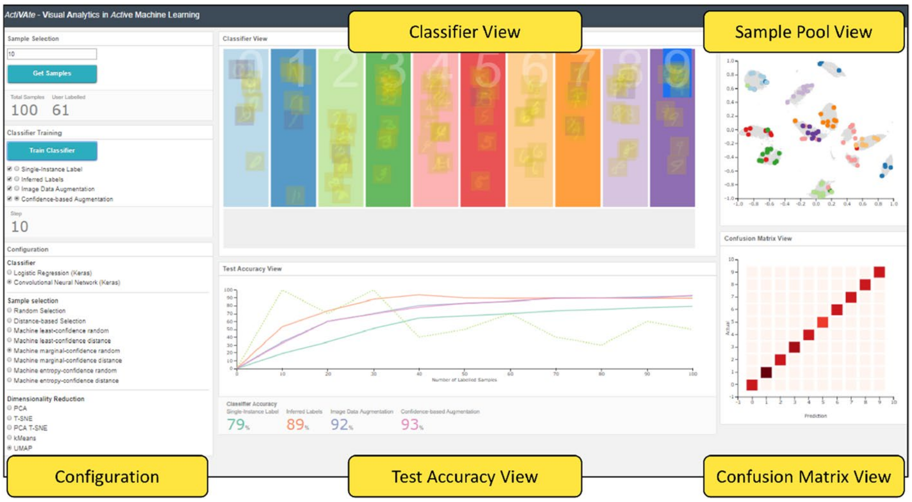
There are four forms where confidence can be utilised within the framework (shown by the colour-coded arrows within the loop in Fig. 1). Firstly, when the user queries for more samples, the machine can use the existing classifier model to identify samples that it is unconfident about. If the machine can identify samples where it is not confident and query these with the user, then over time the machine should become more capable of classifying these cases. In addition, the user may decide to select samples manually—which may be due to either high confidence in being able to offer a label, or because of observing that the machine has low confidence in a given class. The selection of samples can be driven by either the machine, the user, or a combined approach of the two. Following sample selection, the machine will provide a label based on the current classifier model. With each label is an associated confidence (given by the weights in the output layer of the neural network) that can be reported to the user. This allows the user to see what class the machine believes the sample to be, and how confident the machine is in each case. Finally, the user can either confirm or refine the label assign to each of the selected samples, and express a level of confidence back to the machine. With each loop, we would expect the classifier to improve as more samples are labelled. In addition, whilst initially the user may need to refine labels, over time we would expect user refinement to decrease as more machine samples are labelled correctly. With each iteration, user interaction with the system enables a more detailed understanding of how the classifier performance has improved and why; for example, if particular samples have caused the classifier to significantly improve or worsen since the previous iteration, these samples can be studied in greater detail. This interaction may further inform the choice of samples that the user wishes to provide labels for on the next iteration of the loop. At each loop, the classifier state is saved so that this can be deployed as a trained model once the user is sufficiently happy with the performance obtained.

ActiVAte system design

We developed a visual analytic software tool called *Acti*VAte (**V**isual **A**nalytics in *Acti*ve Learning) that is designed to facilitate the process of active machine learning using visual analytics. The software is deployed as a client-server model with a web-based user interface, and is written using Python and Javascript. Key to this is the ability to interface with popular machine learning libraries such as Tensorflow and Keras. The system provides a variety of interactive views to facilitate human-machine collaboration, transparency, and trust, during the iterative process of classifier training.

The system is designed for end-users who may wish to iteratively train a machine learning classifier for a particular task, but who may not necessarily be 'expert' in the field of machine learning or capable of developing their own machine learning applications. Machine learning is attracting attention in many new domains and so there is a need to address this user group who may be working with data but who are not necessarily able to code a machine learning algorithm. The system would also be beneficial to those who wish to inspect the samples used for training a classifier to ensure robustness and discriminative power between classes. To ensure that the system fulfils this purpose, it should:

- Facilitate automated and manual sample selection using various confidence- and distance-based techniques, such that effective training samples can be identified for labelling.
- Be able to infer appropriate labels for unlabelled samples, based on the labelling provided by the user.



Sample Selection
10
Total Samples 100 User Labelled 61

Classifier Training
☑ ○ Single-Instance Label
☑ ○ Inferred Labels
☑ ○ Image Data Augmentation
☑ ● Confidence-based Augmentation
Step 10

Configuration
Classifier
○ Logistic Regression (Keras)
● Convolutional Neural Network (Keras)

Sample selection
○ Random Selection
○ Distance-based Selection
○ Machine least-confidence random
○ Machine least-confidence distance
● Machine marginal-confidence random
○ Machine marginal-confidence distance
○ Machine entropy-confidence random
○ Machine entropy-confidence distance

Dimensionality Reduction
○ PCA
○ T-SNE
○ PCA T-SNE
○ kMeans
● UMAP

Classifier Accuracy
Single-Instance Label 79%
Inferred Labels 89%
Image Data Augmentation 92%
Confidence-based Augmentation 93%

Fig. 2 *ActiV*Ate user interface. Key components include the classifier view for dragging new instances to corresponding classes, sample pool view, which offers a dimensionality reduction view of the unlabelled pool of samples, and the accuracy view that shows the accuracy against the test set at each iteration of training. Configuration options are also available to the user in the left column

- Be able to train a classifier based on labelled samples, and allow the the user to explore the classifier performance to better identify cases of mis-classification.
- Be able to assist in labelling, by predicting sample labels using the available classifier model, such that labelling effort from the user can be minimised.
- Be transparent—facilitating both actors to understand the uncertainty in each other's (mental or machine-learned) models and decisions.
- Provide a dynamic and engaging experience for labelling and training the classifier, such that acceptable accuracy can be achieved from a limited sample set in minimal time compared to batch training.

Overview

The visual analytics interface (Fig. 2) consists of various supporting linked views:

- *Sample pool view*: This panel enables users to visualise the pool of labelled and unlabelled samples based on dimensionality reduction from the original image space to a 2-dimensional scatter plot view. Users can select samples by hand, and can also observe machine sample selection from the pool. The visualisation enables users to assess whether the sample distribution is even across the space, or whether this is uneven and bias towards particular classes. It can also be used to facilitate user understanding of when weak samples are mis-classified (e.g., a 4 that appears within a cluster of 9's in sample space may be a weak example of a 4). However, this can also be informative since it may be this weak sample that is required to improving classifier robustness and further the discriminative features of the classifier.
- *Classifier view*: This panel enables users to provide labels for samples based on drag-and-drop from the unlabelled area (grey) into the respective 10-class coloured regions. Users can associate a level of confidence with their label based on the verti-

cal positioning of the sample within the respective region. Similarly, the machine will report to the user by presenting predicted instances in their respective class regions, positioned based on confidence. The user can then accept the machine prediction or refine it by dragging the sample to the correct region. Samples are shown either as a yellow highlight where the machine has predicted the value, and the user has not acted on the sample, blue where the machine predicted value has been confirmed by the user, and red where the machine predicted value has been corrected by the user. This drag-and-drop approach for sample labelling is akin to real-world document classification where items may be grouped together, and so offers an intuitive representation of the task. It also allows all samples to be 'scattered' in front of the user, to enable them to better compare and contrast samples with each other.

