Responsibility & Machine Learning: Part of a process

Jatinder Singh*†, Ian Walden+, Jon Crowcroft*, Jean Bacon*

*Computer Laboratory, University of Cambridge
+Centre for Commercial Law Studies, Queen Mary University of London

Abstract

As machine learning (ML) becomes increasingly prevalent, concerns are mounting over its use. This discussion paper explores notions of responsibility with regard to ML, focusing on transparency and control. We recognise that such concerns extend beyond the ML technology itself, to the workflows and processes in which the ML operates, i.e. its potential impact. As such, it is important to consider not only the nature of machine learning techniques, but also the data involved and its fit within a broader process. Each of these aspects relate to responsibility, as they represent points for choice and intervention.

1. Setting the scene

Machine learning (ML) is currently the subject of much hype.[1] There has been significant attention given to recent high profile achievements, such as Google/Deepmind's *AlphaGo* that recently defeated a grandmaster in the game *Go*,[2] and less-savoury outcomes, such as Microsoft's Twitter-bot *Tay* that became foul-mouthed and racist after being exposed to Internet trolls.[3]

Much of the surge in enthusiasm regarding ML stems from its promise of enabling new insights and efficiencies in a wide range of areas,[4] including medicine (diagnostics, precision medicine), finance (fraud detection, trend prediction), transport (automated cars, traffic management), cyber security (intrusion detection) to name but a few. ML is already actively used in a number of areas, for example for voice and handwriting recognition, spam detection, automated translation, and e-commerce recommendation systems.

Machine learning[5] works to uncover patterns in data, to build and refine representative mathematical models of data that can be used to make predictions and/or describe data to gain

† Contact author: jatinder.singh@cl.cam.ac.uk

This paper has been produced by members of the Microsoft Cloud Computing Research Centre (MCCRC), a Microsoft funded collaboration between the Cloud Legal Project, Centre for Commercial Law Studies, Queen Mary University of London and the Computer Laboratory, University of Cambridge. A draft was first presented in Sept 2016 at the MCCRC Symposium on "Machine Learning: Technology, Law & Policy". We thank Toby Miller, the rest of the MCCRC team, and attendees of the symposium for their valuable comments and feedback. We also thank Microsoft for their financial support. Responsibility for views expressed remains with the authors.

[1] 'Gartner 2015 Hype Cycle: Big Data Is Out, Machine Learning Is in', accessed 25 July 2016, http://www.kdnuggets.com/2015/08/gartner-2015-hype-cycle-big-data-is-out-machine-learning-is-in.html.

[2] 'AlphaGo | Google DeepMind', accessed 25 July 2016, http://deepmind.com/alpha-go.

[3] Helena Horton, 'Microsoft deletes "teen girl" AI after it became a Hitler-loving sex robot within 24 hours', *The Telegraph*, 24 March 2016. http://tinyurl.com/hjsdsjh

[4] For some examples, see Gartner, 'Machine Learning Drives Digital Business', 11 August 2014, http://www.gartner.com/document/2820120.

[5] Note that this paper considers ML in a more immediate future. We do not explore issues as they might relate to superintelligence, artificial general intelligence, the singularity, or artificial consciousness. Rather, we focus on what is termed narrow or weak artificial intelligence; i.e. ML/AI techniques applied to achieve specific goals: see Stuart Jonathan Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach* (Prentice Hall, 2010).

Electronic copy available at: https://ssrn.com/abstract=2860048

knowledge and insight.[6] As a discipline, ML is not particularly new; however, several factors have led to increased awareness.

First is the emergence of *'big data'* and the *Internet of Things* (IoT), as a growing source of such data, which in light of the so-called '*data revolution'*[7] is motivating the search for new data-driven insights and efficiencies. There is a need for new and better mechanisms for dealing with the volume of data available: to assist in finding the signals within the noise, the needles within the ever-growing haystacks; as well as for increasing scope for automation. ML has much to offer in this space.

Technical developments are also relevant. Lower storage costs enable ever more data to be collected. Improvements in network infrastructures, for transferring and distributing data, and in computational power, e.g. GPU technologies that suit certain ML methods,[8] bring more capacity for supporting and developing ML processes. Cloud computing, by providing access to (potentially) vast resources on-demand, facilitates further research, development, experimentation and deployment of more complex ML techniques. Access to state-of-the-art ML functionality is improving: *Machine Learning as a Service (MLaaS)* offerings continue to emerge,[9] and a number of ML platforms and libraries are openly available.[10]

Further, there have been advancements in the machine learning techniques themselves. Particularly high profile is the emergence of *deep learning*,[11] which in artificial neural networks for example, involves multiple (hidden) layers that enable more complex models to be constructed (see §2.1). AlphaGo is a well-known example that involves deep learning techniques.[12]

1.1 Responsibility in ML systems

ML raises interesting questions regarding responsibility. Generally the concern is that if ML operates without being specifically programmed – that is, by learning a *model* from data – where does, or should, responsibility lie for the consequences of decisions and actions that result from its outputs?

In law, responsibility generally leads to liability, i.e. a potentially adverse consequence for the person held responsible, given that harm may be caused to persons, property or various states of being.[13] One function of a legal system is to assign liability (*read* responsibility) in specified situations; to establish some legal certainty where complexity exists. Under a product liability regime, for example, the 'producer' is designated as strictly liable for any harm caused by the product.[14]

---

[6] Ethem Alpaydin, *Introduction to Machine Learning* (MIT Press, 2014).

[7] Rob Kitchin, *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences* (SAGE, 2014).

[8] 'How the GPU Is Revolutionizing Machine Learning | NVIDIA Blog', accessed 26 July 2016, https://blogs.nvidia.com/blog/2015/12/10/machine-learning-gpu-facebook/.

[9] Some examples include BigML.com, AmazonML, Google Cloud Machine Learning, Microsoft Azure Machine Learning and IBM BlueMix.

[10] For some examples see: https://daoudclarke.github.io/machine%20learning%20in%20practice/2013/10/08/machine-learning-libraries/

[11] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, 'Deep Learning', *Nature* 521, no. 7553 (28 May 2015): 436–44, doi:10.1038/nature14539.

[12] David Silver et al., 'Mastering the Game of Go with Deep Neural Networks and Tree Search', *Nature* 529, no. 7587 (28 January 2016): 484–89.

[13] E.g. public order or the administration of justice.

[14] See Noto La Diego and Walden, *Contracting for the Internet of Things: Looking into the Nest,* forthcoming in the European Journal of Law and Technology.

Electronic copy available at: https://ssrn.com/abstract=2860048

Regarding ML, the key consideration is the *impact* and *effect* of the ML. We emphasise the importance of recognising that *ML operates as part of a system, entailing different processes and comprised of different components*.

This discussion paper does not directly examine issues of legal liability,[15] but instead considers the component elements of ML systems to identify those points where a person (whether natural or legal) can be said to exercise control and therefore may be held to be responsible. Our goal is to illuminate aspects of responsibility as they relate to the design, development, use and ongoing management of ML systems.

Specifically, we explore *control* and *transparency* relating to the construction and use of ML systems, considering the nature of:

- ML techniques,
- data (training and operational),
- ML outputs, and
- the broader systems context; i.e. the workflows, processes, and system supply chains surrounding and integrating the ML.

Each of the above relate to responsibility, as they entail design decisions and represent points for intervention.

