OPEN ACCESS

University of BRISTOL

Gurevych, I., Meyer, C., Binnig, C., Johannes, F., Kersting, K., Roth, S., & Simpson, E. (2018). Interactive Data Analytics for the Humanities. In *CICLing 2017: Computational Linguistics and Intelligent Text Processing* (pp. 527-549). (Lecture Notes in Computer Science ; Vol. 10761). Springer. https://doi.org/10.1007/978-3-319-77113-7_41

Peer reviewed version

Link to published version (if available):
10.1007/978-3-319-77113-7_41

Interactive Data Analytics for the Humanities

Iryna Gurevych, Christian M. Meyer, Carsten Binnig, Johannes Fürnkranz, Kristian Kersting, Stefan Roth, and Edwin Simpson

Technische Universität Darmstadt
Department of Computer Science
http://www.informatik.tu-darmstadt.de

Abstract. In this vision paper, we argue that current solutions to data analytics are not suitable for complex tasks from the humanities, as they are agnostic of the user and focused on static, predefined tasks with large-scale benchmarks. Instead, we believe that the human must be put into the loop to address small data scenarios that require expert domain knowledge and fluid, incrementally defined tasks, which are common for many humanities use cases. Besides the main challenges, we discuss existing and urgently required solutions to interactive data acquisition, model development, model interpretation, and system support for interactive data analytics. In the envisioned interactive systems, human users not only provide annotations to a machine learner, but train a model by using the system and demonstrating the task. The learning system will actively query the user for feedback, refine its model in real-time, and is able to explain its decisions. Our vision links natural language processing research with recent advances in machine learning, computer vision, and data management systems, as realizing this vision relies on combining expertise from all of these scientific fields.

1 Challenges in Analyzing Humanities Data

Automated data analytics, aka. data mining and machine learning, is a key technology for enriching and interpreting data, making informed decisions, and developing new data-driven scientific methods across many disciplines in industry and academia. Although the potential of interactive problem solving was recognized early on [16], this field has not progressed very far beyond the initial work. In particular, interactive machine learning and data analytics have only recently received increased attention [98]. Current data analytics solutions focus predominantly on well-defined tasks that can be solved by processing large, homogeneous datasets available in a structured form. Consider for example recommender systems, which suggest new products based on the product's properties, the products that the customer has previously bought, and the collective behavior of the customer database [43]. The state-of-the-art relies on huge amounts of data—over one billion pairs of users and news items passively gathered—to train a deep neural network [28]. This may explain, why data analytics is conceived in a rather impersonal way, with algorithms working autonomously on passively collected data, although practice is quite the opposite. Most of the influence practitioners have, comes through interacting with data, including crafting the data and examining results.

In the late 1990s, digitized data became widely available in the humanities as well. Since then, there has been a clear demand for data analytics approaches to tap into these

The final publciation is available at Springer via http://dx.doi.org/10.1007/978-3-319-77113-7_41

textual and visual data, including cultural heritage collections. The research questions and strategies in the humanities are, however, radically different from data analytics tasks in other disciplines.

First, despite the large amount of digitized data, there is typically only a tiny fraction that qualifies as training data for machine learning systems, because most of the data lacks cleaning, preprocessing, and gold standard labels. Data preparation tasks are often highly complex in the humanities. For text, they range from transcribing Gothic script or handwriting through the labeling of references to persons and their actions to a manual analysis of the text's argumentative structure. For images and video, e.g., we need to correct distortions, annotate gestures, or manually describe scenes. Rather than depending on big input data, future data analytics methods for the humanities must therefore be able to cope with *small data scenarios*, generalize from few input signals, and at the same time avoid overfitting to the idiosyncrasies of the dataset.

Second, the analysis of humanities data requires highly specific *expert knowledge*. This may include historical and legal facts, understanding ancient and special languages, or recognizing gestures or architectural properties in images and video. Relying on expert knowledge further limits our possibilities to manually label data, as common annotation procedures, such as crowdsourcing [54] or gamification [2], can only be used for certain subproblems or must be customized for laypeople. An even more severe problem is, however, interpreting the output of a data analytics system, which is only possible with expert domain knowledge. So far, training such a system requires vast machine learning expertise, preventing domain experts from directly participating in the development process. Inspecting and refining a model is particularly challenging in neural network architectures, as there is still little insight into the internal operation and behavior of complex models [109]. Future methods need to communicate directly with domain experts and allow them to steer the data analytics process.

Third, most research questions in the humanities are not clearly defined in advance, but developed over time as the research hypothesis evolves. We therefore need data analytics methods that allow for *fluid problem definitions*. This is particularly true for subjective tasks, for which multiple, partially contradicting theories co-exist. Examples are different schools and traditions in philosophy as well as disparate sources and opinions in history or law. Rather than aiming at a single, universal problem definition, we thus need methods that adapt to particular users or theories and recognize shifting goals.

Although some of these challenges are relevant for data science tasks in general (e.g., the small data scenario in the biomedical domain [93]), fluid problem definitions are prototypical for the humanities, as researchers have to pursue and develop competing theories and standpoints before judging them according to their merits. The humanities therefore need specific solutions for future data analytics. This requires a close cooperation of natural language processing and computer vision with machine learning and data management systems research.

2 Interactive Data Analytics

In this paper, we advocate research on *interactive machine learning* approaches for *data analytics tasks in the humanities*. Interactive machine learning is characterized by incremental model updates based on a user's actions and feedback, yielding a system that is simultaneously developed and used. Rather than teaching a machine learning system with a predefined set of training instances, as is the most common practice today, we envision an intelligent system that a user teaches by *using* it. This is triggered by the insight that a user will not necessarily start with a pre-defined concept that must be modeled as accurately as possible (as is often assumed in machine learning); the concept sought after is likely to evolve during the discovery process and, hence, during the process of selecting data and training a machine learning system.

Indeed, this is akin to *active learning*—the system may ask the user to label a certain instance while learning—but transgresses it by removing the strong focus on data labeling. Active learning removes the passivity of the learning system which, in the classical setting, only receives data, and allows it to actively pose questions on the data. However, the teacher (i.e., the human expert) is passive in the sense that she has no direct influence on the models that the learner induces from the data. Her only way of influencing the results is via the provided data or labels. For that reason Shivaswamy and Joachims [94] extended this towards *coactive learning*, where the teacher can also correct the learner during learning if necessary, providing a slightly improved but not necessarily optimal example as feedback.

In *interactive learning*, we envision a process where the teacher and the learner not only interact at the data and example level, but also at the model level itself. The user should be enabled to directly interact with the model, to provide feedback on the model that influences the learner, or to even directly modify parts of the learner. This way, learning becomes a fully co-adaptive process, in which a human is changing computer behavior, but the human also adapts to use machine learning more effectively and adjusts his or her data and goals in response to what is learned. This requires on the one hand ways for communicating information or feedback about the models to the learner, and, on the other hand, relies on innovative methods for communicating learned models to a domain expert who is typically inexperienced in machine learning. Thus, we envision future interactive data analytics to essentially consist of four components:

Interactive Data Acquisition: The domain expert and the learning system need to interact to acquire the appropriate data as well as for annotating and labeling the data.

Interactive Model Development: Besides influencing the learning process by providing suitable training data, the domain expert can interact with the learning algorithm during the model's construction and use this to continually alter and refine the model.

Interactive Model Interpretation: The learned model is not passive and intransparent, but can be actively understood and explored by the domain expert.

Interactive System Support: To support the iterative learning process and the effective human–computer interaction under real-time constraints, it is essential to link interactive machine learning with data management systems.