- *Test accuracy view*: This panel indicates the current accuracy of the classifier for each of the training schemes being tested, shown by the coloured lines that correspond to the coloured percentage results. The line plot is updated each time the classifier is trained to reflect the change over time in how the accuracy has improved. The line plot can also give an indication of user effort for each iteration, defined as the number of cases that the user has re-labelled for that iteration. This is scaled as a percentage of samples provided for that iteration of training, and is shown by the dashed line. This reinforces the concept of transparency, to assess how the classifier performance varies over time in accordance to the samples that have been provided.
- *Confusion matrix view*: This panel indicates the current performance of the classifier using a confusion matrix. The confusion matrix shows the correspondence between predicted values and actual values for all cases in the test set, as a colour-scaled matrix. The ideal case is where the predicted values corresponds with the same actual value, giving a diagonal across the matrix. Typically, there will be some misclassifications (e.g., a 4 may be predicted as a 9), and so the confusion matrix allows the user to identify such cases. The combination of both the confusion matrix and the sample pool is designed to further inform user sample selection, and the generation of knowledge on how samples improve the classifier performance.
- *Configuration view*: This panel allows the user to select the number of samples to draw from the unlabelled pool for the next iteration. It also allows the user to train the classifier using different schemes: single-instance labelling, inferred labelling, image data augmentation, and confidence-based augmentation (which we describe in the subsequent section). It also allows the user to configure the classifier type (logistic regression or convolution neural network), the sample selection scheme currently used by the machine, and what reduction technique should be used for the sample pool view. These parameters can also be adjusted during training iterations as desired by the user. It is not expected or required to interact and adjust these parameters, however more advanced users may wish to have access to this configuration.

The visual analytics interface supports five key tasks: (1) representation of unlabelled sample pool; (2) user/machine sample selection; (3) user labelling and confidence feedback; (4) training of the machine classifier; and (5) machine labelling and confidence feedback. The following sections detail each of these tasks, and how the visual analytics interaction can better facilitate these cases.

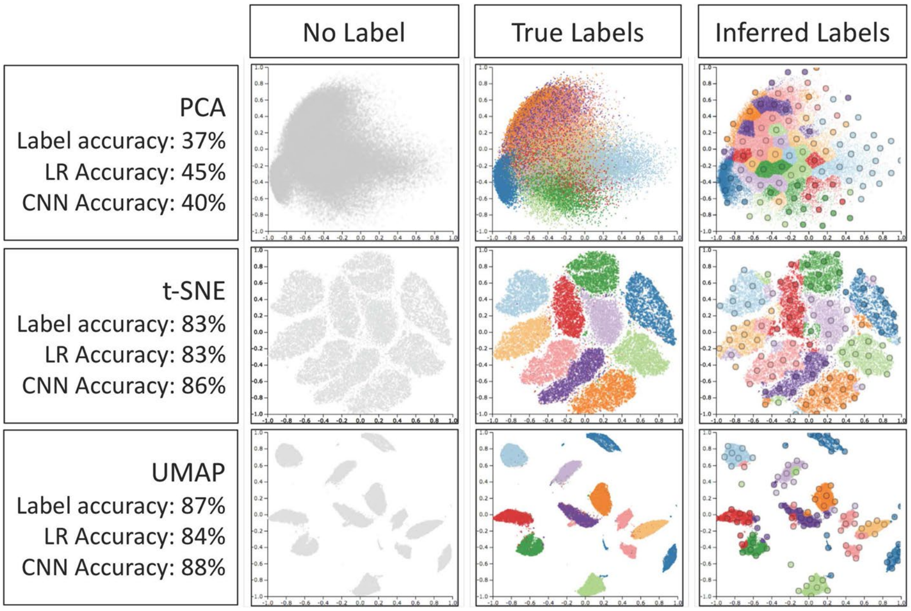


Fig. 3 Dimension reduction techniques for exploring the unlabelled sample pool consisting of all training images. For each row, the unlabelled pool is shown (left), the pool with true labels (middle), and the pool with inferred labels from 100 user-labelled samples (right). Results of labelling accuracy from the inferred scheme, and the accuracy achieved on the standard MNIST test dataset are given using Logistic Regression (LR) and Convolutional Neural Network (CNN)

Sample space representation and sample selection

We can consider the complete dataset to be a pool of unlabelled samples from which we wish to decide which samples to select that will best inform the training of a classifier. For the MNIST data, each image is 28 × 28 (784 pixels). Treating each image as a point in a 784-dimensional space, a common technique is to use dimensionality reduction techniques, to then map each image to a 2-dimensional projection whilst aiming to preserve distance and similarity between samples from the high-dimensional space. This aids our ability to visualise and reason about the relationship of samples. ActiVAte incorporates commonly-used methods such as Principal Component Analysis (PCA), t-distributed Stochastic Neighbourhood Embedding (t-SNE) [37] and Uniform Manifold Approximation and Projection (UMAP) [38].

Figure 3 shows the complete dataset within the three different projection spaces: PCA, T-SNE and UMAP. Each plot consists of 55,000 points. The figure shows the general distribution of the complete dataset under each of the three dimensionality reduction techniques, when no label is shown (first column), where the true label is shown by colour (second column), and where an ‘inferred label’ is shown by colour (third column). The inferred label is given by sampling 100 uniform points and providing a label for each, and then assigning labels for the remaining points using a nearest-neighbour approach. This is intended to show how a simple classifier could be developed if the samples can be clustered well initially (using the dimensionality reduction techniques).

It can be seen that PCA does not provide any clear clustering in the general distribution, whereas there are distinct clusters that are visible in both the t-SNE and UMAP representations (albeit at an increased computational cost to perform these methods).

When plotting the true labels, it can be seen that the clusters identified by both t-SNE and UMAP do indeed correspond relatively well with the underlying class labels. Using the projection space to obtain 100 uniform samples, we can then train a logisitic regression (LR) classifier, or even a Convolutional Neural Network (CNN) with this. Using UMAP with and a CNN classifier, we can obtain 88% accuracy against the standard test set. Active learning can often be considered as one of three scenarios: membership query synthesis, stream-based selective sampling, or pool-based sampling [3]. Our approach primarily aligns with pool-based and stream-based techniques, where the machine-driven selection may select from a pool of unlabelled samples, and user-driven selection may select individual samples in some order. We can consider the result presented here as a useful benchmark to compare collaborative selection against, where a more intelligent approach to sampling is adopted through the human-machine working partnership.

User-driven sample selection

At any stage of interaction, the user can browse the projection space by hovering the mouse cursor over each point to see the original image data. They can click on a sample to select it from the pool. If the classifier has not yet been initialised, the sample will be placed in the unlabelled region of the classifier view. If the classifier has been initialised then the sample will be placed in the predicted class region in the classifier view, vertically positioned according to the machine confidence where a higher position indicates a higher level of confidence. In addition to the projection space, the user can also explore the confusion matrix view. The confusion matrix shows the distribution of the most recent training between predicted values and actual values. This is a common technique for being able to identify weaknesses where predicted values do not correspond with actual values (i.e., values that do not sit within the diagonal). The combination of the sample pool, the confusion matrix, and the classifier view can be used to inform users of suitable sample selections that may help to further improve the classifier accuracy.

Machine-driven sample selection

At any stage of interaction, the user can request that the machine provides a set of N samples for the user. There are eight different selection schemes from the sample pool that the machine can utilise: random selection (RS), distance-based selection (DS), least-confidence random selection (LCRS), Least-confidence distance-based selection (LCDS), marginal-confidence random selection (MCRS), marginal-confidence distance-based selection (MCDS), entropy-confidence random selection (ECRS), and entropy-confidence distance-based selection (ECDS).

As the name suggests, random selection (RS) simply selects a sample from the pool at random to query with the user. In distance-based selection (DBS), the machine will iteratively select the point furthest away from all previously-selected points, until N samples have been retrieved, with the aim of optimally selecting points that provide coverage over the entire distribution. Distances can be measured either in the projection space or in the original dimensionality, however in the interest of computation speed we typically use the projection space. In the case of the confidence-based methods (LCRS, LCDS, MCRS, MCDS, ECRS, and ECDS), we make use of various uncertainty sampling techniques used within active learning [3]. The machine selects a number of samples (e.g.,

N^2, either randomly or distance-based) that the system then predicts a label for using the current state of the classifier. The output layer of the classifier gives a probability distribution across each class that can inform on the confidence of each prediction. In the case of least-confidence, the machine selects the N samples that achieve the lowest prediction scores. In the case of marginal-confidence, the machine selects the N samples that have the minimum separation between the predicted class and the second-highest prediction (i.e., cases that may be borderline between the top two predicted classes). In the case of entropy-confidence, the machine selects the N samples that have the highest entropy across the set of predicted class scores (i.e., where there is high randomness within the output layer distribution).