By '*control*', we mean both a legal right and the ability to make decisions about how an ML system is made, deployed, managed and utilised. One without the other can also give rise to responsibility, and therefore liability, but are beyond the scope of this paper.[16] The concept of 'control' has gained widespread usage in UK law for addressing the regulation of information and communication technologies ('ICT'), such as ML systems.[17]

In terms of '*transparency*', those persons exercising control (including building the system) will obviously have a need in order to be able make effective decisions in respect of the ML system, such as evaluating and tuning the system. However, there is also the need for demonstrable transparency towards third parties, such as end users and regulators.[18] Mandatory transparency is also a common regulatory response to ICT developments, designed to facilitate choice and accountability.[19] The provision of 'raw information' is generally not sufficient to discharge an obligation of transparency; such information must also be 'intelligible' to the person to whom it is given.[20]

We conclude by indicating some technical research directions, which may assist with managing obligations and clarifying responsibilities.

---

[15] See Chris Reed, Elizabeth Kennedy, and Sara Nogueira Silva, 'Responsibility, Autonomy and Accountability: Legal Liability for Machine Learning', Queen Mary School of Law Legal Studies Research Paper No. 243/2016. Available at SSRN: https://ssrn.com/abstract=2853462, Oct 2016.

[16] E.g. unauthorised access to a system.

[17] E.g. the Computer Misuse Act 1990, s. 17(5), "entitled to control access of the kind in question to the program or data"; the Regulation of Investigatory Powers Act 2000, s. 1(3), 'a person having the right to control the operation or the use of a private telecommunication system'; Communications Act 2003, s. 32(4)(b), "under the direction or control of another person".

[18] See Dimitra Kamarinou and Christopher Millard, 'Machine Learning with Personal Data', SSRN, 2016 (to appear).

[19] Privacy and Electronic Communications Regulations 2003 (SI No. 2426), r. 6; Electronic Commerce Regulations 2002 (SI No. 2013), at r. 9-10.

[20] E.g. Data Protection Act 1998, s. 7(1)(c).

2. Machine learning algorithms

ML entails learning patterns and relationships from data to build *generalisable models* that, when exposed to new or unseen data, assist in a range of general tasks, including categorisation, profiling, prioritisation, filtering and prediction.[21] This section focuses on the ML techniques – *the algorithms* – for learning a model from the data. Note that we take a narrow view of the term 'algorithm', specifically considering the techniques for model building, as opposed to some literature that uses the term in a broader sense to encompass the surrounding systems and processes.

There is a wide variety of algorithmic ML techniques. In an algorithmic context, *transparency* relates to the degree the internals of the algorithms can be seen and understood; for instance, exposing the features of the data the learned model takes into account, the associations and rules that were derived, and the extent to which the model relies on these. Notions of *control* are very much related, concerning how algorithms can be managed and constrained, for example to prevent particular associations being made, or setting bounds on outputs.

Selecting an appropriate approach that is relevant to the problem domain is a challenge in itself.[22] However, it follows that algorithmic selection not only impacts the quality of the ML model, but the degree to which the inner workings of the ML algorithm and learned model can be interpreted and controlled *depends on the technique used*.

2.1 Algorithmic opacity

Some ML algorithms are more amenable to meaningful inspection[23] (see Table 1) and management than others.

	Decision Trees	*Neural Networks*	*Naïve Bayes*	*kNN*	*Support Vector Machines*	*Rule learners*
Explainability / Knowledge transparency	****	*	****	**	*	****

Table 1: Comparison of learning algorithms for classification and their levels of interpretability ('**' being most interpretable). Table adapted from Kotsiantis.[24]**

Generally, logic-based ML approaches tend to be easier to interpret,[25] for instance, those based on learning decision trees.[26] As Fig 1. illustrates, a decision tree essentially entails a graph

[21] Nicholas Diakopoulos, 'Accountability in Algorithmic Decision Making', *Commun. ACM* 59, no. 2 (January 2016): 56–62, doi:10.1145/2844110.

[22] For an illustration of appropriate techniques for various problem domains as offered by the Azure platform: http://download.microsoft.com/download/A/6/1/A613E11E-8F9C-424A-B99D-65344785C288/microsoft-machine-learning-algorithm-cheat-sheet.pdf

[23] 'Meaningful' in the sense that it can be readily interpreted and understood by humans: see intepretability in Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, '"Why Should I Trust You?": Explaining the Predictions of Any Classifier', *arXiv:1602.04938 [Cs, Stat]*, 16 February 2016, http://arxiv.org/abs/1602.04938.

[24] S. B. Kotsiantis, 'Supervised Machine Learning: A Review of Classification Techniques', in *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies* (Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007), 3–24.

[25] Ibid.

[26] J. R. Quinlan, 'Induction of Decision Trees', *Machine Learning* 1, no. 1 (n.d.): 81–106, doi:10.1007/BF00116251.

(flowchart) representing the associations between variables in the data. Such a representation is more amenable to interpretation (when compared to other methods, though of course, the degree of interpretability depends on the complexity of the tree – highly complex trees can be very difficult to interpret);[27] e.g. in a situation where a recommender system is used to assist employment decisions, to check whether inappropriate variables or associations such as race or gender improperly feature within the tree. Regarding the more transparent and intuitive probabilistic methods,[28] such as Bayes approaches, there are means for the *latent* (the inferred, hidden) variables that form part of the model to be uncovered,[29] which can help illuminate whether discriminatory associations are being made.

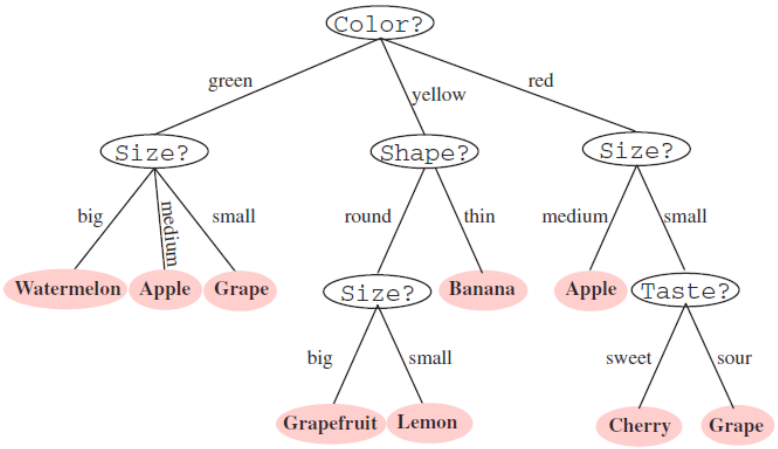


Figure 1: An example decision tree for classifying fruit[30]

In terms of control, note that it is often not enough simply to remove variables of possible concern (e.g. race in a discrimination context) from the data source (or model), as not only does this reduce the information available for learning, but these variables might be highly correlated with other attributes of the dataset (e.g. postcodes that represent particular demographics),[31] and/or the issues may be the result of combining variables (e.g. age, gender *and* postcode).[32] Work is being done on constraining the learning process to prevent such improper features and associations from being made.[33]

When discussing ML as a 'black-box', *artificial neural networks*[34] are often the 'go to' example. A common form of neural network consists of three layers each with a number of nodes: an input

---

[27] There is work on trying to constrain the size of decision trees to maintain interpretability, e.g. Minos Garofalakis et al., 'Building Decision Trees with Constraints', *Data Mining and Knowledge Discovery* 7, no. 2 (n.d.): 187–214.

[28] Datta, Anupam, Shayak Sen, and Yair Zick, 'Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems', *IEEE Symposium on Security and Privacy*, May 2016; Kotsiantis, 'Supervised Machine Learning'.

[29] Nevin L Zhang, Thomas D Nielsen, and Finn V Jensen, 'Latent Variable Discovery in Classification Models', *Artificial Intelligence in Medicine*, Bayesian Networks in Biomedicince and Health-Care, 30, no. 3 (March 2004): 283–99, doi:10.1016/j.artmed.2003.11.004.