All four components have, to some extent, been explored in the literature before, but for interactive data analytics it is essential that all four are realized and tightly integrated so that their synthesis facilitates the interaction between the domain expert and the analytics tool at multiple levels. Figure 1 shows how the four components enrich the traditional data analytics process based on explicit feedback in the form of labeled data.

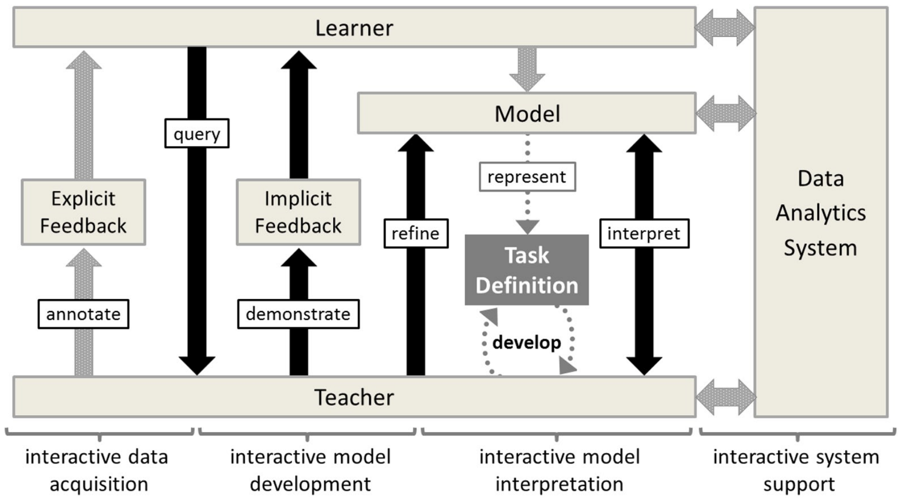

Fig. 1. Overview of our interactive data analytics vision: Besides the traditional approach (gray arrows) of training a learner by explicit feedback in the form of labeled data, the learner should be enabled to actively pose questions to the teacher, integrate implicit feedback and direct changes to the underlying model as well as foster interpretation of the learned model yielding a fluid task definition. The machine learning setup is backed by interactive system support to ensure the learner's response in real-time.

By putting the human in the loop, we can approach data analytics methods also in small data scenarios requiring expert knowledge, as it is the case in the humanities. The interactive learning paradigm does not only allow a user to steer the learning process, but also to simultaneously develop the actual task and learning goal.

It should be noted that the general idea of interactive machine learning is anything but new. De Raedt and Bruynooghe [80] used the term already in 1992 for a logical rule learning system that interactively queries the user whether a newly learned rule is considered correct. Interactive machine learning gained also some attention during the *Intelligent User Interfaces* conference series, since Fails and Olsen [29] introduced an interactive approach for image segmentation in 2003. Unlike in previous works, the users of Fails and Olsen's system are more than oracles for assessing the learning process. Instead, the users roughly crayon the outlines of an object and iteratively refine their input as the system updates its prediction. Or consider the GrabCut system due to Rother et al. [86]. It takes this further and considers different interaction modes to make the communication between the user and the learner more efficient. This is what Amershi et al. [4] later entitled as "power to the people": the users teach a machine learning system by demonstrating how it should behave, rather than just providing a (large) number of hand-labeled training instances.

This learning paradigm is known as *imitation learning* or *learning from demonstration* [88,6,5,77]. Though it is an active research topic at the *Neural Information Processing Systems* (NIPS) conferences, research continues to focus mostly on teaching robots. To facilitate data analytics for the humanities, however, we need to leverage methods for text and visual data, which to date have only been cursorily researched. Despite image segmentation [29,86,37], there is recent work on natural language generation [55] and natural language understanding [104]. The advantages of learning from demonstration are backed up by the user study by Cakmak et al. [15], who found that a standard approach to active learning is not perceived as a "real interaction" and some users complained about an imbalanced and badly structured stream of inquiries that hindered effective teaching to some extent. This was different in a setup that allowed users to ask the robot themselves, as the learning felt more natural.

Moving on with interactive approaches to data analytics for text and visual data is relevant for a large number of tasks in the humanities. A prototypical application is argumentation analysis. This is relevant for communication science (e.g., the analysis of political speeches), philosophy and ethics (e.g., controversial standpoints about cloning), history (e.g., changes in the public debate about corruption), and journalism (e.g., evidence retrieval and fact checking in news items). Typically, these use cases have a high impact on our society, but no clear, predefined task definition. There are, for instance, multiple competing theories of argumentative structures or how to define a fact. Instead, researchers approach a task from different perspectives with research questions that evolve while working with the data. Most tasks would benefit from a multimodal analysis, as text, images, and video (e.g., of political debates or to generate evidence for fact checking) contain complementary information.

In the remainder of the paper, we describe in detail our vision of interactive data analytics for the humanities and how we need to go beyond previous work. In section 3, we introduce an integrated example for a fluid task in argumentation analysis within the social sciences. We cover a wide range of methods and techniques from machine learning, natural language processing, computer vision, and data management systems. Following our four components, we first discuss techniques for interactive data acquisition in section 4. In section 5, we then argue for means of allowing users to directly participate in the model development, for instance, by demonstrating how the learner should behave. This is closely linked to interactively interpreting the learned model in section 6 by tracing a learner's decision and understanding the model internals, allowing the teacher to effectively guide the learning. Finally, we discuss methods of interactive system support in section 7 to deal with real-time constraints and effective visualization of the results. In section 8, we conclude the paper.

3 A Visionary Example: Argumentation Analysis

To demonstrate the potential of the interactive data analytics paradigm, we consider the following integrated example targeted towards argumentation analysis in the humanities. Imagine a social scientist (S) investigating a controversial research question such as "Should Europe accept more refugees?" by using the envisioned interactive data analytics system (D) with the goal of compiling a customized summary of the relevant standpoints and arguments present in arbitrary web sources.

In the first step, S acquires data by advising D to crawl relevant documents and video clips from a newspaper, YouTube, and a number of online forums dedicated to discussing the European Union. Instead of implementing this step as a preprocessing step tweaked by an information retrieval specialist, D would come up quickly with a few first results and asks S to select the ones that best fit her needs. D would pick up this feedback and iteratively refine both the crawling and the relevance ranking of the results. Already while the corpus is growing, S and her team would annotate some of the retrieved documents and individual scenes in the video clips, e.g., for claims and premises. From these annotations, D would develop a machine learning model that cannot only improve the information retrieval and exploration, but also pre-annotate the crawled documents to allow S to correct the system responses rather than having to create all annotations from scratch. Since S is not satisfied with the initial quality of the model, she first asks her colleague for a large annotated dataset of claims and premises and guides the system in transferring knowledge about how to effectively detect claims and premises although the newly crawled data spans totally different genres and domains. In the interplay of responding to system queries and annotating new data, S explores the dataset while it is developing. By observing which documents S prefers and that she tends to skip the first two minutes of a video, D gets a better notion of what is important for the task.

As the data grows and S feels that the claim and premise annotations work much better, she starts to label argumentative fallacies, such as the *shifting the burden of proof* fallacy as in: "There must be thousands of terrorists immigrating to our country. I challenge you to prove me wrong!" To this end, S starts with rather broad fallacy categories and iteratively refines them. D needs to adapt the learned model on-the-fly and assist S, for example, by suggesting a decision boundary between overly large, inhomogeneous groups of fallacious arguments. At several points, S is puzzled why D suggests a certain fallacy type, so she asks the system for explanation. D might respond that the decision is largely influenced by certain parts of a neural network, for which D shows a visualization. S soon finds that the voice recognition component keeps conflating *refugee* and *refuse*. Therefore, she draws an improved decision boundary into the visualization, which yields a strong constraint during retraining of the model.