Before a classifier has been initialised, a small subset of samples are selected (either by the user directly, or automatically by the machine) and displayed in the 'unlabelled' region of the classifier view. This small dataset is used to 'seed' the classifier training. The user can then position each sample in the corresponding class region indicated by the coloured segments using drag-and-drop interaction. As each sample is labelled, the corresponding point in the sample pool view is also coloured to match the assigned class. For the user, this is particularly useful for identifying clusters of similar images within the projection space. The user can make use of the vertical positioning of samples to inform the machine of how confident they are in the label—for example, an exemplar of a '5' may be positioned high whereas a poor sample may be positioned lower in the region. This allows the user to inform the system not only the class label, but how much they believe it to be of that particular class.

Training the classifier

At each iteration, the classifier can be trained on all currently-labelled instances. The system allows the classifier to use either a Logistic Regression (LR) model and a Convolutional Neural Network (CNN) model using standard ML libraries, making it quite possible to extend to other forms of classifier. ActiVAte allows four different configurations for how the training data should be presented to the classifier: single-instance labelling (SIL), inferred labelling (IL), image data augmentation (IDA), and confidence-based augmentation (CBA). The user can select whether to run all four training schemes simultaneously, or whether they wish to only run selected schemes.

Single-instance labelling

The simplest case is to train the classifier on only the samples where the user has provided a label, which we refer to as single-instance labelling (SIL). In early stages of training (e.g., with 10 samples), it may be expected that the classifier fails to perform well due to a lack of data. However, as the user provides more labels, the performance would be expected to improve. Coupled with the different selection schemes, it may well be that the classifier can be trained to a sufficient standard with a small subset of high-quality and well-selected samples (e.g., 100), rather than requiring the full set of 60,000 images as used in batch training. This method serves as a baseline for our training, as it represents the direct labelling provided by the user, however with small training samples it is likely that the system will fail to generalise well.

Inferred labelling

To overcome the issue of generalisation above, inferred labelling (IL) adopts a nearest-neighbour approach to obtain labels for the rest of the unlabelled pool based on the knowledge provided by the user. This approach essentially means that the classifier can be trained on all samples available in the pool, however the samples may not necessarily be labelled correctly. Given that the classifier is tested on a consistent test set of 10,000 images, we can mitigate some of the risk in this approach. The classifier may have some performance issues depending on how the nearest-neighbour scheme is computed (e.g., in high-dimensional space, or one of the available projection spaces), however it can help to obtain a quick approximation, for which then the problematic cases can be explored further by the user. It is important to note however that this approach assumes that a large unlabelled sample pool is readily-available, which often would not be the case in online learning tasks. However, in cases where the pool is available, but the cost of human-labelling is high, it can potentially save much user effort.

Sample and confidence-based augmentation

To address the limitations of SIL and IL, a common technique used for training image classifier is to generate augmented copies based on the samples where the user has provided a label. We refer to two schemes here: image data augmentation (IDA), and confidence-based augmentation (CBA). IDA is the typical approach that introduces some subtle transformation (e.g., translation, rotation, scale, and skewness), such that the class remains the same yet the sample is slightly different. Using this approach, the user can label a small sample of images (e.g., 10) and the classifier can be trained on any number of possible combinations of transformation to increase the robustness of the training set. CBA incorporates user confidence as part of the augmentation process, based on the vertical positioning of samples within the visual analytics tool (where a higher sample positioning suggests a higher level of confidence that the sample corresponds to that class). In both IDA and CBA, we duplicate samples to create a new training set. In CBA, each sample is duplicated based on the confidence score assigned. In IDA, each sample is duplicated by a constant. From this, we use the Keras function ImageDataGenerator to generate subtle augmentations of the samples. We duplicate the samples in both IDA and CBA so that the Keras function produces the same number of new instances for subsequent training (so that this does not unintentionally introduce a bias). For the ImageDataGenerator, we use a batch size of 32, giving an augmented total of $32N$ samples, where N is the number of original labelled samples.

Classifier feedback

Following each iteration of training, the system will report the test accuracy scores for each of the selected training schemes as a percentage. A line plot is used to show the percentage of each scheme over time (where time is equivalent to the number of samples given to the classifier for training). As is standard in many machine learning applications, the system is tested against a separate testing dataset to ensure the classifier is generalisable towards new data observations. The user can then also inspect the confusion matrix to examine how the classifier performed in more detail. This shows the occurrence of predictions against their corresponding actual values, for each sample in the test set. This is particularly useful for identifying where mis-classifications have occurred so that sample selection can target uncertainty within the classifier.

After the first iteration of training the classifier, when the user requests a sample from the unlabelled pool, each sample can be positioned in the classifier view based on the current prediction of the machine, using any of the training schemes as selected by the user. The sample is positioned in the horizontally based on the appropriate class label, and vertically based on the confidence associated with that class label. The confidence of the machine prediction can be obtained from the output layer of the classifier model that essentially serves as a probability distribution across all possible classes. This is the case for both the logistic regression model, and the convolutional neural network model, and would extend to many other classifier models also. Samples are positioned with a yellow highlight applied, indicating to the user that this is a machine prediction. The user can then confirm the class decision, or refine the decision by moving the sample to a new class region. If the user confirms the decision, the sample is shown as blue, and if the user refines the decision, the sample is shown as red. This serves as a effective visual cue to the distribution of machine-labelled and human-labelled samples within each class. The highlighting of yellow samples also provides a effective means of 'seeing' the classifier improve over time, as machine-positioned samples gradually become positioned higher up in each class region with each iteration of training. As before, the user can also manually select samples from the sample pool and see how these are positioned within the classifier view, giving a significantly more effective analysis of the classification performance compared to the higher-level overview of the confusion matrix and test accuracy scores. The number of 'corrected' labels provided by the user can also be shown as a bar chart if desired. This indicates the number of cases where the machine label is incorrect and a user has therefore had to relabel (regardless of the user's confidence). This could also be considered as 'user effort', which ideally we would hope to minimise using active machine learning. In our experimentation, we report on user effort for the machine-driven, user-driven, and collaborative selection strategies, in conjunction with the achieved accuracy of the classifier.

Experimentation

We use the *Acti*VAte system to conduct a range of experiments to study the effectiveness of collaborative human-machine active learning, and how visual analytics can aid this process. We test eight schemes of machine-driven sample selection, as described in the previous section (RS, DS, LCRS, LCDS, MCRS, MCDS, ECRS, and ECDS), to study the impact of machine-confidence in active learning. We test against each of the four labelling stategies described previously (SIL, IL, IDA, and CBA). SIL results represent a baseline expectation of the system when only a single observation of each sample is provided. Likewise, IL results represent the assumption of having all samples available for labelling, essentially giving a 'best possible case' when limited sample labels are provided. Primarily, we are interested in how well the two augmentation methods, IDA and CBA, compare with IL given that only a limited set of samples are being trained upon, and to what extent can CBA improve on traditional IDA.