[30] Image from Michigan State University, *CSE 802 Pattern Recognition and Analysis Lecture Notes*: http://www.cse.msu.edu/~cse802/DecisionTrees.pdf

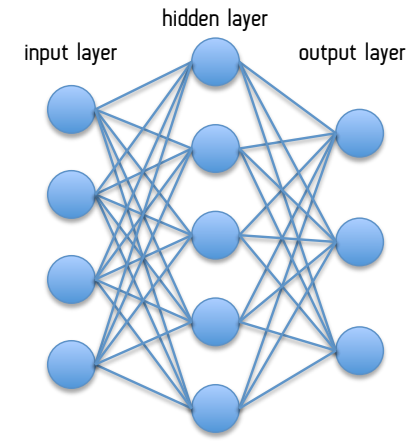
[31] Toon Calders and Sicco Verwer, 'Three Naive Bayes Approaches for Discrimination-Free Classification', *Data Mining and Knowledge Discovery* 21, no. 2 (27 July 2010): 277–92.

[32] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini, 'Discrimination-Aware Data Mining', in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08 (New York, NY, USA: ACM, 2008), 560–568.

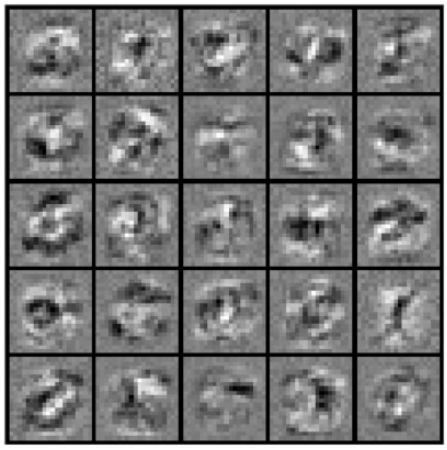
[33] Calders and Verwer, 'Three Naive Bayes Approaches for Discrimination-Free Classification'; Muhammad Bilal Zafar et al., 'Learning Fair Classifiers', *arXiv:1507.05259*, 19 July 2015, http://arxiv.org/abs/1507.05259; Sara Hajian, Josep Domingo-Ferrer, and Antoni Martínez-Ballesté, 'Rule Protection for Indirect Discrimination Prevention in Data Mining', in *Modeling Decision for Artificial Intelligence*, Lecture Notes in Computer Science 6820 (Springer, 2011), 211–22.

[34] B. Yegnanarayana, *Artificial Neural Networks* (PHI Learning Pvt. Ltd., 2009).

layer, in which inputs are received (or perceived), a hidden layer, where various operations are performed, and an output layer for the results. Layers are often fully interconnected, where each node in one layer is connected to all nodes in the adjacent layer (see Fig 2a). Each connection is associated with a weight, such that a node receives an input that is adjusted according to the weight of the connection through which it was received. A node performs a calculation based on the inputs received, and depending on the result, will send various outputs to its connected (output) nodes. Learning entails adjusting the weights of these connections, thereby refining the overall network.



(a) A example ANN topology



(b) An illustration of a hidden layer of a neural network for digit recognition (from Burrell).[35] Each square visually represents the aspects of the handwritten digit the node considers (akin to a "pixel heatmap").

Figure 2: Neural network illustration

Neural networks are considered opaque primarily because their structures provide little insight into the underlying model. To illustrate, Fig 2b from Burrell[36] visually depicts the nodes of the hidden layer for a handwritten digit recognition neural network, by showing the values for the pixels that each node considers based on its input connection weights. We see that though the model is accurate in its classifications, it is far from intuitive. Issues of interpretability are exacerbated in the more recent *deep learning*[37] approaches that entail a number of hidden layers, thereby increasing complexity.

There is, however, ongoing research into mechanisms for *rule extraction,* to assist in understanding by extracting knowledge from more opaque approaches and expressing it in a more intelligible form,[38] such as a decision tree.[39] There are also ways to try to describe what aspects of the input led to a *particular decision* (rather than describing the model as a whole) such as highlighting features of an image that led to a particular classification.[40] These assist in assessing the appropriateness of the model.[41] Control tends to be more challenging for the

[35] Jenna Burrell, 'How the Machine "thinks": Understanding Opacity in Machine Learning Algorithms', *Big Data & Society* 3, no. 1 (1 June 2016).

[36] See Ibid. for a detailed elaboration of issues transparency in neural networks and support vector machines.

[37] G. P. J. Schmitz, C. Aldrich, and F. S. Gouws, "ANN-DT: An Algorithm for Extraction of Decision Trees from Artificial Neural Networks," *IEEE Transactions on Neural Networks* 10, no. 6 (November 1999): 1392–1401.

[38] For an overview, see Jan Zilke, 'Extracting Rules from Deep Neural Networks', *TU Darmstadt, Knowledge Engineering Group*, 2015.

[39] G. P. J. Schmitz, C. Aldrich, and F. S. Gouws, 'ANN-DT: An Algorithm for Extraction of Decision Trees from Artificial Neural Networks', *IEEE Transactions on Neural Networks* 10, no. 6 (November 1999): 1392–1401.

[40] Ribeiro, Singh, and Guestrin, '"Why Should I Trust You?'

[41] Ibid.

more opaque models; though there is continuing work on general means for improving and providing control (such as 'fairness') across approaches.[42]

It is worth noting that often several ML approaches or models are combined in an attempt to improve overall accuracy. This is known as *ensemble learning*.[43] Common approaches include *bagging* and *boosting,* which involve varying training sets for each approach, the latter based on the results of other learners,[44] and *stacking* in which the outputs of several approaches are combined by another learning algorithm.[45] Ensembles can increase levels of complexity; the degree of transparency and control will depend on the nature of the particular ensemble.

2.2 Learning approaches

Often algorithms are categorised according to the class of learning approach they take.

Supervised learning[46] involves learning from input data that is labelled with the desired output/result. For example, there may be a set of images of skin lesions, with some lesions labelled as malignant, others benign. The learning process involves using labels to attempt to infer a model that can be applied to new data; here, to learn what aspects of the image support a malignant or benign diagnosis.

Unsupervised Learning[47] is where there are no desired outcomes/results associated with inputs to guide the learning process; i.e. data is unlabelled. Unsupervised approaches are used to find patterns and relationships based on the characteristics of the data. Such techniques suit clustering problems, separating inputs into 'like groups', and dimensionality reduction, to help reduce the number of variables that require consideration. As such, unsupervised approaches are useful for data mining, and can be effective in assisting in preparing and labeling data for use with supervised learning techniques.[48]

Reinforcement learning differs from supervised/unsupervised learning as it is action-oriented: here the algorithm will take an action, the consequences of which are evaluated by a reward function.[49] A model is formed by the ML process learning the actions to take in order to maximize its reward. Reinforcement learning is often used in control systems, e.g. for robotics and game systems, for instance, the work by DeepMind in combining reinforcement and deep learning approaches to excel at video games.[50]

---

[42] Zafar et al., 'Learning Fair Classifiers'.

[43] David Opitz and Richard Maclin, 'Popular Ensemble Methods: An Empirical Study', *Journal of Artificial Intelligence Research* 11 (1999): 169–198.

[44] Eric Bauer and Ron Kohavi, 'An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants', *Machine Learning* 36, no. 1–2 (n.d.).

[45] David H. Wolpert, 'Stacked Generalization', *Neural Networks* 5 (1992): 241–259.

[46] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, 2nd ed. 2009. Corr. 7th printing 2013 edition (New York, NY: Springer, 2011).