Having analyzed much of the crawled data, S gets interested in ordering the arguments in a timeline. She asks D to do this by marking where the document or video creation time can be found. She also demonstrates that D should select the most important arguments and place the oldest argument at the top and the newest argument at the bottom. While doing so, she decides that it would make more sense to mark related arguments, which is why she introduces a separate column per argumentative strand.

This integrated example demonstrates that concepts sought after are likely to evolve during the discovery process and, hence, when selecting data and training a system. S starts with practically no data, but generates everything while she develops the research question and the result format. The envisioned system has to be highly flexible and interactive so that S can make all these inputs herself while D responds in real-time—even though the processing of the entire dataset might still be running in the background.

4 Interactive Data Acquisition

In a traditional data analytics setup, a machine learning algorithm is trained with massive amounts of data. This is particularly true in recent approaches making heavy use of deep neural networks [57]. Acquiring such large amounts of data is, however, a key problem in the humanities, where annotations typically depend on expert knowledge. We are thus facing small data scenarios, in which the learner often has no initial data for the current domain or task at all—which is generally known as the *cold-start problem*

To address this problem, new methods are needed to make better use of existing data and obtain new annotations for learning as efficiently as possible. Interactive annotation processes offer the opportunity for learners to request feedback when they are uncertain (active learning) and for human teachers to gradually refine the model while they are annotating (online learning) or intervene when they encounter mistakes. With interactive data acquisition, learning and annotation should become a single intertwined process that is guided by the teacher to rapidly learn a good model.

Research in *active learning* provides a first step into interactive data analytics by transgressing the conventional model of machine learning [20,91]. Active learning has already found frequent use in natural language processing (e.g., [70,3,33]), in particular also for annotating texts [103]. For example, the open-source annotation tool WEBANNO has recently added active learning techniques for suggesting potential annotations to the annotator [108]. In computer vision, active learning has been investigated for object categorization [45], where a human teacher interactively labels images with the corresponding object categories, or for object attributes [74]. This paradigm has also been applied to domains where only experts are able to provide the appropriate fine-grained category information [12].

Typically, active learning techniques focus on identifying examples for which the currently learned hypothesis is most uncertain in its prediction. Using the most unreliable matches of the current hypothesis in the text to query the annotator for more information is the key idea of uncertainty sampling [58], a variant of which has, e.g., been applied to learning statistical grammars [8]. Bayesian active learning is a commonly used technique to globally optimize uncertainty [44]. Many classifiers not only yield a prediction but also a confidence score or probability value indicating the reliability of the prediction. Alternatively, uncertainty can be measured using the disagreement in a committee of diverse classifiers [92]. It may also be good to select batches of examples instead of single ones [13].

The active learning paradigm is particularly suited for dealing with the cold-start problem in the humanities, as it yields very steep learning curves [42]. To date, however, the available active learning methods are severely limited in the types of annotation and learning tasks they may be applied to. For instance, the active deep learning networks proposed by Zhou et al. [110], focus on atomic user annotations (e.g., sentiment labels). In contrast, our humanities setting requires a suitable representation of complex analysis units composed of multiple variables of different types, such as events, claims, or gestures. The representation also has to reduce the burden of the domain experts to express their expertise to the learning system. Experts may have many years of experience, and simply using only data and labels ignores all of the valuable insights that they could offer. As an example, the expert may say that if the author of a short story is “Edgar Allan Poe” then the preferred genre of the story is “mystery”. Thus, users may provide programs that label some subset of the data as proposed by Ratner et al. [82]. This *data programming* is an instance of *statistical relational learning* [26], which learns models in domains such as the humanities with both complex relational structure—variable number of objects of different types with relations among them—and rich probabilistic structure. This generalizes weakly-supervised learning [107] and allows for a seamless integration of different learning systems and knowledge bases.

However, while users are domain experts they are not machine learning experts. Thus, deciding what knowledge to provide to the learner apriori is a difficult problem. Even if users were able to intuitively offer their knowledge, it is impractical for them to completely summarize years of experience before the learning starts. Hence, the learning algorithm should *actively seek advice* from the user as proposed by Odom and Natarajan [69]. For instance, the learner may ask the user, "What is your choice of label if a student and a professor are co-authors?" The expert replies saying, "I prefer the student to be advised by the professor". This preference is then explicitly weighed against the data while continuing to learn the model.

Making use of unlabeled data for training is generally an attractive way of addressing the cold-start problem and small data scenarios. Unsupervised (as well as semi- and weakly-supervised) learning methods incorporate general knowledge in the model design and use this to *extract latent structure* from unlabeled data. This latent structure simplifies the learning problem without relying on annotated training data. Early work aimed at giving recommendations based on sparse data [89]. Similar techniques have been transferred to a number of natural language processing (e.g., semantic analysis [10,102]) and computer vision [84,73] tasks. Unsupervised methods can further accelerate the learning process by identifying structure in the data before training data is available. Bayesian and approximate Bayesian methods, such as variational auto-encoders [52], provide novel techniques for handling this uncertainty within a deep model. The challenge is to identify such suitable models that are general enough to suit the variety of learning tasks that the system must adapt to.

We can also leverage data that has been labeled for different, but related annotation schemes or tasks using *transfer learning* [72]. Daumé III [25] introduced a simple method for domain adaptation using an augmented feature space, and Kim et al. [51] suggest using label embeddings for cases in which the annotation schemes vary substantially. Recent work has introduced new methods for transferring information from different hidden layers in deep neural networks for image representations [71,62]. The potential for transfer learning in expert-based data analytics tasks has not yet been fully explored and typically has not considered the transfer of personalized models between similar people. However, this will be necessary for subjective tasks in which humanities researchers follow alternative hypotheses or analysis strategies (e.g., different argumentation theories). Though these researchers aim at developing a personalized, user-centered model, they can benefit from integrating general latent properties of the task that hold across multiple strategies, or that are highly similar to other users.

Crowdsourcing has been successfully used to generate large amounts of labeled data in both natural language processing and computer vision. Major challenges to applying crowdsourcing for humanities tasks are fluid task definitions and the need for expert knowledge. There have been several attempts at modeling complex annotation tasks as games with a purpose [2] (e.g., for predicting protein structures [21]), but so far there is little work for humanities data. As crowdsourcing is limited to clearly defined tasks

[14], it will be necessary to interactively translate a vaguely defined task to a clear-cut description that is comprehensible for lay workers – or identify subproblems for which this is possible. Recent work approached the task of intelligently selecting a worker's task to optimize annotation cost and quality [105,47,95]. While this improves learning rates, it is also a way to iteratively match a worker's skills with the (latent) demands of the task. Other works have focused on machine learning methods that are suitable for learning in the presence of the high numbers of comparably unreliable labels that often result from crowdsourcing annotations [41].

5 Interactive Model Development

Humanities experts each have their own personal working style and complex, changing goals, yet existing tools typically assume a static model that cannot adapt on-the-fly to the user's needs. These tools do not account for the different steps the user may take to complete a task, and do not learn how best to present information to assist the user at each stage. Furthermore, model development depends entirely on explicit annotations from the user. A new, dynamic approach for interactive model development is needed to adapt to such fluid problem definitions through both explicit and implicit user feedback.