Importantly to note for this study, it is known that Convolutional Neural Networks (CNN) can perform very well on the MNIST digit classification task, obtained in the

Table 1 Summary of classifier accuracy and AUC results after 100 samples, for machine-driven, user-driven, and collaborative selection

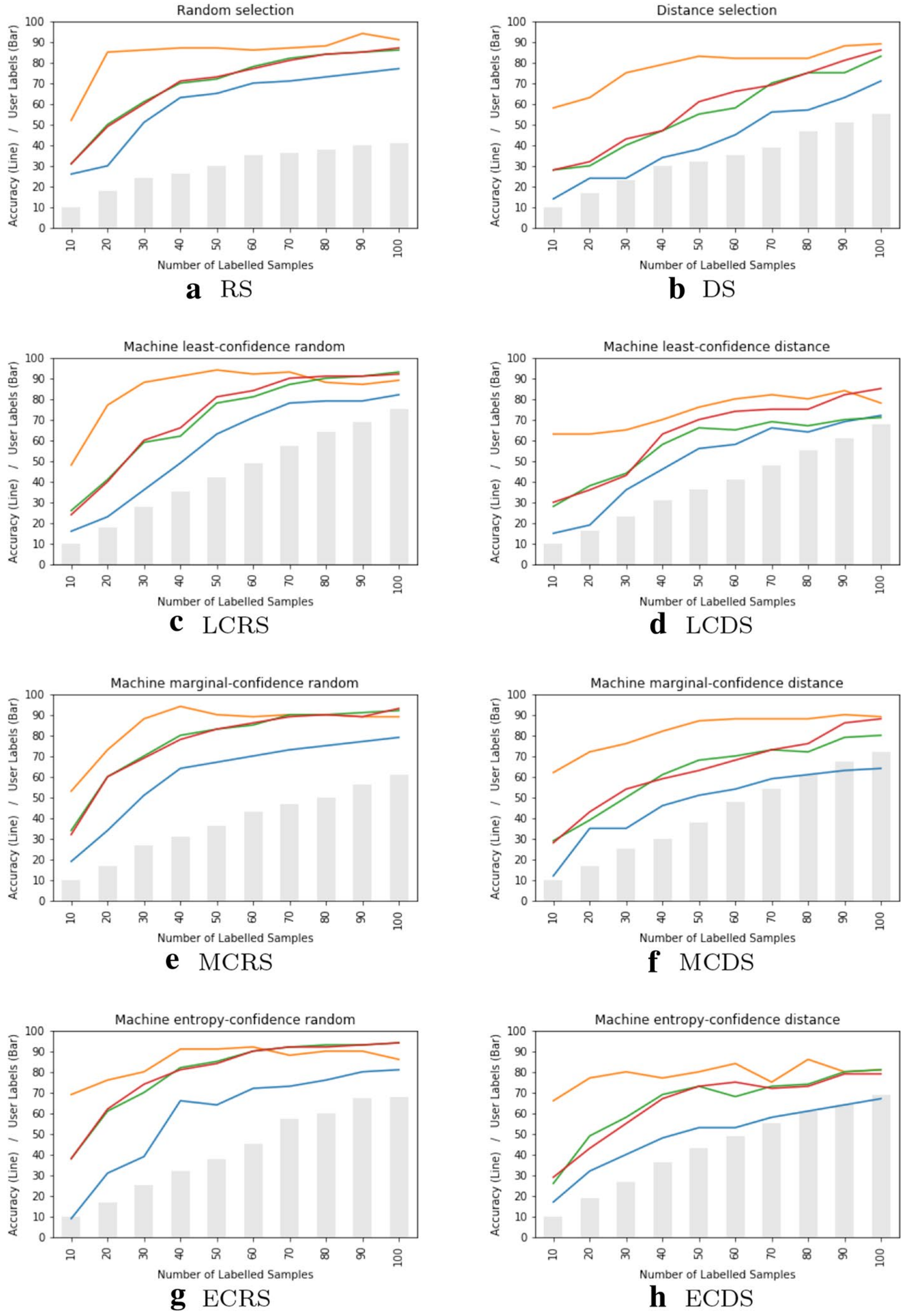
Method	User relabelled	Highest accuracy				Area under curve (AUC)			
		SIL	IL	IDA	CBA	SIL	IL	IDA	CBA
Machine-driven									
RS	41	77	94	86	*87*	54.95	77.15	64.05	63.90
DS	55	71	89	83	*86*	38.35	70.75	50.55	*53.10*
LCRS	75	82	94	93	92	52.70	77.85	64.85	*66.10*
LCDS	68	72	84	71	*85*	45.75	67.05	52.65	*57.55*
MCRS	61	79	94	92	*93*	56.00	77.40	71.20	70.65
MCDS	72	64	90	80	*88*	44.20	74.65	56.65	*58.00*
ECRS	68	81	92	94	94	54.60	77.55	73.20	*73.40*
ECDS	69	67	86	81	79	45.10	71.25	59.75	59.10
User-driven	37	80	96	89	88	60.05	84.45	73.20	*73.30*
Collaborative	48	82	95	90	*94*	54.50	83.05	72.60	*74.80*

The number of user re-labelled samples, and accuracy scores for the four labelling schemes: single-instance labelling (SIL), inferred labelling (IL), image data augmentation (IDA), and confidence-based augmentation (CBA). Results in italic highlight where CBA improves on the IDA approach

region of 97–99% accuracy (the Tensorflow MNIST tutorial shows how to train a model to 99.2% accuracy[1]). However, this is when training a classifier using the complete set of 55,000 labelled images (with 5000 as a validation set and testing on 10,000). Since we are interested in how systems can be trained with limited data samples, we purposely only allow a maximum of 100 labelled training images to be included, since it is clear that manual labelling of thousands of examples is simply not practical for a user, or for any real-world application.

For each study, 10 iterations of training were performed. At each iteration, 10 new samples were appended to the training set, either queried by the system (machine-driven selection), chosen by the user (user-driven selection), or a combination of the two (collaborative selection). Each queried sample was assigned a label by the machine based on the current prediction model. If the label was incorrect, then the user can 'relabel' to correct the machine. The user can also assign their confidence to the label based on vertical position in the class column. For training, each labelling scheme was used to train a separate instance of a CNN model. At each iteration, each model was trained for 5 epoches on the available set of labelled instances, so that computation time was kept at a minimum between user interactions. Each epoch in the inferred case took approximately 1 minute, whilst all other schemes took less than 10 second per epoch. After each iteration, we observe the classification accuracy as assessed against the standard MNIST test dataset. The complete set of accuracy scores also allows us to calculating area-under-curve results (AUC) over the complete training period, which provides a more accurate generalisation of how well the classifier has performed across all iterations, and how rapidly the classifier converges to higher accuracy scores. We also assess the amount of label 'relabelling' required by the user, as a form of correction or intervention required. The complete set of results are summarised in Table 1.

[1] https://www.tensorflow.org/versions/r1.2/get_started/mnist/pros.



a RS

b DS

c LCRS

d LCDS

e MCRS

f MCDS

g ECRS

h ECDS

Fig. 4 Learning curve plots for machine-driven sample selection results. Line: single-Instance (Blue), inferred (Yellow), image-augmentation (Green), confidence-augmentation (Red). Bar: cumulative number of samples relabelled by user

Machine Selection from Sample Pool

In the first experiment, we study the performance of the classifier when the machine is solely responsible for selecting which samples should be labelled by the user. Figure 4 show the learning curve results of the classifier using machine-based sample selection for 100 samples. Table 1 shows that LCRS achieves the highest results of both classifier accuracy and AUC, with 94% and 77.85 respectively. The highest accuracy and AUC for the augmentation methods is achieved by ECRS, with 94% accuracy for both methods, and an AUC of 73.40 and 73.20 for CBA and IDA respectively. This result shows that data augmentation methods can dramatically improve the classifier when training on limited samples, achieving as good an accuracy as when training on all original samples with inferred labels. Given that in most real-world cases, we would not have access to the full training set to perform an inferred approach, this result is extremely encouraging and highlights the benefits of augmentation techniques.

Looking at the augmentation techniques further, we can see that CBA improves on the IDA result in 5 out of 8 cases, and achieves the same accuracy as IDA in 1 case. Similarly, the AUC results for CBA improve on the IDA results in 5 out of 8 cases also, showing that both overall accuracy, and incremental accuracy can be improved using confidence-based augmentation. This improvement may become more noticeable as more training samples are assigned a level of confidence by the user.