[47] Zoubin Ghahramani, 'Unsupervised Learning', in *Advanced Lectures on Machine Learning*, Lecture Notes in Computer Science 3176 (Springer Berlin Heidelberg, 2004), 72–112.

[48] *Semi-supervised* learning combines supervised and unsupervised approaches, typically where there is a small set of labelled data (perhaps due to the cost of labeling), which helps direct the analysis concerning the larger mass of unlabelled data. See Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien, *Semi-Supervised Learning*, 1st ed. (The MIT Press, 2010).

[49] Though methods can also be combined and in some cases reformulated. For instance, a reward function can be built based on the labels of the supervised learning model (Andrew G. Barto et al., 'Reinforcement Learning and Its Relationship to Supervised Learning', in *Handbook of Learning and Approximate Dynamic Programming* (John Wiley & Sons, Inc., 2004), 45–63.)

[50] Volodymyr Mnih et al., 'Human-Level Control through Deep Reinforcement Learning', *Nature* 518, no. 7540 (26 February 2015): 529–33.

Given the current focus on 'big-data' applications, reinforcement learning is less common than supervised/semi/unsupervised approaches. However, it is relevant to this discussion not only because it paves the way for more 'autonomous agents', but also because the evaluation function and range of possible actions represent points for designer control. In situations such as video games, much of this is defined by the possible directions/inputs of the controller, and the rules of the game. Moving forward, the actions that may be taken must be appropriate for the context in which the system will operate (see §4).

Offline (batch) and online methods[51]

Many learning approaches are *offline* (or *batch*) in nature, in that they use defined set(s) of fully-available data in order to build a *generalizable* model.[52] *Online* methods differ by taking a sequential approach to learning, by using individual data points: the model is presented a data item, and depending on the error/cost of the model's hypothesis to that item, may result in refinement. In this way, the model reflects the data the learner has seen so far.[53]

2.3 Model fit and evaluation

Clearly it is important to rigorously test and evaluate any learned model, to explore its fit for purpose. In practice, often a number of different algorithms will be tried and evaluated in order to select an approach (or ensemble) that best suits the particular problem area.

Evaluating a model concerns its alignment with the general underlying trend(s) in the data. This entails examining outputs to measure a model's performance (error),[54] which includes checking for underfit (bias), where the model fails to sufficiently capture the trends in the data, and overfit (variance) where the model is overly tuned to the data on which the model was built, thereby reducing its general applicability.

In supervised learning, model evaluation typically involves splitting the labelled data into that used for testing and that for training. The algorithms operate on the training data to build the model. Accuracy is evaluated by applying the model against the test data, i.e. 'new' data previously unseen by the model, and comparing outputs with the relevant labels. The closeness of fit of the model to the training data is useful for detecting underfit, and closeness to the test data for detecting overfit.[55]

Evaluating the results of an unsupervised learning approach often relates to the degree of separation between clusters (high intra-cluster similarity, low inter-cluster similarity) and/or considering various statistical features of the data/model. This can involve using test and

---

[51] Note that here, the terms '*online*' and '*offline*' are unrelated to notions of networks and communication, such as "being on the Internet." Further, 'online' does not imply real-time learning!

[52] For more detail on online and offline learning, see: Ofer Dekel and Yoram Singer, 'Data-Driven Online to Batch Conversions', in *Advances in Neural Information Processing Systems 18*, ed. Y. Weiss, B. Schölkopf, and J. C. Platt (MIT Press, 2006), 267–274, http://papers.nips.cc/paper/2775-data-driven-online-to-batch-conversions.pdf.

[53] N. C. Oza, 'Online Bagging and Boosting', in *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, 2005, 2340–2345.

[54] Hastie, Tibshirani, and Friedman, *The Elements of Statistical Learning*.

[55] There are methods that aim at improving a model's performance (test error) estimates. A common approach is *k-fold cross-validation*, which involves partitioning the data into a number of sets, then iteratively running the training exercise using different sets for training and testing, with the error averaged over all iterations. This aims at reducing the propensity for overfit and is particularly useful in situations where data is limited, though the iterations introduce extra computational cost. See Ron Kohavi, 'A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection', in *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'95, 1137–1143.

training data, but as data is unlabelled the comparison concerns whether inputs with similar characteristics are treated in a similar manner.

In reinforcement learning, performance relies heavily on the appropriateness of the reward functions, and the range of possible actions the system is allowed to take. It follows that simulation has a role to play, providing a test environment to help validate that behaviour is proper. Clearly, any simulation environment must be sufficiently representative of the environment in which the model will operate.

Note also that re-evaluation may be necessary, for instance, where the data distribution underlying the application domain changes (see §3.2 and §4.2).

From a responsibility perspective, there is more to consider than just the performance of the learned model.

Depending on the application domain, explainability may well be a significant consideration, which may arise, for example, from data protection obligations.[56] This would impact model assessment, perhaps precluding the use of particular approaches (even if more accurate!) and may warrant the use of knowledge extraction techniques (the outputs of which would also require assessment and evaluation).

Further, in many contexts, intuition and input from domain experts will play an important role in validating models and their results. In an unsupervised learning environment, for instance, external validation can assist in determining whether the clusters appear sensible; e.g. whether a series of news articles grouped by topic and/or source. Any intervention by domain experts, or individuals in general, represents a point of where responsibility may lie. Who is involved, their level of expertise and the nature of their involvement appear relevant considerations.

3. The role of data

ML is data driven: (1) the data involved in the training/learning phases determines the model, and (2) the live data on which the model is applied determines the results/outcomes.

In terms of responsibility, managing the data exposed to a ML process represents a point of control. At one end of a continuum, control may be absolute, determining the precise and complete data used when the algorithm is learning and live. At the other end, control becomes more indirect and might include decisions about the deployment location of sensors to collect data directly from their environment or the design and configuration of a sensor. Whatever the scenario, there is generally the possibility to design in a capability to exercise some degree of control over input data.

3.1 Data for learning

As §2 described, ML algorithms use input data to learn a representative model. Therefore for a model to be appropriate, the data used to build the model *must properly reflect the domain in which the model will be applied*. This includes ensuring that there is appropriate *coverage*, in the sense that the data is sufficiently rich to enable a proper model to be formed, and considering issues of *selection (sampling) bias*, whether the data accurately represents that of the underlying problem domain. Consider an employment application; if the recommender system was built on data concerning Cambridge Dons, the learned model would favour a very particular demographic. For the purposes of a study into algorithmic explanation, a neural

[56] Kamarinou and Millard, 'Machine Learning with Personal Data'.

network for distinguishing between pictures of wolves and huskies was (deliberately) poorly trained,[57] where every picture of a wolf had snow in the background, whereas the huskies did not. The model built would work to classify anything with snow in the background as a wolf, regardless of the animal, its pose, etc.

It follows that the efforts concerning the data: selection, cleansing, *feature selection and engineering* – determining the input data to the ML algorithm – are considered key to ML in practice.[58] This involves aspects of data selection, representation, transformation, and processing,[59] as appropriate for both the ML technique and the learning goals. To quote Domingos:[60]

> *"First-timers are often surprised by how little time in a machine learning project is spent actually doing machine learning. But it makes sense if you consider how time-consuming it is to gather data, integrate it, clean it and pre-process it, and how much trial and error can go into feature design."*

He goes on to note that learning is an iterative process, that often requires re-engineering the data and repeating the learning processes. It is therefore natural that there is work on automating feature selection, e.g. by using learning techniques that can help identify relevant features in datasets.[61]

Feature engineering focuses on technical aspects of learning. However, there are associated responsibility implications to ensure the data is (and remains) representative and appropriate to the area in which it will apply (thereby influencing the model), and the data usage accords with any data protection (e.g. purpose limitation) or other legal or regulatory obligations (e.g. non-disclosure).[62]

The volume of data is also important. A sample of insufficient size will be unrepresentative, making it more difficult to detect situations of under/overfit.