Explicit annotation is time-consuming and typically constrains the user to a single, narrow way for passing information to the model—most often by labeling instances with one of multiple predefined classes. Annotation costs can often be considerably reduced by learning from multiple types of user feedback, including the implicit information in user navigation patterns recorded as mouse clicks. This feedback may not be in the form of class labels that can directly be used to train a model but may instead represent a choice for one action over another. For example, a user clicking on an item in a list may be interpreted as a preference for that item over the other items in the list [79]. Developing a ranking model from such pairwise comparisons is the goal of *preference learning* [31]. For example, Dzyuba et al. [27] infer a general ranking function for patterns from user-provided feedback over a small set of patterns. Training preferences for such models can be implicitly inferred from the user's behavior [79].

A further complication is the inconsistency of such implicit feedback signals, which are likely to have varying levels of noise or bias over time. Bayesian techniques have been successfully used to handle such unreliability when learning from pairwise preferences [18] or combining crowdsourced classifications with labelers whose behavior changes over time [96]. Such techniques could be used to train models for analyzing language or image data with multiple types of user feedback, and can be integrated using variational inference [7], which allows composition of models in a modular fashion. Recent works on deep exponential families [81] and variational auto encoders [52] show how this idea can be executed to create deep models using approximate Bayesian methods for complex modeling tasks.

As well as tailoring models of data to individual users, an interactive approach to humanities tasks could adapt the way that models are used to select and present information to assist users with different steps in a complex analysis task. Depending on the end user's perspective, the way that model outputs are presented may have very different costs or benefits. For example, omitting a crucial piece of information from a summary of an argument may have higher cost than including redundant text. The learner may also request feedback explicitly, but the future benefits of learning from this information must be traded off against the time the user takes to provide it and the need to provide her with immediate benefits. This balance is known as the *exploration vs. exploitation trade-off*, and can be optimized using reinforcement learning (RL) techniques [99]. To apply RL, we view our interactive scenario as a partially observable Markov decision process (POMDP), in which the agent aims to maximize a cumulative future reward by choosing an action given the current state of its environment. The state includes the available text and visual data, latent structures inferred from that data, such as arguments, as well as user behavior data (e.g., a record of clicks) and latent variables representing the user's preferences and task. The agent can perform different actions such as choosing information to present to the user, how this should be presented (e.g., the order of a list), and explicitly requesting feedback. The reward indicates the value of the new state to the user but may not be provided explicitly, so may need to be inferred from implicit feedback. The task of the learning system is to learn a so-called policy that lets it choose its actions in a way that maximizes the expected reward. A successful approach for complex tasks with large state spaces are relational RL [56] and deep RL [66], but this may require a large number of steps to train. In practice, the agent may encounter previously unseen states, which can be handled effectively using Bayesian RL to account for the uncertainty in the best course of action [34].

A crucial issue for interactive model development is that the learning system develops a model of the user and her task, so that techniques such as RL can effectively reduce the amount of interaction required. In machine learning, several techniques have recently been developed to facilitate what is also known as *apprenticeship learning*. Such techniques can be employed to learn human skills that the experts cannot directly communicate, or to personalize interaction processes.

Most notably, variants of RL have been developed that do not aim at optimizing a system's behavior by trial and error given a numeric feedback signal, but instead try to mimic observed behavior. The corresponding field—*learning from demonstration*—has become particularly successful in robotics [88,6,5] but has not yet been popularized in applications in the humanities. *Inverse reinforcement learning* [1,68,67,77] is such a technique, where the goal is to learn a hidden reward function, which may guide the teacher's observed behavior.

When working with a fluid problem definition, the relevance of previously accumulated data varies over time depending on the user's current task. Models must adapt to this *concept drift*, for example, by employing Bayesian techniques to handle the uncertainty caused by changing user behavior [96]. The long-term value of data must also be taken into account when acquiring explicit feedback from a user. Techniques for *lifelong machine learning* [19] could be brought to bear on this problem by providing mechanisms for balancing long-term value against the cost of interrupting a user [46].

A necessity for an interactive collaboration between a data analytics system and a human domain expert is that the machine learning algorithm does not need to be configured by a data science expert. This includes problems such as the automated selection of an appropriate algorithm [60] and tuning its hyperparameters [40]. For example, AUTO-WEKA is an extension of the WEKA data mining and machine learning library, which can automatically find an appropriate configuration for a given learning task [101]. Humanities experts could also be empowered to directly modify the model, for example, by defining logical rules that alter neural networks models [39]. Another relatively unexplored approach would be to allow users to provide feedback to an *attention mechanism*, which directs a neural network to focus on the relevant parts of an image [24] or piece of text [63]. Given suitable user interfaces, humanities users could also create new features on the fly that provide useful abstractions from raw data. Intuitive latent features, such as topic clusters, could be modified directly by users, for example, by moving items between clusters. However, extensive modification of the internal components of a model depends on suitable techniques for interpreting models, which we discuss in the next section.

6 Interactive Model Interpretation

Most existing machine learning algorithms are integrated into applications as a black box that presents users with one type of output, such as classifications, without exposing them to the underlying workings of the method. However, sophisticated models for vision or language understanding often have multiple components, such as the different layers in a deep network, or multiple algorithms used together in a pipeline. While users should not be expected to understand the details of how a method works, complete opaqueness can undermine a user's confidence in an algorithm because its mistakes become unpredictable, meaning that the user may spend more time checking the automated method's work. An algorithm can indicate confidence in its decisions through probabilities, but these do not provide an explanation of the decision and therefore do not solve the problem. Understanding the model or important parts of it can be crucial when the algorithm encounters a new domain and must transition from a state of ignorance to earn the user's trust in carrying out its intended task. Moreover, in many applications the goal is not so much to maximize predictive performance, but to gain insight into the data. For this reason, one commonly distinguishes between *predictive* and *descriptive data mining*.

Algorithms for descriptive data mining typically rely on a rule-based representation of the results because rules offer the best trade-off between human and machine understandability [30]. Nevertheless, the aspect of interpretability still needs to be further explored. For example, it is conventional wisdom in machine learning that shorter explanations are better. Occam's Razor, *"Entia non sunt multiplicanda sine necessitate"*,[1] is often cited as support for this principle. Typically, it is understood as *"given two explanations of the data, all other things being equal, the simpler explanation is preferable"*. However, there are a few rule learning algorithms that explicitly aim for longer rules, and it is not clear that shorter rules are indeed more comprehensible for human experts [97]. Other criteria, such as semantic coherence of the conditions of a rule, should thus be considered in the learning process [32].

For other types of learning algorithms, it is harder but often nevertheless crucial to be able to explain and justify the outputs of the learned model to the user. For example,

[1] Entities should not be multiplied beyond necessity.

the strength of many recent learning algorithms, most notably deep learning [57,90], word embeddings [64] or topic modeling [11], is that latent variables are formed during the learning process. Understanding the meaning of these hidden variables is crucial for transparent and justifiable decisions. Consequently, visualization of such model components has recently received some attention [17,109,85]. Several works addressed the visualization of a network's long short-term memory (LSTM) and attention mechanisms, e.g., in machine translation [87]. Hendricks et al. [38] identify discriminating features in a deep visual classification task and learn to associate explanations in natural language with such features. Learning human-readable explanations at an appropriate level of abstraction is a core open research question. In any case, the need for learning interpretable models has been identified in several disciplines, and, not surprisingly, workshops at various conferences have been devoted to this topic [49,106,35].