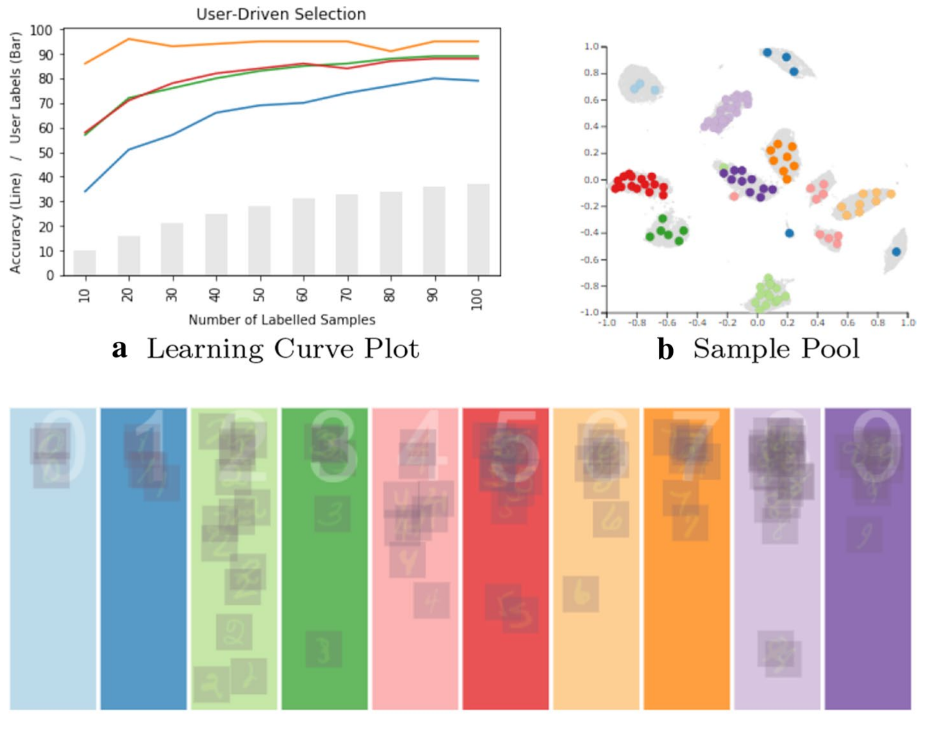
Looking at the difference between the random selection and distance-based selection techniques, it seems that the random selection techniques perform better than the distance-based. The distance-based selection aims to obtain a uniform sample across the complete projection space, however the results suggests that this does not necessarily improve the classifier. From a user perspective, the distance-based approach shows that they are considering the full sample space and so may help to provide a more robust classifier, compared to a random selection where some classes may well become poorly represented. This issue is an important topic in order to trust the reliability of the classifier, where visual analytics can help the inspection the classifier to assess robustness.

The last key observation to make is the amount of samples that are relabelled by the user. The random selection technique required 41 of a possible 100 relabels. The confidence-based measures require significantly more relabels, in the range of 61 to 75. This result is interesting to observe, however it stands to reason. If the machine queries when it is unconfident, the likelihood is that the machine is incorrect, and so naturally, the user will be required to relabel. This relates to the previous point of sample selection, in that this process may make for a more robust classifier when being deployed for use in live production, compared to a example case such as MNIST digit classification.

User selection from sample pool

In this second stage of the study, we train the classifier based on user-selection from the sample pool view. We conducted experiments with four users who were tasked with performing the user selection task for 100 samples to train the classifier. Figure 5 shows the learning curve results for an individual case, along with sample pool view and classifier view.

With the UMAP dimensionality reduction technique, we have already seen that there are distinctive clusters that appear in the sample pool view. A user can leverage the sample pool view to identify such clusters for their selection. By labelling just 10 samples (one from each cluster), the inferred technique scores in the region of 90% accuracy, demonstrating the effectiveness of dimensionality reduction with interactive labelling, however further experimentation would be necessary to see how it performs for other more challenging datasets. After 10 iterations, the IL method achieved results of 96%,



a Learning Curve Plot

b Sample Pool

c Classifier View

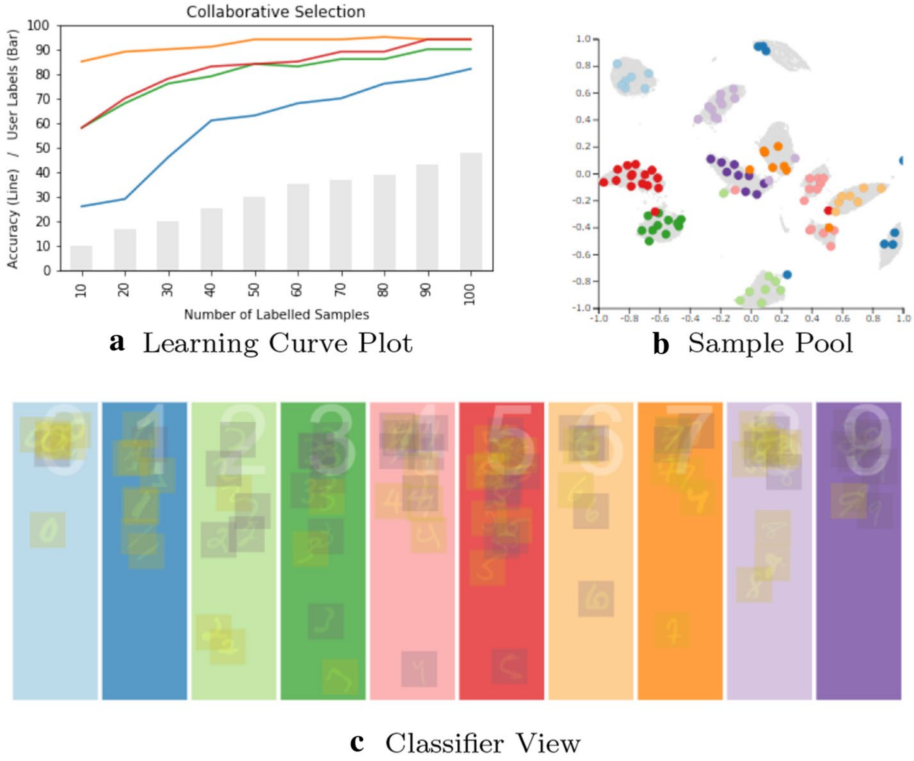
Fig. 5 Results for user-driven sample selection (37 out of 100 user labelled) showing **a** learning curve plot, **b** sample pool view and **c** classifier view

whilst IDA and CBA scored 89% and 88% respectively. In this example, only 37 relabels were made by the user.

One of the remarks from users when performing this task was that whilst the initial selection of samples was fairly intuitive due to the clusters present in the sample pool view, it became increasingly challenging to know which samples should be selected to improve the classifier performance. This remark was in line with our expectation, since with user selection it is for the user to decide which samples would be beneficial to label, which beyond the initial selection can become difficult to know what selections may best improve the classifier. The confusion matrix aims to overcome this to some extent, since users can see where mis-classifications occur, and therefore use this to guide their selection. However, it may be that offering a combination of both machine- and user-selection techniques can help to overcome such bottlenecks more efficiently, placing less cognitive burden and effort on the user.

Collaborative selection from sample pool

In the final experiment, we allow the combination of both machine- and user-selection techniques. The system allows the user to select points from the sample pool view, and also to query the machine for samples using any of the machine-selection techniques. As an example, a user may initially select a point from each cluster, then train, then allow the machine to select the next round of samples based on where the machine confidence



a Learning Curve Plot

b Sample Pool

c Classifier View

Fig. 6 Results for collaborative sample selection (48 out of 100 user labelled) showing **a** learning curve plot, **b** sample pool view and **c** classifier view

is low. This then enables the user to select when they are confident in sample selection (e.g., when clusters are clearly visible), whilst having the machine intervene when they are no so certain about what selections may be beneficial.