In specialised areas, such as medicine and cybersecurity for example, domain-specific expertise will also be required – both to generate and provide data as input, in addition to assisting in the design process, such as assessing the value and appropriateness of outputs. In practice it is prudent to have redundancy, i.e. having several opinions in an attempt to minimise error/bias.

Such concerns are relevant for all learning approaches (§2.3). Supervised learning introduces an extra consideration, to ensure that data is properly labelled. In practice, the processing of labelling can be outsourced. It is not uncommon to see jobs advertised on Amazon's Mechanical Turk[63] that offer people a small sum to label particular data, and CAPTCHAs – simple tasks for humans that attempt to filter for 'bots' – can double as a labelling tool.[64] These crowdsourced

---

[57] Ribeiro, Singh, and Guestrin, '"Why Should I Trust You?'.

[58] Andrew Ng: "Applied machine learning is basically feature engineering"
http://ai.stanford.edu/~ang/slides/DeepLearning-Mar2013.pptx

[59] Possibly including accounting for sampling error: Corinna Cortes et al., 'Sample Selection Bias Correction Theory', in *Algorithmic Learning Theory*, ed. Yoav Freund et al., Lecture Notes in Computer Science 5254 (Springer Berlin Heidelberg, 2008), 38–53; Aditya Khosla et al., 'Undoing the Damage of Dataset Bias', in *Computer Vision – ECCV 2012*, ed. Andrew Fitzgibbon et al., Lecture Notes in Computer Science 7572 (Springer Berlin Heidelberg, 2012), 158–71.

[60] Pedro Domingos, 'A Few Useful Things to Know About Machine Learning', *Commun. ACM* 55, no. 10 (October 2012): 78–87.

[61] A Coates, H Lee, and AY Ng, 'An Analysis of Single-Layer Networks in Unsupervised Feature Learning', *AISTATS*, 2011.

[62] Kamarinou and Millard, 'Machine Learning with Personal Data'.

[63] http://www.mturk.com

[64] http://www.google.com/recaptcha/intro/index.html

approaches suit simple perception tasks such as describing what is in an image, and are particularly useful where there are large volumes of data to label. In more complex application domains, expert opinion may be required for labelling. Again, redundancy can be used to deal with issues of label quality, where inputs are separately labelled several times by different parties. This helps in gaining consensus, or indeed, in identifying interesting edge cases.

As noted earlier, issues of responsibility and control will arise with respect to the appropriate choice of experts, as well as how that expertise is discharged. While transparency obligations regard data input, any labeling practices and the use of experts would facilitate accountability.

3.2 Data in the wild

Any model built for practical deployment[65] will operate on real-world data. As such, it is important to ensure that (1) the model remains representative, and (2) the data provided to the model is constrained appropriately.

Ongoing representativeness

We have outlined how a model is learned from data: for a model to be both accurate and useful, that data on which it was built must be suitably representative of the domain in which the model will be applied. However, in the real world, things change; over time, the properties of the data distribution may change and evolve (*concept drift)*, rendering a once accurate model obsolete.[66] Such a change may be gradual or sudden, and occur in a wide range of application domains – particularly notable are areas such as spam, intrusion and fraud detection, where an adversary directly attempts to circumvent a model.[67]

It is therefore important to monitor and evaluate the properties and nature of the data of the application domain, so that models can be updated where necessary (see §4.2). Towards this, there are methods for detecting,[68] and learning approaches that account[69] for concept drift.

Input constraints

Recall the example of Tay, Microsoft's Twitter bot that would take tweets as input data and use these to produce her own tweets. Tay presented a teenage persona, demonstrating a natural language capability using millennial slang.[70] However, once 'live,' Tay took as input tweets from a variety of Internet trolls, which in turn resulted in Tay's tweets becoming foul-mouthed and racist.[71]

In practice, the appropriate measures will depend on the context in which the system operates. In the Twitter example, such an outcome is rather foreseeable given notoriety of Internet trolls. This situation could have been mitigated through data management techniques; very simple

[65] cf. those purely aimed at research.

[66] See: Alexey Tsymbal, 'The Problem of Concept Drift: Definitions and Related Work', Tech. report, Department of Computer Science, Trinity College Dublin, (2004).

[67] Indrė Žliobaitė, 'Learning under Concept Drift: An Overview', *arXiv:1010.4784 [Cs]*, 22 October 2010, http://arxiv.org/abs/1010.4784.

[68] Maayan Harel et al., 'Concept Drift Detection Through Resampling', in *International Conference on Machine Learning (ICML-14)*, 2014, 1009–17.

[69] For links to a range of approaches, see: Yamini Kadwe and Vaishali Suryawanshi, 'A Review on Concept Drift', *OSR Journal of Computer Engineering* 17, no. 1 (2015): 20–26.

[70] http://www.digitaltrends.com/social-media/microsoft-tay-chatbot/

[71] Horton, 'Microsoft deletes "teen girl" AI after it became a Hitler-loving sex robot within 24 hours'.

approaches could include filtering tweets based on the types of words used,[72] or only accepting tweets from preselected or reputable users (e.g. based on number of followers, number of tweets).

3.3 Responsibility and data management

From the above, we see there are two key concerns: *quality* – the (continued) representativeness of the data and any metadata (labels), and *constraints* – to ensure that only the appropriate data is used as input. Data quantity is perhaps best seen as a dimension of quality rather than its own criterion; since more data may, or may not necessarily, mean better quality.[73]

This relates to responsibility. Data input selection represents a point for intervention, allowing the ML processes to be managed by way of the data used, even if the inner workings of the algorithm are fairly opaque.

Beyond the specific statistical/feature engineering considerations, ongoing systems research may assist with managing the responsibilities associated with data used in ML. We indicate some areas of this in §6; in short, there is work on data provenance to illuminate the origins and path of data, useful for assessing the quality, reliability and representativeness of data; particularly where data is combined from a number of sources. *Differential privacy* techniques[74] are relevant where data is personal (see §6). General security mechanisms such as access controls are also relevant, for example, to set bounds on the inputs of data to a deployed ML system.

3.4 Where does the value lie?

While the algorithm and the models it learns are seen as being where the value resides in ML, the data is often viewed as a free or under protected raw material within the process. In reality, an algorithm and its model are intrinsically tied to the data used as part of the model building process.

In terms of public law treatment, an ML algorithm, as a form of computer program, would usually be automatically protectable as a copyright work and potentially a patentable invention.[75] By contrast, data input into a ML system would generally only be protected as a 'compilation' in copyright or a 'database' if the data set had been assembled in an appropriate manner.[76] In terms of the learned model, i.e. the output from the ML algorithm and the input data, English law recognizes the concept of a 'computer-generated' work, which grants copyright authorship to "the person by whom the arrangements necessary for the creation of the work are undertaken";[77] as well as being protectable as a form of confidential information.[78]

---

[72] For example, a 'swear filter' was introduced for IBM's Watson after it had learned slang from UrbanDictionary.com: http://www.ibtimes.com/ibms-watson-gets-swear-filter-after-learning-urban-dictionary-1007734

[73] http://www.datasciencecentral.com/profiles/blogs/7-cases-where-big-data-isn-t-better

[74] Cynthia Dwork, 'Differential Privacy: A Survey of Results', in *Theory and Applications of Models of Computation*, ed. Manindra Agrawal et al., Lecture Notes in Computer Science 4978 (Springer Berlin Heidelberg, 2008), 1–19.