A common restriction for most of the above methods is that even though the explanation quality is lifted from a technical to a semantic level, the user is still a comparably passive consumer of the presented models. A key step forward would be if users could directly interact with the provided models, visualize them from different (semantic) angles, pose multiple question types to the data, and, eventually, even correct the models. Systems like MININGZINC [36] and relational mathematical programming frameworks [48,65], which allow users to declaratively define a data analytics problem with a high-level constraint-based language, are currently being developed.

A further step ahead would be to allow the user to directly modify parts of the model in interaction with the learner. For example, Beckerle [9] explored an interactive rule learning process where learned rules could be directly modified by the user, thereby causing the learner to re-learn subsequently learned rules. Alternatively, one can imagine a user who is able to directly interact with other types of model components, such as hidden layers in a neural network. For example, Hu et al. [39] proposed a combination of deep neural models with structured logic rules to foster the interpretation of the model and allow users to (indirectly) steer the learning process. Such an interactive approach is perhaps closer to the way that people train each other – by explaining how they make decisions as well as providing examples. This approach could therefore improve both the user's trust in the model and increase training speed for new tasks and domains by reducing the need to provide numerous examples before the important features are identified. Research towards such truly interactive machine learning systems has just started.

7 Interactive System Support

Many systems and tools exist already to support developers when curating complex machine learning models for text and image data (e.g., R and Spark MLLib or TensorFlow). However, these tools are limited in a number of fundamental ways in their support for a human-in-the-loop when curating or developing models. First, existing tools require well-trained data scientists to select the appropriate techniques and adjust the hyperparameters to build models and to evaluate their outcomes. Second, even when working on small labeled datasets, many of these tools still require large amounts of data as background knowledge (e.g., large knowledge bases, corpora, pre-trained embeddings) and thus are often too slow to provide interactive feedback to domain experts in the model development process. Third, many of the machine learning techniques require heavy data pre-processing steps before text and visual data can be used for the actual analytics task. This can further limit the overall interactivity of the system.

In this section, we discuss how existing machine learning tools have to change to better support interactive data analytics from a systems perspective. This systems perspective complements the interactive data acquisition, model development, and model interpretation components described in the previous sections.

Interactive Data Acquisition: The first step in data acquisition is typically the pre-processing of the text and visual data used as input or background knowledge. One important step when pre-processing text documents is, for example, to retrieve the embeddings for each word from an existing corpus of pre-trained word vectors and then apply the pre-trained model (e.g., for classifying arguments as supportive or not). However, corpora of word embeddings can be huge (e.g., billions of vectors in the case of [76]). Therefore, we require efficient techniques for storing and retrieving embeddings or similar input data. Furthermore, pre-processing steps must be able to be applied incrementally to new data sources to support a progressive execution of the upstream machine learning pipeline. In the computational argumentation example, the user does not first want to pre-process the complete set of documents before applying it to the classification model to find out which arguments support her hypothesis. Instead, pre-processing and classification should be intertwined to provide progressive answers to the user while streaming over the text of the document.

The already mentioned cold-start problem is another challenge for system support to data acquisition. While there exist many techniques in active learning, none of these techniques focuses on how to suggest new examples at interactive speeds. Instead, their main objective is to find examples where the current model is most uncertain in its prediction, since labeling these items promises the biggest benefit. However, finding the most uncertain examples can be extremely expensive if large amounts of data are involved. One idea to achieve interactivity in this learning process is to use ideas from neighbor-sensitive hashing [75] to quickly find the k-nearest unlabeled neighbors of already labeled examples that are close to the decision boundary (i.e., where the model is the most uncertain).

For solutions based on transfer learning, we require systems that are able to store a large number of datasets and models and provide efficient search capabilities that allow users to interactively retrieve related data or pre-trained models that are best suited for their task and data at hand.

Interactive Model Development: Throughout the overall model training and development process, data analytics tools must consistently provide response times low enough to guarantee fluid user interactions and integrate user feedback. In fact, a recent study [61] has shown that even small delays of more than 500 ms significantly decrease a user's activity level, dataset coverage, and insight discovery rate. However, none of the existing tools can guarantee interactive latencies [53]. Previous work by Crotty et al. [23] approached this problem for structured data, but we are not aware of any work focusing on unstructured data or use cases in the humanities.

One important challenge for incorporating user feedback is to enable the retraining of models in real-time when new user input is available either through explicit labeling or through implicit feedback. While there has been significant work in online machine learning that allows models to be progressively updated, existing techniques often cannot be applied directly, as the incremental retraining might not be able to keep up with the high update rate resulting from implicit feedback (e.g., clickstreams). Another issue is that models may become too complex and thus updating the model incrementally might exceed the interactive threshold. There are multiple ways to tackle this issue: adaptively batching updates based on the incoming update rate, employing parallelization [83] or online learning algorithms such as MIRA [22], and avoiding that the model forgets already learned concepts, e.g., by applying experience replay [59] to speed up an RL-based learner.

Since implicit feedback with its heavily varying quality is of a different nature to explicit feedback, it might turn out useful to adaptively drop low quality updates by applying techniques known in the systems community for load shedding [100]. However, these techniques must be adapted to perform load shedding based on the quality of the user input. Another interesting direction to address this problem is to approximate the model (e.g., by representing weights in a neural network in an approximate manner). Finding the right approximation to achieve the best model quality under a fixed time budget to apply the update appears a promising avenue of research.

To make interactive learning tools accessible to humanities experts, we discussed automatically selecting machine learning algorithms and their hyperparameters in section 5. However, this is a challenging task due to the large search space, which means that naïve grid search approaches, for instance, do not allow us to compute a good set of hyperparameters in real time. Recent techniques such as scalable kernel composition [50] provide viable solutions for certain types of machine learning models, but further research is required to enable automated model selection in real time.

Interactive Model Interpretation: Interactive systems support is also required for many of the model interpretation tasks outlined in section 6. The reason is that models, such as neural networks, can get large and thus exploring the model and summarizing important aspects at interactive speeds is a challenge on its own. Furthermore, as a result of model interpretation, users might want to manually adjust the model internals. These techniques are sometimes referred to as Specialized Programming [78]. That is, the user "programs" the machine learning model, for instance, by directly modifying internal layers of a neural network. The corresponding challenge from the systems perspective is again to retrain the model based on the user modifications and interactively allow the user to inspect the model quality after updating the model.

Generally, if interactive systems are to have truly broad impact, building and maintaining them needs to become substantially easier. They should support the rapid combination, deployment, and maintenance of existing data analytics algorithms and domain knowledge. For that, one should identify and validate programming and data abstractions as building blocks. Identifying, optimizing, and supporting abstractions as primitives could make systems for interactive data analytics substantially easier to setup, to understand the user, and to scale. This can bring us a step closer to unleashing the full potential of data analytics in various domains, even beyond humanities. To ensure that such a platform is accessible to many users, the programming interface must be small, clean, and composable to enhance productivity and enable users to try and accommodate many data analytics algorithms; the ability to integrate diverse data sources and types requires the data model of the programming interface to be versatile. A combination of the relational data model and a statistical relational language such as Markov logic and relational mathematical programs [26] satisfies these criteria. In combination with imperative languages this seems to be a promising direction but further research is required to realize interactiveness.

8 Conclusions

Current data analytics is mostly limited to tasks for which large-scale homogeneous benchmarks exist. However, real-world use cases typically have different properties: they are highly heterogeneous and involve infrequent and complex phenomena, for which only small-scale datasets are available. The need for new data analytics approaches is particularly pressing for the humanities, as the use cases in these disciplines do not only require background expert knowledge for developing a system, labeling data, and interpreting the results, but are also only vaguely defined in advance. Such a fluid problem definition – which a human expert develops while doing the actual task – calls for a totally different approach to data analytics.