Users commented that they found this method most engaging - they would first employ the machine-confidence scheme to propose samples, then observe the Sample View as they moved mislabelled items, in order to guide them to explore (and select samples from) areas where confusion lay. As with the user-selection task, the confusion matrix view could be used which some users found useful for directing their sample selection based on classifier errors and target their efforts more effectively. The combination with the machine selection helped in cases where the user was not clear of what a suitable selection may have been. Users reported that they primarily made use of the machine-confidence distance-based selection, and random selection methods.

Figure 6 shows the learning curve results for the collaborative selection task for an individual, along with sample pool view and classifier view. In this particular example, the following pattern of activity was used for the 10 iterations: user-selection, LCRS, user-selection, MCDS, user-selection, ECRS, user-selection, RS, user-selection, ECRS. From this pattern, it can be seen that a variety of machine-confidence techniques were chosen by the user, with the user intervening at every other iteration.

In this example, the collaborative selection achieves 95% for IL, 94% for CBA, and 90% for IDA, with 48 samples that required relabelling. This result for CBA is equal to the highest scoring machine-driven selection (ECRS), yet here we achieve

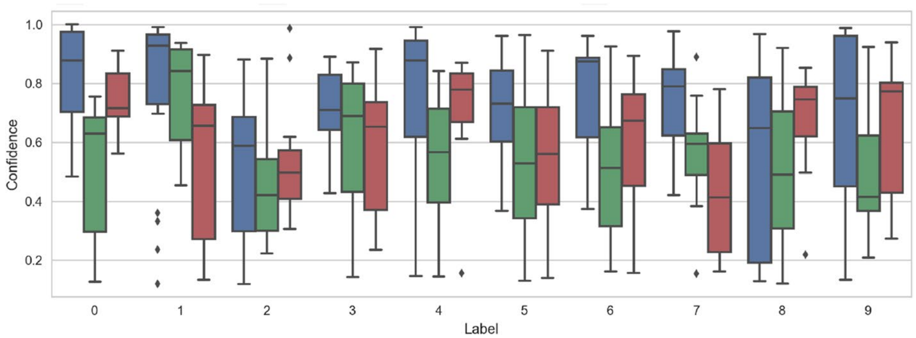


Fig. 7 Confidence ratings for each class based on three methods of machine-driven selection. Legend: DS (column 1—blue), LCDS (column 2—green), LCRS (column 3—red)

this result with 20 less 'relabels' required (48 compared with 68). Therefore, we have minimised the amount of relabelling effort required by the user whilst maintaining high classifier accuracy. Only the random selection method required less relabelling, however it also achieved a lower classifier accuracy which suggests that the collaborative classifier is much more robust. Compared with user-selection, we believe that the improvements come when the user is unsure of which samples may best improve the classifier, beyond the early selection of samples from each class. Understanding the full extent of classifier robustness, beyond classifier accuracy against a test set, is a topic that we intend to pursue in the future.

Analysis of machine-driven selection methods

In addition to our main experimentation, we conduct an analysis of the confidence scores assigned to each class, looking at least-confidence selection and distance-based selection. Figure 7 shows the distribution of confidence scores for each class under three different schemes: distance-based selection (DS), least-confidence distance-based selection (LCDS), and least-confidence random selection (LCRS). From our previous results, we have seen that LCRS scores significantly higher across all four classifier schemes than the other two methods here. From the box plot, one trait that can be seen is how the confidence scores for classes 1 and 7 were much lower with LCRS. These are two classes that can often be mis-classified, and so this may contribute towards the higher accuracy result. It is important to consider that the three methods all consist of different samples being selected. However what this result may suggest is that LCRS was able to identify unconfident samples that are often mis-classified (1's and 7's), and so by having these queried with the user (who may have also been low in confidence about their intended class), meant that the robustness of the classifier could be improved, resulting in the increase of accuracy. To validate this further we conducted this same study for three independent users. What we found was that whilst different schemes give different results, the different users actually performed the same, with LCRS and LCDS revealing samples that are labelled by users as lower in confidence, whilst achieving higher classification accuracy in these schemes.

Discussion

Following our experimentation, we discuss our findings along with the further research challenges that have come as a result of this study.

The inferred labelling scheme gave a result of 94% in the case of least-confidence random selection, 96% in the case of user-selection, and 95% in the case of collaborative selection. This method was used as a measure of 'best possible case', where training can be performed on a large sample pool by inferring labels, rather than providing labels directly to all samples. Whilst this can be effective in some cases, the primary drawback of this approach is clearly having access to a large sample pool to begin with (e.g., in the case of MNIST this is 55,000 samples). In the context of our study, we use IL to compare the performance of the other three methods that only consider samples that are labelled directly. Of the three methods (SIL, IDA, and CBA), the greatest performance was achieved using the confidence-based augmentation scheme, scoring 94% accuracy with both the entropy-confidence random selection (ECRS) and also the collaborative selection. Given the significant reduction in the number of original samples required between CBA and IL, this demonstrates the benefits of data augmentation when working with limited data samples. Investigating this further, the collaborative approach required 20 less user 'relabels' than the ECRS approach. Also, the AUC result for the collaborative approach was 74.80, compared to 73.40 for the ECRS. The AUC result would suggest that the collaborative approach converged and maintained a higher accuracy score earlier in the training process than when using the ECRS method. Compared against machine-driven techniques, users reported a greater sense of engagement with the collaborative approach that enabled them to better inspect the process of sample selection and training of the classifier, to understand the classifier performance further. Our findings suggest then that the collaborative approach achieves high accuracy, low 're-labelling', and faster convergence, compared to machine-driven selection.

In the case of machine-driven selection, the use of a single technique (distance-based, confidence-based, random) will focus effort on only one metric in terms of how samples are selected. For example, the distance-based approach will obtain excellent coverage across the sample pool space, however will fail to extract subtle difference between closely-positioned sample that may exhibit different labels (e.g., a '3' may be positioned near to an '8' in the sample pool due to the similarity in shape). Figure 8 shows an example of this. Likewise, the confidence-based selection methods are powerful for understanding the current weaknesses of the classifier, however by focussing *only* on the weak samples, the classifier may fail to obtain sufficient information across all classes within the 100 samples tested. Therefore it is highly advantageous to consider multiple machine-based selection schemes, and couple this with the collaborative workflow approach.

Much of the current work on active deep learning still assumes a pool of approximately 1000 labelled instances [39]. Compared to hand-labelling these samples, our approach allows much faster preparation of a training set by utilising a classifier based on a small sample set, and so could easily work in conjunction with other methods. We purposefully restrict our experimentation to labelling only 100 samples, treating this as the upper limit of what a user would deem as acceptable labour requirements for this task before fatigue or complacency impact the quality of the user's responses. To overcome

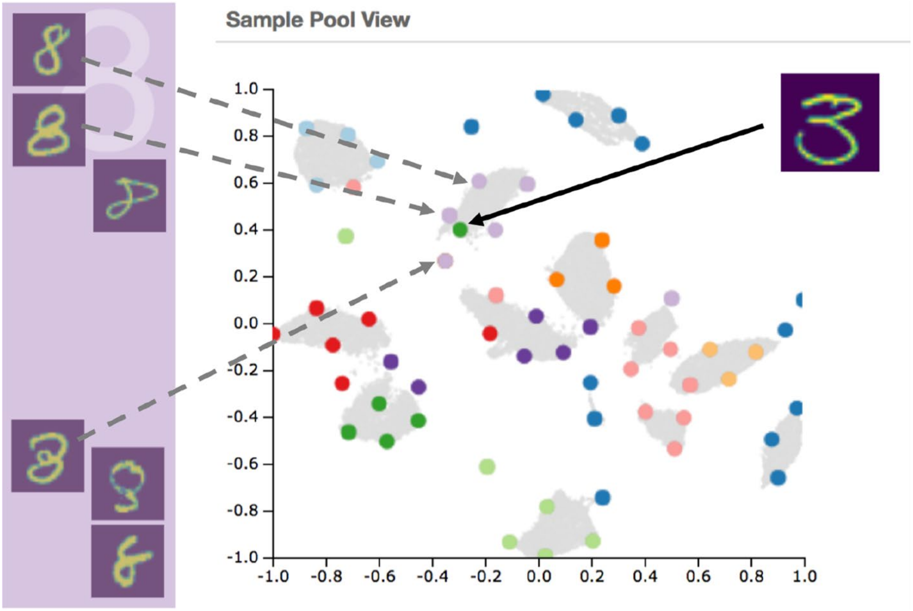


Fig. 8 Example to show limitations of distance-based selection scheme. 60 samples have been selected by the machine and a cluster of 8's can be observed. Yet, there is a 3 directly by a selected sample in this cluster (manually selected by user). This mis-classification would not be detected by distance-selection until the distance parameter is sufficiently low for a neighbouring point to be selected. A collaborative approach that incorporates other machine-selection schemes such as machine-confidence random selection (MCRS) scheme would help to overcome these limitations