[75] Copyright Designs and Patents Act 1988 ("CDPA'), s. 3(1)(b) and the Patents Act 1977, s. 1(2)(c). While the latter states that "a scheme, rule or method for performing a mental act, playing a game or doing business, or a program for a computer" is not patentable subject matter, this is limited to where the application relates to that thing 'as such', which permits certain software-implemented inventions. See generally Abbott, R., "'I think, therefore I invent: Creative computers and the future of patent law', *Boston College Law Review*, vol. 57, no. 4, 2016.

[76] A 'compilation' is recognised as a 'literary work' at CDPA, s. 3(1)(a), while a 'database' is protected under the Copyright and Rights in Databases Regulations 1997 (No. 3032).

[77] CDPA at s. 9(3) and 178.

In terms of private law, data banks and data brokers will be able to control the use of their data collections through contract or licence agreements; although public law intervention may be required where such mechanisms have adverse public interest impacts, such as anti-competitive consequences, although it could potentially include the effectiveness of an ML system.

While it is beyond the scope of this paper to examine the interaction between ML and intellectual property laws, it is worth noting that, to the extent that the legal treatment of the different components of the ML process varies, it will impact the value of each component which may, in turn, have an impact on the responsibility for, and control exercised over, each component.

4. Systems and processes

The previous discussion largely focused on the ML itself. The practical effects, however, concern the deployment of the technology, as it is integrated into workflows and into broader systems.

4.1 Workflows & processes

Integrating ML into a workflow entails its operation on particular data, in a particular context, the outputs of which bring about a particular result.[79]

A human in the loop

Some ML processes will require people to be directly involved. This might be to select the appropriate inputs to feed into the ML process, or to revert back to humans when particular situations arise, e.g. to ask for clarification or a judgment to be made, or even explicitly to hand back control given the gravity of the particular context or decision, such as whether to apply lethal force.[80] ML outputs may also directly feed back to an individual. Recommender systems are a clear example, such as IBM's Watson, that can provide physicians with a list of "potential diagnoses along with a score that indicates the level of confidence for each hypothesis."[81] Where the outputs of a ML system are to an individual, it allows a human to decide the next steps (and take consequent responsibility).

It follows that having a human in the loop represents a clear point for exercising judgement, intervention and control.

There are, of course, concerns as to the degree of agency: does the person just blindly follow the machine?; does one have sufficient information to inspect and verify the quality of the output?[82] Further, there are questions as to whether it is appropriate to defer to a person in the

---

[78] See generally *Gurry on Breach of Confidence* (Eds. Aplin, Bently, Johnson and Malynicz), OUP, 2012.

[79] §3 considered input data, which must be within scope of the data in which the model was trained. In this section, we focus more on the practical effects of ML outputs.

[80] For instance, the US Department of Defense Directive 3000.09 (2012) states "Autonomous… weapon systems shall be designed to allow… appropriate levels of human judgment over the use of force."

[81] http://www-05.ibm.com/innovation/uk/watson/watson_in_healthcare.shtml

[82] This will depend on the ML technique used. IBM's Watson allows inspection by presenting the documents forming the basis for the decision. There is also work on more general mechanisms to provide insight into how one particular decision/output was made, in order to improve levels of user trust in the 'machine': Ribeiro, Singh, and Guestrin, '"Why Should I Trust You?'

first place or, indeed, whether it may be more appropriate to appeal to a machine![83] Such considerations will depend on the circumstances; however, having a person explicitly included in the workflow represents a clear, well-established point of responsibility.

Automated environments

Systems may also be constructed such that ML outputs *automatically* result in actions – moving through workflow processes – *without* direct human involvement. This can make assessment and intervention more difficult as actions are taken in real-time without the defined 'stopping points' in which systems defer to humans.

This becomes particularly relevant in the context of the emerging *Internet of Things* (IoT) – which aims towards seamlessness, automation and personalisation – in an environment where 'things' have mass and velocity, whereby actuations effect changes in the physical world.[84] For instance, there are visions for personal digital assistants that automatically control one's surrounding environment, such as the lighting, heating, music, and much, much more, based, for instance, on learned individual preferences and current context such as an individual's mood.[85] These agents could potentially conflict, giving rise to disputes over control hierarchies, e.g. who is 'in charge' of the family home?

The potential for harm can be exacerbated in automated environments. The systems and interactions involved tend to be more opaque; inappropriate or harmful outputs/actions may occur and go unnoticed (for a period), means for intervention can be less explicit, and exercising control can be more difficult given issues of complexity and timing – in general humans are slower than computers!

Where a system has some autonomy to act, and thereby has the potential to cause harm, responsibility would seem to reside with the persons that control (or manage) its inputs, functions, actions and deployment; while transparency obligations may be owed to those on the receiving end of the process.

Therefore, in practice, not only is it important to ensure the learned model is appropriate, but that there are the appropriate checks and constraints in the workflows and data inputs/outputs that surround it, and that all these are recorded to enable subsequent audit. That is, the inputs and outputs of ML systems may require bounds, subject to particular parameter ranges (e.g. temperature limits for a thermostat), constraints regarding the frequency of actions (e.g. cannot turn something on/off repeatedly); or be passed through sanitisation/verification procedures before being processed or actioned. There is extra complexity in that not all of these aspects can be defined *a priori,* but in many situations it will be necessary to account for environmental and contextual information that will change (in anticipated and unexpected ways) over time. This renders mechanisms for rigorous audit particularly important, to aid transparency and investigation where necessary.

Flow-on effects

On complexity, some ML processes will operate in a separate, standalone context. For example, ML in a game or control system represents a closed and well-understood environment. Systems that feed back to humans, such as the skin lesion diagnosis example mentioned earlier, are

[83] Kamarinou and Millard, 'Machine Learning with Personal Data'.

[84] J. Singh et al., 'Twenty Security Considerations for Cloud-Supported Internet of Things', *IEEE Internet of Things Journal* 3, no. 3 (June 2016): 269–84.

[85] Mireille Hildebrandt, *Smart Technologies and the End(s) of Law: Novel Entanglements of Law and Technology* (Cheltenham, UK: Edward Elgar Pub, 2015).

naturally constrained in that a human must take the next step. In both situations there is clearly potential for harm resulting from the ML process; however, the environment, workflows and points of interaction are (comparatively) well-defined, facilitating risk assessment and mitigation.

Consider large-scale computing environments, for example IoT-enabled smart cities, that entail 'systems of systems'.[86] Such environments have many 'moving parts' – including a range of different software, services, agents (and people!) – all of which might use or be affected by a range of ML models. Managing responsibility in these environments presents a significant challenge. There will be feedback loops between systems, where the outputs/actions of one system can feed into others in real-time. The interactions can be direct, e.g. competing for resources, or more indirect, through 'butterfly effects', where (subtle) actions of a system can (potentially dramatically) affect others. It is therefore important that management regimes can be applied broadly, within and across workflows and systems (see §5).

4.2 Deployed ML models and updates

A deployed ("in production") ML model, i.e. integrated into a system/workflow, operates on real-world (or 'live') data, to make predictions, classifications, decisions, etc.

There will be many situations in which a model will need to be updated and changed. This might be, as mentioned in §3.2, due to the nature of the data evolving over time which affects a model's accuracy. Or alternatively, updates may be required due to a broader or overt change in circumstance that requires an active intervention; for example, the advent of a new social media platform would impact sentiment analysis models, or an autonomous vehicle failing to identify obstacles in particular lighting conditions[87] raises urgent safety considerations. Again, such concerns relate to the responsibility to ensure a model remains fit for purpose.