In this paper, we have laid out our vision towards future data analytics in the humanities based on interactive machine learning. The main idea is to put the human in the loop and iteratively refine the model based on the user's feedback. By focusing on the (expert) user and her task, we need to think beyond natural language processing and closely cooperate with computer vision to enable multimodal systems to learn jointly from text and visual data and mutually benefit from recent advances in the research of suitable (deep) machine learning architectures. Interactive data analytics also requires core research in machine learning, since existing techniques almost exclusively learn from indirect input in the form of labeled examples or an algorithm's parameter settings. Instead, our vision is that a human expert can steer the learning process by using the system for her task, demonstrating how the system should behave and interpret the learned model in order to identify specific patterns or errors. Even though there is little labeled data, we will have to include large amounts of background knowledge and computationally heavy learning algorithms, which must be able to return their estimations in real-time. Research into efficient data management and systems engineering is therefore the fourth major pillar of our vision.

In all four fields of study, there is already a vast body of existing work with which we can fulfill parts of our vision. However, there is a clear demand for future efforts to close the gaps in interactive model development and interpretation as well as systems supporting this approach. If natural language processing joins forces with computer vision, machine learning, and data management systems, we can make a great leap forward.

References

1. Abbeel, P., Ng, A.Y.: Apprenticeship Learning via Inverse Reinforcement Learning. In: 21st International Conference on Machine learning (ICML). ACM, New York (2004)
2. von Ahn, L.: Games With A Purpose. Computer 39(6), 96–98 (2006)
3. Ambati, V., Vogel, S., Carbonell, J.G.: Active Learning-Based Elicitation for Semi-Supervised Word Alignment. In: 48th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 365–370. ACL, Stroudsburg (2010)
4. Amershi, S., Cakmak, M., Knox, W.B., Kulesza, T.: Power to the People: The Role of Humans in Interactive Machine Learning. AI Mag 35(4), 105–120 (2014)
5. Argall, B.D., Chernova, S., Veloso, M., Browning, B.: A survey of robot learning from demonstration. Robot Auto Syst 57(5), 469–483 (2009)
6. Atkeson, C.G., Schaal, S.: Robot learning from demonstration. In: 14th International Conference on Machine Learning (ICML). pp. 12–20. Morgan Kaufmann, San Francisco (1997)
7. Attias, H.: A Variational Bayesian Framework for Graphical Models. In: Advances in Neural Information Processing Systems 12 (NIPS). pp. 209–215. MIT Press, Cambridge (2000)
8. Becker, M., Osborne, M.: A Two-Stage Method for Active Learning of Statistical Grammars. In: 19th International Joint Conference on Artificial Intelligence (IJCAI). pp. 991–996. Morgan Kaufmann, San Francisco (2005)
9. Beckerle, M.: Interaktives Regellernen [in German]. Diploma thesis, Technische Universität Darmstadt (2009)
10. Bejan, C.A., Harabagiu, S.: Unsupervised event coreference resolution. Comput Linguist 40(2), 311–347 (2014)
11. Blei, D.M.: Probabilistic topic models. Commun ACM 55(4), 77–84 (2012)
12. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual Recognition with Humans in the Loop. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV, LNCS, vol. 6314, pp. 438–451. Springer, Berlin, Heidelberg (2010)
13. Brinker, K.: Active Learning of Label Ranking Functions. In: 21st International Conference on Machine Learning (ICML). pp. 129–136. ACM, New York (2004)
14. Burger-Helmchen, T., Pénin, J.: The limits of crowdsourcing inventive activities: What do transaction cost theory and the evolutionary theories of the firm teach us? In: Proceedings of the Workshop on Open Source Innovation. pp. 1–26. Strasbourg, France (2010)
15. Cakmak, M., Chao, C., Thomaz, A.L.: Designing Interactions for Robot Active Learners. IEEE T Auton Ment De 2(2), 108–118 (2010)
16. Chambers, R.A., Michie, D.: Man-machine co-operation on a learning task. In: Parslow, R.D., Prowse, R., Elliott-Green, R. (eds.) Computer Graphics: Techniques and Applications, pp. 179–185. Plenum, London (1969)
17. Chaney, A.J., Blei, D.M.: Visualizing Topic Models. In: 6th International Conference on Weblogs and Social Media (ICWSM). AAAI Press, Palo Alto (2012)
18. Chen, X., Bennett, P.N., Collins-Thompson, K., Horvitz, E.: Pairwise ranking aggregation in a crowdsourced setting. In: 6th ACM International Conference on Web Search and Data Mining (WSDM). pp. 193–202. ACM, New York (2013)
19. Chen, Z., Liu, B.: Lifelong Machine Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool, San Rafael (2016)
20. Cohn, D.A., Atlas, L., Ladner, R.: Improving Generalization with Active Learning. Mach Learn 15(2), 201–221 (1994)
21. Cooper, S., Khatib, F., Treuille, A., Barbero, J., Lee, J., Beenen, M., Leaver-Fay, A., Baker, D., Popovi, Z., Foldit Players: Predicting protein structures with a multiplayer online game. Nature 466, 756–760 (2010)

22. Crammer, K., Singer, Y.: Ultraconservative online algorithms for multiclass problems. J Mach Learn Res 3, 951–991 (2003)
23. Crotty, A., Galakatos, A., Zgraggen, E., Binnig, C., Kraska, T.: The case for interactive data exploration accelerators (IDEAs). In: Workshop on Human-In-the-Loop Data Analytics (HILDA@SIGMOD). p. 11. ACM, New York (2016)
24. Das, A., Agrawal, H., Zitnick, L., Parikh, D., Batra, D.: Human Attention in Visual Question Answering: Do Humans and Deep Networks look at the same regions? In: 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 932–937. ACL, Stroudsburg (2016)
25. Daumé III, H.: Frustratingly Easy Domain Adaptation. In: 45th Annual Meeting of the Association of Computational Linguistics (ACL). pp. 256–263. ACL, Stroudsburg (2007)
26. De Raedt, L., Kersting, K., Natarajan, S., Poole, D.: Statistical Relational Artificial Intelligence: Logic, Probability, and Computation. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool, San Rafael (2016)
27. Dzyuba, V., van Leeuwen, M., Nijssen, S., De Raedt, L.: Interactive Learning of Pattern Rankings. Int J Artif Intell T 23(6) (2014)
28. Elkahky, A.M., Song, Y., He, X.: A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems. In: 24th International Conference on World Wide Web (WWW). pp. 278–288. International World Wide Web Conferences Steering Committee, Geneva (2015)
29. Fails, J.A., Olsen, Jr., D.R.: Interactive Machine Learning. In: 8th International Conference on Intelligent User Interfaces (IUI). pp. 39–45. ACM, New York (2003)
30. Fürnkranz, J., Gamberger, D., Lavrač, N.: Foundations of Rule Learning. Springer, Berlin, Heidelberg (2012)
31. Fürnkranz, J., Hüllermeier, E. (eds.): Preference Learning. Springer, Berlin, Heidelberg (2010)
32. Gabriel, A., Paulheim, H., Janssen, F.: Learning Semantically Coherent Rules. In: Cellier, P., Charnois, T., Hotho, A., Matwin, S., Moens, M.F., Toussaint, Y. (eds.) 1st International Workshop on Interactions between Data Mining and Natural Language Processing. CEUR Workshop Proceedings, vol. 1202, pp. 49–63 (2014)
33. Gambäck, B., Olsson, F., Täckström, O.: Active Learning for Dialogue Act Classification. In: 12th Annual Conference of the International Speech Communication Association (INTERSPEECH). pp. 1329–1332. International Speech Communication Association, Baixas (2011)
34. Ghavamzadeh, M., Engel, Y., Valko, M.: Bayesian policy gradient and actor-critic algorithms. J Mach Learn Res 17, 1–53 (2016)
35. Gillies, M., Fiebrink, R., Tanaka, A., Garcia, J., Bevilacqua, F., Héloir, A., Nunnari, F., Mackay, W.E., Amershi, S., Lee, B., D'Alessandro, N., Tilmanne, J., Kulesza, T., Caramiaux, B.: Human-Centered Machine Learning. In: CHI Conference on Human Factors in Computing Systems. pp. 3558–3565. ACM, New York (2016)
36. Guns, T., Dries, A., Nijssen, S., Tack, G., De Raedt, L.: MiningZinc: A declarative framework for constraint-based mining. Artif Intell 244, 6–29 (2017)
37. He, H., Daumé III, H., Eisner, J.: Imitation Learning by Coaching. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25 (NIPS). pp. 3149–3157. Curran Associates, Red Hook (2012)
38. Hendricks, L.A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., Darrell, T.: Generating Visual Explanations. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV. LNCS, vol. 9908, pp. 3–19. Springer, Cham (2016)
39. Hu, Z., Ma, X., Liu, Z., Hovy, E., Xing, E.: Harnessing Deep Neural Networks with Logic Rules. In: 54th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 2410–2420. ACL, Stroudsburg (2016)