the issue of limited training images, the two augmentation methods provide generated samples from the user labelled cases, whilst not requiring additional samples outside of those that a genuine label has been given to. These show a significant improvement in performance over the single-instance approach (beating it in every iteration of every test run), and in many cases, rival closely the inferred approach that serves as a 'best possible case'. In terms of how confidence was incorporated with the augmentation stage of training, our results suggest some minor improvements here. The limit to this improvement may be due to the limited amount of "augmentation" we allowed—just x and y shifts-possibly the use of 'shears' or rotations would have yielded improved performance. However, our purpose is to explore a general approach, rather than fine-tune parameters of the machine learning algorithms and training. The other aspects of how confidence is incorporated into the system (e.g., sample selection based on machine confidence, and machine reporting of confidence) both proved effective for encouraging user interaction and engagement, and so we believe that there remains much interesting research to pursue on how confidence relates to collaborative interactive systems.

Currently, the confidence-based augmentation method weights samples depending on the positioning of the sample by the user. We opt for a higher positioning suggesting that it is a good quality exemplar for that class, and therefore should be emphasised within the training process. This results in duplicates of good quality samples to weight the system towards these. This also essentially reduces the impact that bad quality examples can have, without having to completely disregard them from the training process. This is particular important in cases where a user is unconfident about the label (e.g., in the case of a poorly written MNIST digit). In our testing, many digits can easily be mis-classified (e.g., 5 and 6's are similar, so are 4's and 9's, and some are very difficult to distinguish). If a user provides a label incorrectly, then this will contaminate the training process. However, if they can provide the label and state that they are uncertain, this would have a much lesser impact on the overall classifier performance. One possible alternative to weighting by user confidence would to weight samples based on user perceived difficulty, where an image that is deemed 'difficult' to clearly recognise is placed higher. Difficult cases are likely to be towards the boundaries of multiple classes, such as the boundary between '4' and '9', or between '3' and '8'. If the system was weighted towards the edge cases, could this improve the classifier performance? Our initial experimentation suggests that users deliberately select examples across the boundaries, so there may be other forms of how we account for this information. We believe that there is further work to be done on exploring how confidence-based augmentation of samples may be performed to influence the training set towards the exemplar cases or difficult cases.

Traditionally, machine learning has often been considered to be a 'black box', and so part of this research has been to explore better forms of interactions with machine learning to facilitate understanding. The system allows users to understand the classifier performance at intervals between training iterations via a number of interactive forms, including the reported test accuracy, the confusion matrix that gives a overview of prediction and actual label correspondence, and sample selection to observe the prediction and confidence assigned by the current classifier. The combination of a confidence-based visual analytic loop, with an active learning framework, means that we can encourage a more dynamic process for collaboration in the task of labelling and training a machine learning classifier, providing a more transparent view of how the machine performs, where the current classifier limitations are, and how to assist the user for further training iterations.

Conclusion

In this work, we have studied how active machine learning can be further enhanced through visual analytics to provide greater emphasis towards human-machine collaboration. This is particularly advantageous where limited data samples are available or where labelling costs are high. Our visual analytic tool, *Acti*VAte, integrates a confidence-based visual analytic loop to facilitate human-machine collaboration, inspection, transparency, and trust, in the training of machine learning algorithms. We also explore a variety of sample selection techniques driven by different actors. From our experimentation, we find that collaborative selection provides high learning accuracy (both in terms of maximum accuracy achieved, and in terms of how quickly an accurate classifier can be achieved) whilst also minimising the labour requirements of user correction and relabelling. We also show how confidence can be integrated within the training process, both in terms of how confidence the machine is about a prediction, how confident a user is regarding a label, and how samples can be selected based on confidence. We know that both machines and users are not infallible, and so there is a need to encourage greater interactivity with machine learning systems to allow users to understand how they are trained, and whether there are trained 'correctly' based on what the user provides as input, and what the user may expect as an output. Our system can learn from the user, and the user can learn from the system, whilst recognising that neither is an all-knowing oracle but that together they can facilitate the other to promote knowledge generation and mutual transparency in the process.

Future work will explore this capability of collaborative human-machine learning further. There are a variety of different learning models that could benefit from this approach, such as recurrent neural networks that consider the sequential nature of observations. We intend to pursue how such learning mechanisms can be supported by visual interfaces to further human-machine understanding. As machine learning continues to be adopted in many different applications in society, understanding and interacting with such models is crucial to ensure that there is accuracy and trust on both sides of the collaboration.

Authors' contributions

PL led the research, contributing towards all aspects of the design and development of the *ActiV*Ate system. JS contributed towards the design and development of the *ActiV*Ate system, addressing the introduction of confidence assessment, transparency, and the experimental setup. AD worked on the project and the early implementations of the *ActiV*Ate system. All authors contributed towards the preparation of this manuscript. All authors read and approved the final manuscript.

Acknowledgements

We would like to thank Richard Preen who assisted with some of the early experimentation for this study.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

The MNIST dataset used in this work is publicly available to download: http://yann.lecun.com/exdb/mnist/. The *ActiV*Ate software tool is available for research use. Please contact the corresponding author to request this.

Funding

This work was supported in part by the Defence Science and Technology Laboratory (DSTL), through the HASTE (Human-centric Active-learning for decision Support in Threat Exploration) project.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 2 August 2018 Accepted: 28 January 2019
Published online: 14 February 2019

References

1. LeCun Y, Cortes C (2010) MNIST handwritten digit database
2. Smith JE, Tahir MA, Sannen D, Brussel Hv (2012) Making early prediction of the accuracy of machine learning applications. In: Lughofer E, Sayed-Mouchaweh M (eds) Learning in non-stationary environments: methods and applications. Springer, New York, NY, USA, pp 121–151
3. Settles B (2012) Active learning. Synthesis lectures on artificial intelligence and machine learning. Morgan & Claypool Publishers, San Rafael. https://doi.org/10.2200/S00429ED1V01Y201207AIM018
4. Attenberg J, Provost F (2011) Inactive learning?: Difficulties employing active learning in practice. ACM SIGKDD Explorat Newslett 12(2):36–41. https://doi.org/10.1145/1964897.1964906
5. Raghavan H, Madani O, Jones R (2006) Active learning with feedback on features and instances. J Machine Learn Res 7:1655–1686
6. Donmez P, Carbonell JG (2008) Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In: Proceedings of the 17th ACM conference on information and knowledge management. CIKM '08, pp. 619–628. ACM, New York. https://doi.org/10.1145/1458082.1458165. http://doi.acm.org/10.1145/1458082.1458165
7. Lin CH, Mausam, Weld DS (2016) Re-active learning: active learning with relabeling. In: Schuurmans D, Wellman M (eds.) Proc. Thirtieth AAAI conference on artificial intelligence. AAAI'16, pp. 1845–1852. AAAI Press, Palo Alto
8. Lughofer E, Smith JE, Tahir MA, Caleb-Solly P, Eitzinger C, Sannen D, Nuttin M (2009) Human-machine interaction issues in quality control based on on-line image classification. IEEE Trans Syst Man Cybern A Syst Hum 39(5):960–971
9. Smith J, Legg P, Kinsey K, Matovic M Predicting user's confidence during visual decision making. ACM transactions on interactive intelligent systems (in press)
10. Cook A, Wu P, Mengersen K (2015) Machine learning and visual analytics for consulting business decision support. In: Engelke U, Heinrich J, Bednarz T, Klein K, Nguyen QV (eds) Big data visual analytics (BDVA). IEEE Press, Piscataway, pp 1–2. https://doi.org/10.1109/BDVA.2015.7314299