Towards this, a model *"build(retrain/update)-and-redeploy"* approach is common practice.[88] The nature of any such process will vary, depending on the situation. Consider self-driving vehicles, where learning tasks and data are complex, and there are broader concerns such as safety. Here, given the vast amounts of traffic, city and vehicle data that require processing, one would expect that driving models would be computed in a cloud (or cluster), where the learned models are rigorously tested before updates are sent to vehicles. A vehicle would install the model, and apply it on the inputs it receives from its many sensors. The vehicle would also send certain data back to the cloud for analysis and to assist future learning. Updates in such a circumstance may be relatively infrequent and ad-hoc, compared to other application areas such as finance or news-trend analytics, where periodic retraining might occur more frequently – perhaps in the order of hours.[89] For these later examples, the retraining and updating processes will likely be automated.

Note that online learning approaches – that learn based on individual inputs (§2.3) – are amenable to scenarios where the model is continually and automatically refined rather than redeployed.[90] However, from a responsibility perspective, similar concerns remain. That is, even where a model can 'self-adapt', care must be taken to ensure that the proper monitoring

[86] J. A. Stankovic, 'Research Directions for the Internet of Things', *IEEE Internet of Things Journal* 1, no. 1 (February 2014): 3–9, doi:10.1109/JIOT.2014.2312291.

[87] http://electrek.co/2016/07/01/understanding-fatal-tesla-accident-autopilot-nhtsa-probe/

[88] See, for example, Amazon's and Microsoft's guidance on model retraining: http://docs.aws.amazon.com/machine-learning/latest/dg/retraining-models-on-new-data.html, and https://azure.microsoft.com/en-gb/documentation/articles/machine-learning-retrain-models-programmatically/

[89] Ibid.

[90] For an overview and links to various online approaches to dealing with adaptive data, see Žliobaitė, 'Learning under Concept Drift' and Kadwe and Suryawanshi, 'A Review on Concept Drift'.

and constraints are in place. Further, there will still exist situations requiring more direct interventions, e.g. to prevent and respond to incidents that (could) cause harm.

4.3 Provisioning ML

Also relevant to issues of responsibility are the practical aspects of provisioning ML-driven systems.

Skills & expertise

Before considering more specific aspects of ML provisioning, it is worth noting that skills and expertise play an important role. Machine learning brings together a range of specialities, including computer science, mathematics, statistics, probability, optimisation, information theory, and data management. Expertise and skill are involved in: selecting the appropriate ML technique(s), data and features, as appropriate for the problem domain, which must also be tuned and managed; defining the appropriate learning, evaluation and optimisation functions and procedures; avoiding overfit/underfit, where the model fails to align with the general underlying trend(s); and importantly, as we see ML-driven systems increasingly deployed, ensuring that system behaviour remains appropriate, in line with the requisite responsibilities and obligations,[91] which may well evolve over time.

It follows that engineering ML systems requires significant expertise. Beyond building functionality, an inappropriate model and/or a model's unintended or improper use may have significant consequences. From a responsibility perspective, it is therefore important that those involved in the technical development of a ML system recognise (to the extent possible), and avoid, potential issues. This also entails considering the broader context in which ML systems will, or have the potential, to be used, and may require input from domain experts.

There is a challenge in that despite the increasing demand for ML-driven applications, there are reported shortages in those skilled in the relevant disciplines.[92] Experience is crucial; indeed, it is said that "developing successful machine learning applications requires a substantial amount of black art".[93] ML tools and services (e.g. MLaaS) are becoming more widely available, which work to improve access to ('democratising')[94] ML functionality, by reducing the level of skills and expertise required for leveraging such techniques. However, given the complexity and domain knowledge associated with ML, coupled with the difficulty in determining software quality and correctness,[95] questions will arise regarding what aspects can practically be 'outsourced' to others, and how this impacts responsibility, transparency and control.

Skills and training must also be considered for those providing input into ML processes (i.e. domain rather than technical experts), as well as for those using and/or subject to the ML systems, e.g. to ensure they fully understand the context and any limitations.

---

[91] Reed, Kennedy, and Silva, 'Responsibility, Autonomy and Accountability: Legal Liability for Machine Learning'; Kamarinou and Millard, 'Machine Learning with Personal Data'.

[92] House of Commons, Science and Technology Committee, 'Digital Skills Crisis', *Second Report of Session 2016-17* HC 270 (n.d.).

[93] Domingos, 'A Few Useful Things to Know About Machine Learning'.

[94] R. Barga, D. Gannon, and D. Reed, 'The Client and the Cloud: Democratizing Research Computing', *IEEE Internet Computing* 15, no. 1 (January 2011): 72–75.

[95] See *System Supply Chains*, below.

Risk assessments

An appropriate risk assessment should be carried out prior to any deployment. The EU General Data Protection Regulation, for example, obliges a 'controller' to carry out a 'data protection impact assessment' where new technologies are deployed and the processing is likely to have a 'high risk' of interfering with the rights and freedoms of data subjects.[96] Given ML entails states that are not explicitly pre-programmed, and given that the degree for intervention and imposing constraints can be limited or difficult, risk assessments are likely to become integral to the process of deploying and operating ML systems.

There also appears a role for standards,[97] to assist and guide ML development and deployment processes and to ensure appropriate consideration has been given to the risks. Indeed, given the potential for uncertain outcomes, periodic assessments may need to be carried out throughout the life cycle of the ML application.

Generally, it would appear prudent to maintain detailed records of the decisions and processes involved in all aspects of building an ML system – from data collection and management, feature engineering, model building and systems integration.

System supply chains

We currently see much high-profile ML work taking place in the context of a single organization. Moving forward, the provisioning of ML-driven systems could entail long, potentially complex system supply chains, encapsulating offerings across a range of organisations.

Often those seeking to build and deploy ML systems will seek additional data to that they hold 'in-house'. This includes data sets, stored data such as the sales transaction history for an organisation, as well as access to live feeds, e.g. from sensors or stock-quote data. Relevant data may come from a wide range of sources, especially in an IoT context,[98] and has the potential to be useful for a variety of purposes (known as *data repurposing*[99]). Data aggregators and/or brokers are emerging. These need not be centralised and/or privacy invading.[100]

Cloud will play an important role in facilitating the development and deployment of ML systems. Firstly, cloud facilitates data management and access in light of the potential volume, range of sources, and geographic spread of the data involved. Some ML learning processes, such as deep learning, require significant compute resources to undertake complex learning tasks. These capabilities are highly amenable to a cloud-based offering, improving accessibility.

There are offerings that aim to simplify aspects of ML engineering. Some MLaaS services provide platforms for building and refining custom models, for example through facilitating experimentation.[101] And again, ML software toolkits are available that provide access to ML techniques 'out of the box'. Clearly these will still require some degree of expertise and skill to leverage. We are also seeing the emergence of MLaaS offerings, particularly by the larger firms

---

[96] See General Data Protection Regulation 2016/679 (OJ L 119/1, 4.5.2016), at Article 35. See also Kamarinou and Millard, 'Machine Learning with Personal Data'., at 2.4.

[97] Diakopoulos, 'Accountability in Algorithmic Decision Making'.

[98] Stankovic, 'Research Directions for the Internet of Things'.

[99] Nuffield Council on Bioethics, 'The Collection, Linking and Use of Data in Biomedical Research and Health Care: Ethical Issues', 3 Feb 2015, http://nuffieldbioethics.org/wp-content/uploads/Biological_and_health_data_web.pdf.

[100] For example, see http://www.databoxproject.uk/ and http://hubofallthings.com/

[101] https://techcrunch.com/2016/02/16/google-makes-it-easier-to-take-machine-learning-models-into-production/

that provide pre-trained ML models – further broadening access by offering a 'plug-and-play' capability, e.g. for voice recognition, document translation and natural language processing (e.g. *Parsey McParseface*[102]) that can readily be incorporated into applications and services.