40. Hutter, F., Lücke, J., Schmidt-Thieme, L.: Beyond Manual Tuning of Hyperparameters. Künstl Intell 29(4), 329–337 (2015)
41. Ipeirotis, P.G., Provost, F.J., Sheng, V.S., Wang, J.: Repeated labeling using multiple noisy labelers. Data Min Knowl Disc 28(2), 402–441 (2014)
42. Jamieson, K.G., Jain, L., Fernandez, C., Glattard, N.J., Nowak, R.: NEXT: A System for Real-world Development, Evaluation, and Application of Active Learning. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28 (NIPS). pp. 2638–2646 (2015)
43. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender Systems: An Introduction. Cambridge University Press, Cambridge (2010)
44. Kandasamy, K., Schneider, J., Poczos, B.: Bayesian Active Learning for Posterior Estimation. In: 24th International Joint Conference on Artificial Intelligence (IJCAI). pp. 3605–3611. AAAI Press, Menlo Park (2015)
45. Kapoor, A., Grauman, K., Urtasun, R., Darrell, T.: Active learning with Gaussian processes for object categorization. In: 11th International Conference on Computer Vision (ICCV). pp. 1–8. IEEE, IEEE, New York (2007)
46. Kapoor, A., Horvitz, E.: Principles of lifelong learning for predictive user modeling. In: User Modeling, LNCS, vol. 4511, pp. 37–46. Springer, Berlin, Heidelberg (2007)
47. Karger, D.R., Oh, S., Shah, D.: Iterative Learning for Reliable Crowdsourcing Systems. In: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 24 (NIPS). pp. 1953–1961. Curran Associates, Red Hook (2011)
48. Kersting, K., Mladenov, M., Tokmakov, P.: Relational Linear Programming. Artif Intell 244, 188–216 (2017)
49. Kim, B., Malioutov, D., Varshney, K. (eds.): Proceedings of the ICML-16 Workshop on Human Interpretability in Machine Learning. New York (2016), `https://sites.google.com/site/2016whi/`
50. Kim, H., Teh, Y.W.: Scalable structure discovery in regression using gaussian processes. In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) 2016 Workshop on Automatic Machine Learning. JMLR Workshop and Conference Proceedings, vol. 64, pp. 31–40 (2016)
51. Kim, Y.B., Stratos, K., Sarikaya, R., Jeong, M.: New Transfer Learning Techniques for Disparate Label Sets. In: 53rd Annual Meeting of the Association for Computational Linguistics and 7th International Joint Conference on Natural Language Processing (ACL/IJCNLP). pp. 473–482. ACL, Stroudsburg (2015)
52. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: Proceedings of the International Conference on Learning Representations (ICLR). arXiv:1312.6114, Banff, AB, Canada (2014), `https://arxiv.org/abs/1312.6114`
53. Kraska, T., Talwalkar, A., Duchi, J.C., Griffith, R., Franklin, M.J., Jordan, M.I.: MLbase: A Distributed Machine-learning System. In: 6th Biennial Conference on Innovative Data Systems Research (CIDR) (2013)
54. Kucherbaev, P., Daniel, F., Tranquillini, S., Marchese, M.: Crowdsourcing Processes: A Survey of Approaches and Opportunities. IEEE Internet Comput 20(2), 50–56 (2016)
55. Lampouras, G., Vlachos, A.: Imitation learning for language generation from unaligned data. In: 26th International Conference on Computational Linguistics (COLING). pp. 1101–1112. The COLING 2016 Organizing Committee, Osaka, Japan (2016)
56. Lang, T., Toussaint, M., Kersting, K.: Exploration in relational domains for model-based reinforcement learning. J Mach Learn Res 13, 3725–3768 (2012)
57. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521(7553), 436–444 (2015)
58. Lewis, D.D., Gale, W.: A sequential algorithm for training text classifiers. In: 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). pp. 3–12. Springer, London (1994)

59. Lin, L.J.: Self-improving reactive agents based on reinforcement learning, planning and teaching. Mach Learn 8(3), 293–321 (1992)
60. Lindauer, M.T., Hoos, H.H., Hutter, F., Schaub, T.: AutoFolio: An Automatically Configured Algorithm Selector. J Artif Intell Res 53, 745–778 (2015)
61. Liu, Z., Heer, J.: The Effects of Interactive Latency on Exploratory Visual Analysis. IEEE T Vis Comput Gr 20(12), 2122–2131 (2014)
62. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. In: Bach, F., Blei, D. (eds.) 32nd International Conference on Machine Learning (ICML). JMLR: Workshop and Conference Proceedings, vol. 37, pp. 97–105 (2015)
63. Lu, J., Yang, J., Batra, D., Parikh, D.: Hierarchical question-image co-attention for visual question answering. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances In Neural Information Processing Systems 29 (NIPS). pp. 289–297 (2016)
64. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) Advances in Neural Information Processing Systems 26 (NIPS). pp. 3111–3119 (2013)
65. Mladenov, M., Kleinhans, L., Kersting, K.: Lifted inference for convex quadratic programs. In: 31st AAAI Conference on Artificial Intelligence (AAAI). pp. 2350–2356. AAAI Press, Palo Alto (2017)
66. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature 518(7540), 529–533 (2015)
67. Natarajan, S., Joshi, S., Tadepalli, P., Kersting, K., Shavlik, J.: Imitation Learning in Relational Domains: A Functional-Gradient Boosting Approach. In: 22nd International Joint Conference on Artificial Intelligence (IJCAI). pp. 1414–1420. AAAI Press, Menlo Park (2011)
68. Ng, A.Y., Russell, S.J.: Algorithms for inverse reinforcement learning. In: Langley, P. (ed.) 17th International Conference on Machine Learning (ICML). pp. 663–670. Morgan Kaufmann, San Francisco (2000)
69. Odom, P., Natarajan, S.: Actively interacting with experts: A probabilistic logic approach. In: ECML PKDD 2016, Lecture Notes in Computer Science, vol. 9852, pp. 527–542. Springer, Cham (2016)
70. Olsson, F.: A literature survey of active machine learning in the context of natural language processing. SICS Technical Report T2009:06, Swedish Institute of Computer Science (2009)
71. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1717–1724. IEEE, New York (2014)
72. Pan, S.J., Yang, Q.: A Survey on Transfer Learning. IEEE T Knowl Data En 22(10), 1345–1359 (2010)
73. Papandreou, G., Chen, L.C., Murphy, K., Yuille, A.L.: Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: International Conference on Computer Vision (ICCV). pp. 1742–1750. IEEE, New York (2015)
74. Parikh, D., Grauman, K.: Interactively building a discriminative vocabulary of nameable attributes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1681–1688. IEEE, New York (2011)
75. Park, Y., Cafarella, M.J., Mozafari, B.: Neighbor-Sensitive Hashing. Proceedings VLDB Endowment 9(3), 144–155 (2015)