11. Choo J, Lee H, Liu Z, Stasko J, Park H (2013) An interactive visual testbed system for dimension reduction and clustering of large-scale high-dimensional data. In: Wong DL P C Kao, Hao MC, Chen C (eds.) Proc. SPIE 8654, visualization and data analysis (VDA), pp. 1–15. SPIE, Bellingham. https://doi.org/10.1117/12.2007316
12. Krause J, Perer A, Ng K (2016) Interacting with predictions: Visual inspection of black-box machine learning models. In: Proc. CHI conference on human factors in computing systems. CHI '16, pp. 5686–5697. ACM, New York. https://doi.org/10.1145/2858036.2858529
13. Legg PA, Chung DHS, Parry ML, Bown R, Jones MW, Griffiths IW, Chen M (2013) Transformation of an uncertain video search pipeline to a sketch-based visual analytics loop. IEEE Trans Vis Comput Graph 19(12):2109–2118. https://doi.org/10.1109/TVCG.2013.207
14. Legg PA, Buckley O, Goldsmith M, Creese S (2015) Caught in the act of an insider attack: detection and assessment of insider threat. In: Choi M, Flavin J (eds) IEEE international symposium on technologies for Homeland Security (HST). IEEE Press, Piscataway, pp 1–6. https://doi.org/10.1109/THS.2015.7446229
15. Legg PA (2015) Visualizing the insider threat: challenges and tools for identifying malicious user activity. In: Harrison L (ed) IEEE symposium on visualization for cyber security (VizSec). IEEE Press, Piscataway, pp 1–7. https://doi.org/10.1109/VIZSEC.2015.7312772
16. Endert A, Ribarsky W, Turkay C, Wong BLW, Nabney I, Blanco ID, Rossi F (2017) The state of the art in integrating machine learning into visual analytics. Computer Graphics Forum. https://doi.org/10.1111/cgf.13092
17. Shixia L, Xiting W, Mengchen L, Jun Z (2017) Towards better analysis of machine learning models: a visual analytics perspective. Visual Inform 1(1):48–56. https://doi.org/10.1016/j.visinf.2017.01.006
18. Ren D, Amershi S, Lee B, Suh J, Williams JD (2017) Squares: supporting interactive performance analysis for multiclass classifiers. IEEE Trans Vis Comput Graph 23(1):61–70
19. Phillips RL, Chang KH, Friedler SA (2017) Interpretable active learning. In: Kim B, Malioutov DM, Varshney KR, Weller A (eds.) Proc. 2017 ICML Workshop on Human Interpretability in Machine Learning (WHI). arXiv:1708.00049
20. Rubens N, Sugiyama M (2007) Influence-based collaborative active learning. In: Konstan JA (ed.) Proc. 2007 ACM Conference on Recommender Systems. RecSys '07, pp. 145–148. ACM, New York. https://doi.org/10.1145/1297231.1297257
21. Yang Y, Loog M (2016) Active learning using uncertainty information. In: Bayro-Corrochano E (ed.) 23rd international conference on pattern recognition, pp. 2646–2651. IEEE Press, Piscataway. https://doi.org/10.1109/ICPR.2016.7900034
22. Liu M, Shi J, Cao K, Zhu J, Liu S (2018) Analyzing the training processes of deep generative models. IEEE Trans Vis Comput Graph 24(1):77–87. https://doi.org/10.1109/TVCG.2017.2744938
23. Yao M, Shaozu C, Ruixiang Z, Zhen L, Yuanzhe C, Yangqiu S, Huamin Q (2017) Understanding hidden memories of recurrent neural networks. In: IEEE conference on visual analytics science and technology. IEEE Computer Society, Los Alamitos
24. Kahng M, Andrews PY, Kalro A, Chau DH (2018) ActiVis: visual exploration of industry-scale deep neural network models. IEEE Trans Vis Comput Graph 24(1):88–97. https://doi.org/10.1109/TVCG.2017.2744718
25. Olah C, Satyanarayan A, Johnson I, Carter S, Schubert L, Ye K, Mordvintsev A (2018) The building blocks of interpretability. Distill. https://doi.org/10.23915/distill.00010. https://distill.pub/2018/building-blocks
26. Bilal A, Jourabloo A, Ye M, Liu X, Ren L (2018) Do convolutional neural networks learn class hierarchy? IEEE Trans Vis Comput Graph 24(1):152–162. https://doi.org/10.1109/TVCG.2017.2744683
27. Thomaz AL, Breazeal C (2008) Teachable robots: understanding human teaching behavior to build more effective robot learners. Artif Intell 172(6):716–737. https://doi.org/10.1016/j.artint.2007.09.009
28. Takagi H (2001) Interactive evolutionary computation: fusion of the capabilities of ec optimization and human evaluation. Proc IEEE 89(9):1275–1296. https://doi.org/10.1109/5.949485
29. Caleb-Solly P, Smith JE (2007) Adaptive surface inspection via interactive evolution. mage Vis Comput 25(7):1058–1072
30. Pauplin O, Caleb-Solly P, Smith JE (2010) User-centric image segmentation using an interactive parameter adaptation tool. Pattern Recognit 43(2):519–529. https://doi.org/10.1016/j.patcog.2009.03.007
31. Llorà X, Sastry K, Goldberg DE, Gupta A, Lakshmi L (2005) Combating user fatigue in igas: Partial ordering, support vector machines, and synthetic fitness. In: Proceedings of the 7th annual conference on genetic and evolutionary computation. GECCO '05, pp. 1363–1370. ACM, New York. https://doi.org/10.1145/1068009.1068228. http://doi.acm.org/10.1145/1068009.1068228
32. Amershi S, Cakmak M, Knox WB, Kulesza T (2014) Power to the people: the role of humans in interactive machine learning. AI Magazine
33. Sacha D, Zhang L, Sedlmair M, Lee JA, Peltonen J, Weiskopf D, North SC, Keim DA (2017) Visual interaction with dimensionality reduction: a structured literature analysis. IEEE Trans Vis Comput Graph 23(1):241–250. https://doi.org/10.1109/TVCG.2016.2598495
34. Aung AM, Whitehill J (2018) Harnessing label uncertainty to improve modeling: an application to student engagement recognition. In: 2018 13th IEEE international conference on automatic face gesture recognition (FG 2018), pp. 166–170. https://doi.org/10.1109/FG.2018.00033
35. Smith J, Legg P, Matovic M, Kinsey K (2018) Predicting user confidence during visual decision making. ACM Trans Interact Intell Syst 8(2):10–11030. https://doi.org/10.1145/3185524
36. Bernard J, Zeppelzauer M, Sedlmair M, Aigner W (2018) Vial: a unified process for visual interactive labeling. The Visual Computer. https://doi.org/10.1007/s00371-018-1500-3
37. van der Maaten LJP, Hinton GE (2008) Visualizing high-dimensional data using t-sne. J Machine Learn Res 9:2579–2605
38. McInnes L, Healy J (2018) Umap: uniform manifold approximation and projection for dimension reduction. arXiv:1802.03426
39. Katharopoulos A, Fleuret F (2018) Not all samples are created equal: Deep learning with importance sampling. CoRR **abs/1803.00942**. arXiv:1803.00942