This relates to responsibility, as it entails some level of reliance and trust in the various components used; components may be leveraged without a need for (a detailed) understanding of the internals. Issues of software quality and reliability (bugs) – assessing software and validating its functionality is difficult – become particularly relevant given the algorithmic-intensive nature and complexity of ML software.[103] Concerns are exacerbated when considering the broader systems-context in which ML operates, as a deployment might entail a wide-ranging composition of software, systems and services.

In short, the system supply chains supporting ML systems have the potential to be complex. While there will (or should) be chains of contracts to deal with the composition of these services, *issues of responsibility pervade*; particularly given the transparency and control issues that ML introduces. Being able to map the components of a ML system or process and the manner of their interaction will require a degree of transparency that inevitably goes beyond the words that appear in any chain of contracts.

5. Management technology

There is ongoing research and development into techniques, technologies and tools that may assist in managing responsibility, by improving levels of control, transparency and visibility over ML systems. As we mentioned in §2, this includes work on improving the transparency of ML techniques, particularly where the algorithms are naturally opaque; for instance, rule extraction approaches that aim to uncover the model underpinning a deep neural network, representing it in a more interpretable form.[104] Many techniques to improve model transparency entail a discovery process that involves exploring the impact that inputs have on resulting outputs.[105] An emerging area of work focuses on explainability, to provide evidence and features of data relied upon by the model in producing its output, in order to better indicate to users the nature of the model and decision/prediction.[106]

Generally, there seems scope for ML techniques to not only assist with understanding and inspecting ML models, but also to help manage and control their operation as part of more complex systems. ML techniques are already being explored for measuring software quality,[107] which could pave the way for evaluating components in a systems composition context. Moving forward, one can envisage ML systems that can manage the interface between other ML systems, for example, to mediate between an individual's personal device and the systems in their surrounding physical environment.

---

[102] https://research.googleblog.com/2016/05/announcing-syntaxnet-worlds-most.html

[103] F. Thung et al., 'An Empirical Study of Bugs in Machine Learning Systems', in *2012 IEEE 23rd International Symposium on Software Reliability Engineering*, 2012, 271–80.

[104] Zilke, 'Extracting Rules from Deep Neural Networks'.

[105] Datta, Anupam, Shayak Sen, and Yair Zick, 'Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems'.

[106] Ribeiro, Singh, and Guestrin, 'Why Should I Trust You?'

[107] Saiqa Aleem, Luiz Fernando Capretz, and Faheem Ahmed, 'Benchmarking Machine Learning Technologies for Software Defect Detection', *arXiv:1506.07563 [Cs]*, 24 June 2015, http://arxiv.org/abs/1506.07563; S. Lessmann et al., 'Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings', *IEEE Transactions on Software Engineering* 34, no. 4 (July 2008): 485–96.

We have described how data drives ML systems. It follows that work on data management is of increasing importance to both understand and constrain ML systems.

Data provenance,[108] the ability to track (and therefore visualise) sources of data and the flow of data throughout systems – including across technical, geographic, organizational, administrative and political boundaries – is of increasing importance. Provenance appears useful in managing issues of data quality and representativeness. Further, as outputs from ML (including actions/actuations) are also data, provenance techniques can also be used for tracking the flow of ML inputs and outputs. This could help in improving the visibility of systems as a whole, highlighting the impacts and consequences of the ML system, and indicating the circumstances leading to particular effects. Such information is useful for determining and correcting system errors (i.e. technical concerns), as well as for ascertaining compliance and/or determining fault where breaches occur or harm is caused.

In addition to visibility, there is also work on enabling proactive *control*. Access controls[109] surrounding all the components of ML systems are needed, particularly those that are context aware.

Privacy is a relevant concern,[110] particularly as the world becomes increasingly instrumented (with sensors), and because ML is dedicated to building complex models and associations from data. Methods for managing privacy in data analytics contexts are gaining in prominence, such as *differential privacy* techniques that regulate statistical queries to balance the utility of the results with the probability of identifying individual records.[111] Apple recently announced their uptake of differential privacy techniques.[112] There is also work on new infrastructures; directions in cloud computing aim at supporting 'smaller clouds',[113] that pave the way for personal clouds and data stores that provide more control over the processing operations undertaken and the data released.[114] These could be integrated to improve privacy within the system supply-chain.

Research is ongoing into *policy-driven systems,* exploring mechanisms for managing systems in line with higher-level (user) concerns and environmental context. A current focus of this research community is on wide-scale complex systems, such as systems-of-systems and the IoT.[115] One approach we have explored that we feel shows promise is *Information Flow Control*, where security/management policy is coupled with data, enforced and audited within and across systems – enabling provenance *and* control,[116] which may assist with compliance.[117]

Determining best practices concerning the building, design, testing and impact of ML systems appears a sensible step forward. Formal definitions of these, i.e. standards and procedures,

---

[108] Lucian Carata et al., 'A Primer on Provenance', *Queue* 12, no. 3 (March 2014): 10:10–10:23, doi:10.1145/2602649.2602651.

[109] Vincent C. Hu, David Ferraiolo, and D. Richard Kuhn, *Assessment of Access Control Systems* (US Department of Commerce, National Institute of Standards and Technology, 2006).

[110] Kamarinou and Millard, 'Machine Learning with Personal Data'.

[111] Dwork, 'Differential Privacy'.

[112] https://www.wired.com/2016/06/apples-differential-privacy-collecting-data/

[113] Jon Crowcroft et al., 'Unclouded Vision', in *Proceedings of the 12th International Conference on Distributed Computing and Networking*, ICDCN'11 (Berlin, Heidelberg: Springer-Verlag, 2011), 29–40.

[114] Hamed Haddadi et al., 'Personal Data: Thinking Inside the Box', *arXiv:1501.04737 [Cs]*, 20 January 2015, http://arxiv.org/abs/1501.04737.

[115] Singh et al., 'Twenty Security Considerations for Cloud-Supported Internet of Things'.

[116] T. Pasquier et al., 'CamFlow: Managed Data-Sharing for Cloud Services', *IEEE Transactions on Cloud Computing* PP, no. 99 (2015): 1–1, doi:10.1109/TCC.2015.2489211; T. F. J. M. Pasquier et al., 'Information Flow Audit for PaaS Clouds', in *2016 IEEE International Conference on Cloud Engineering (IC2E)*, 2016, 42–51.

[117] J. Singh et al., 'Data Flow Management and Compliance in Cloud Computing', *IEEE Cloud Computing (Special Issue on Legal Clouds)* 2, no. 4 (July 2015): 24–32, doi:10.1109/MCC.2015.69.

could guide and encourage the development of technologies facilitating compliance; assuming, of course, adoption is incentivised, including through liability regimes targeted at those having control over ML systems.

6. Concluding remarks

ML (and AI) has gone through peaks and troughs of popularity. However, given we are currently in the era of big data and the IoT, with all the supporting infrastructure (networks, storage, compute, cloud), this time ML appears here to stay.

From a responsibility perspective, the key concern is the *impact* of a ML system and the *control* that can be exercised over it. The goal of this discussion paper was to highlight that, in practice, there are a number of aspects to a ML system that require consideration: the ML technique and learning model; the 'training' and 'live' data; the workflows and potential effects of ML outputs; the deployment specifics and supply chains; the integration of the system into environments of scale; the context(s) in which the system operates; and the skills and expertise of the individuals involved.

Managing responsibility concerns the ability to inspect, constrain and intervene. As such, mechanisms that assist audit, transparency, and particularly *control,* within *and across* systems and services, become all the more important.