76. Pennington, J., Socher, R., Manning, C.D.: GloVe: Global Vectors for Word Representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543. ACL, Stroudsburg (2014)
77. Piot, B., Geist, M., Pietquin, O.: Bridging the Gap Between Imitation Learning and Inverse Reinforcement Learning. IEEE T Neural Networ PP, 1–13 (2016)
78. Porter, R., Theiler, J., Hush, D.: Interactive Machine Learning in Data Exploitation. Comput Sci Eng 15(5), 12–20 (2013)
79. Radlinski, F., Joachims, T.: Query Chains: Learning to Rank from Implicit Feedback. In: 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 239–248. ACM, New York (2005)
80. de Raedt, L., Bruynooghe, M.: Interactive Concept-Learning and Constructive Induction by Analogy. Mach Learn 8(2), 107–150 (1992)
81. Ranganath, R., Tang, L., Charlin, L., Blei, D.M.: Deep exponential families. In: Lebanon, G., Vishwanathan, S. (eds.) 18th International Conference on Artificial Intelligence and Statistics (AISTATS). JMLR Workshop and Conference Proceedings, vol. 38, pp. 762–771 (2015)
82. Ratner, A., De Sa, C., Wu, S., Selsam, D., Re, C.: Data Programming: Creating Large Training Sets, Quickly. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29 (NIPS). pp. 3567–3575 (2016)
83. Recht, B., Ré, C., Wright, S.J., Niu, F.: Hogwild: A Lock-Free Approach to Parallelizing Stochastic Gradient Descent. In: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 24 (NIPS). pp. 693–701. Curran Associates, Red Hook (2011)
84. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-Supervised Self-Training of Object Detection Models. In: 7th IEEE Workshops on Application of Computer Vision (WACV). pp. 29–36. IEEE, New York (2005)
85. Rothe, S., Schütze, H.: Word embedding calculus in meaningful ultradense subspaces. In: 54th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 512–517. ACL, Stroudsburg (2016)
86. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": Interactive Foreground Extraction Using Iterated Graph Cuts. ACM Trans Graph 23(3), 309–314 (2004)
87. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 379–389. ACL, Stroudsburg (2015)
88. Schaal, S.: Learning from Demonstration. In: Jordan, M.I., Petsche, T. (eds.) Advances in Neural Information Processing Systems 9 (NIPS). pp. 1040–1046. MIT Press, Cambridge (1997)
89. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and Metrics for Cold-start Recommendations. In: 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 253–260. ACM, New York (2002)
90. Schmidhuber, J.: Deep learning in neural networks: An overview. Neural Networks 61, 85–117 (2015)
91. Settles, B.: Active Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool, San Rafael (2012)
92. Seung, H.S., Opper, M., Sompolinsky, H.: Query by Committee. In: 5th Annual ACM Workshop on Computational Learning Theory (COLT). pp. 287–294. ACM, New York (1992)
93. Shaikhina, T., Lowe, D., Daga, S., Briggs, D., Higgins, R., Khovanova, N.: Machine Learning for Predictive Modelling based on Small Data in Biomedical Engineering. IFAC-PapersOnLine 48(20), 469–474 (2015)

94. Shivaswamy, P., Joachims, T.: Coactive learning. J Artif Intell Res 53, 1–40 (2015)
95. Simpson, E., Roberts, S.: Bayesian methods for intelligent task assignment in crowdsourcing systems. In: Decision Making: Uncertainty, Imperfection, Deliberation and Scalability, Studies in Computational Intelligence, vol. 538, pp. 1–32. Springer, Cham (2015)
96. Simpson, E., Roberts, S., Psorakis, I., Smith, A.: Dynamic Bayesian Combination of Multiple Imperfect Classifiers. In: Decision Making and Imperfection, Studies in Computational Intelligence, vol. 474, pp. 1–35. Springer, Berlin, Heidelberg (2013)
97. Stecher, J., Janssen, F., Fürnkranz, J.: Shorter rules are better, aren't they? In: Calders, T., Ceci, M., Malerba, D. (eds.) Discovery Science, LNCS, vol. 9956, pp. 279–294. Springer, Cham (2016)
98. Subramanian, K., Amor, H.B., Isbell, C.L., Thomaz, A.L. (eds.): Proceedings of the IJCAI 2016 Workshop on Interactive Machine Learning: Connecting Humans and Machines. New York (2016), https://sites.google.com/site/ijcai2016iml/
99. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
100. Tatbul, N.: Load shedding. In: Liu, L., Özsu, M.T. (eds.) Encyclopedia of Database Systems, pp. 1632–1636. Springer, New York (2009)
101. Thornton, C., Hutter, F., Hoos, H.H., Leyton-Brown, K.: Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms. In: 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 847–855. ACM, New York (2013)
102. Titov, I., Khoddam, E.: Unsupervised Induction of Semantic Roles within a Reconstruction-Error Minimization Framework. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT). pp. 1–10. ACL, Stroudsburg (2015)
103. Tomanek, K., Olsson, F.: A web survey on the use of active learning to support annotation of text data. In: NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing. pp. 45–48. ACL, Stroudsburg (2009)
104. Wang, S.I., Liang, P., Manning, C.D.: Learning Language Games through Interaction. In: 54th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 2368–2378. ACL, Stroudsburg (2016)
105. Welinder, P., Branson, S., Belongie, S., Perona, P.: The Multidimensional Wisdom of Crowds. In: 23rd International Conference on Neural Information Processing Systems (NIPS). pp. 2424–2432. Curran Associates, Red Hook (2010)
106. Wilson, A.G., Kim, B., Herland, W. (eds.): Proceedings of the NIPS-16 Workshop on Interpretable Machine Learning for Complex Systems. Barcelona, Spain (2016), https://sites.google.com/site/nips2016interpretml/
107. Yang, Z., Cohen, W., Salakhutdinov, R.: Revisiting semi-supervised learning with graph embeddings. In: Balcan, M.F., Weinberger, K.Q. (eds.) 33rd International Conference on Machine Learning (ICML). JMLR: Workshop and Conference Proceedings, vol. 48, pp. 40–48 (2016)
108. Yimam, S.M., Biemann, C., Eckart de Castilho, R., Gurevych, I.: Automatic Annotation Suggestions and Custom Annotation Layers in WebAnno. In: 52nd Annual Meeting of the Association for Computational Linguistics (ACL): System Demonstrations. pp. 91–96. ACL, Stroudsburg (2014)
109. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV, LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014)
110. Zhou, S., Chen, Q., Wang, X.: Active Deep Networks for Semi-supervised Sentiment Classification. In: 23rd International Conference on Computational Linguistics (COLING). pp. 1515–1523. Tsinghua University Press, Beijing (2010)