

Next challenges for adaptive learning systems

Indre Zliobaite¹
Bournemouth University, UK
izliobaite@bournemouth.ac.uk

Bogdan Gabrys
Bournemouth University, UK
bgabrys@bournemouth.ac.uk

Albert Bifet
University of Waikato,
New Zealand
abifet@cs.waikato.ac.nz

Joao Gama
University of Porto,
Portugal
jgama@fep.up.pt

Katarzyna Musial
King's College London, UK
katarzyna.musial@kcl.ac.uk

Mohamed Gaber
University of Portsmouth, UK
mohamed.gaber@port.ac.uk

Leandro Minku
University of Birmingham, UK
L.L.Minku@cs.bham.ac.uk

ABSTRACT

Learning from evolving streaming data has become a ‘hot’ research topic in the last decade and many adaptive learning algorithms have been developed. This research was stimulated by rapidly growing amounts of industrial, transactional, sensor and other business data that arrives in real time and needs to be mined in real time. Under such circumstances, constant manual adjustment of models is inefficient and with increasing amounts of data is becoming infeasible. Nevertheless, *adaptive* learning models are still rarely employed in business applications in practice. In the light of rapidly growing structurally rich ‘big data’, new generation of parallel computing solutions and cloud computing services as well as recent advances in portable computing devices, this article aims to identify the current key research directions to be taken to bring the adaptive learning closer to application needs. We identify six forthcoming challenges in designing and building adaptive learning (prediction) systems: making adaptive systems scalable, dealing with realistic data, improving usability and trust, integrating expert knowledge, taking into account various application needs, and moving from adaptive algorithms towards adaptive tools. Those challenges are critical for the evolving stream settings, as the process of model building needs to be fully automated and continuous.

1. INTRODUCTION

Our digital universe is rapidly growing. Nowadays, the quantity of data available is doubling every two years. A study by IDC sponsored by EMC Corporation [21] estimates that the data created in 2011 will be 1.8 zettabytes (1.8 trillion gigabytes), and this amount will continue growing by a factor of 9 in the next five years. Adaptive systems need to consider this growth, as in the next years there will be much more data available to mine than in previous years.

The IDC study notes that 75% of the information in the

digital universe is generated by individuals. From this information knowledge about individuals and for individuals can potentially be extracted, bringing new opportunities to businesses, education and leisure. It is estimated that enterprises have some liability for 80% of all the digital information. As amounts of data available for knowledge extraction for enterprises are increasing, so does the demand for real time data analytics using automated data mining tools.

Predictive models are built from data and applied to unseen data. Data arrives and evolves over time, constant manual adjustment of models is inefficient and with increasing amounts of data is quickly becoming infeasible. Thus, predictive models need to have opportunities to update or retrain themselves, otherwise their accuracy will degrade. In research community attention to on-line learning scenarios has been rapidly increasing. In the last decade many adaptive learning algorithms and techniques have been developed (see e.g. [5, 16, 27, 47] for overviews). Nevertheless, *adaptive* learning models are still rarely deployed in business applications in practice.

This article considers critical issues that are limiting deployment of adaptive models. Following our collaborations with industrial partners and scientific discussions, we identify the key challenges and discuss the research directions to be taken to bring adaptive learning closer to applications. This article follows up a panel discussion organized at the workshop on Adaptive Prediction Systems in Bournemouth, UK on the 4th of August, 2011.

Requirements for data mining and machine learning in general and smart adaptive systems specifically have been repeatedly discussed in the last 20 years (for instance, see [11, 13, 17, 22]). Many challenges have been pointed out, some of the issues, such as making learning algorithms adaptive, have advanced a lot. Simultaneously, the context of evolving stream mining has substantially changed in the last decade, presenting a new setting from today’s perspective, with:

- very large and rapidly growing amounts of data;
- new generation of parallel computing solutions;
- retractable on-demand computation (public clouds);
- advanced portable computing devices;

¹corresponding author

- aggregated unstructured rich *big data*.

Firstly, the amounts of data have increased enormously and the digital universe is expected to grow much further and faster than data storage and processing capacities [21]. To handle the situation, computational power has grown enormously and hardware solutions for parallelization, such as Hadoop MapReduce [10], have been developed. Moreover, *resizable* on demand computational power now can be outsourced, for instance, from Amazon EC2¹ or GoGrid². On the other hand, novel portable devices, mobile phones and pads, have been developed that need advanced computational intelligence, and have to deal with very limited computational resources. In addition, the variety of data types and sources has expanded (e.g. social media) and increased in popularity. Nowadays, data is no longer in isolation and data sources have potential to complement each other (e.g. Google web services, social networks), much richer data is available and can be aggregated to extract knowledge.

As a result of those developments, some of the previous challenges in building adaptive learning systems have become more critical and new challenges emerged. In the light of those developments, we revisit and discuss the challenges for adaptive systems from the current time perspective.

This article identifies and discusses six key challenges in designing and building adaptive learning (prediction) systems from the applications perspective:

1. making adaptive systems scalable,
2. dealing with realistic data,
3. improving usability and trust,
4. integrating expert knowledge,
5. taking into account various application needs, and
6. moving from adaptive algorithms towards adaptive tools.

Some of these challenges apply to the traditional data mining as well; however, they are critical for the evolving stream setting. While in the traditional settings an engineer makes the final decisions in model building, in the evolving stream settings models train themselves *automatically* and *continuously*. While in the traditional settings an engineer can coordinate the learning process, verify and re-validate models manually, in the evolving stream settings models train themselves automatically, thus learning and adaptation mechanisms need to be trustworthy and transparent to users, noisy realistic data need to be handled automatically in a robust manner, application specific requirements need to be handled automatically as well. Moreover, in the traditional settings computationally heavy learning algorithms may be tolerated as the model is trained only once; in the stream settings the model may be retrained thousands of times, thus, scalability is critical. Thus, when building adaptive learning systems it is essential to handle explicitly the environment and application specific challenges as well as relax the need for human control.

Along with further discussion of these challenges we present our position where the forthcoming focus of the research and development efforts should be directed to address these challenges. The remainder of the article presents each challenge and discusses their implications.

¹<http://aws.amazon.com/ec2/>

²<http://www.gogrid.com/>

2. MAKING ADAPTIVE SYSTEMS SCALABLE

Nowadays, the quantity of data available is doubling every two years. As stated in the introduction, the ICD study [21] estimates that the data created in 2011 will be 1.8 zettabytes, and this amount will be continue growing by a factor of 9 in the next five years. This growth of data is an important fact that adaptive systems need to consider, as in the next years there will be much more data available to use and predict than in previous years.

To be able to deal with this massive data, it is necessary to scale our current adaptive systems to more demanding data volumes. We need to speed up learning processes, using software or/and hardware techniques. A way to create faster methods is using software data stream mining methodologies, where new arriving elements have to be processed essentially in real time. Scaling up techniques using hardware are based mainly in parallelization, cloud computing, fixed memory, mobile applications, and grid computing.

The most representative software based methodology was presented in [24], where a general method to learn from arbitrarily large databases is proposed. The method consists of deriving an upper bound for the learner's loss as a function of the number of examples used in each step of the algorithm. Then use this to minimize the number of examples required at each step, while guaranteeing that the model produced does not differ significantly from the one that would be obtained with infinite data. This general methodology has been successfully applied in *k*-means clustering [24], hierarchical clustering of variables [41], decision trees [12, 25], regression trees [26], decision rules [20], ensemble methods [6].

Learning from large datasets may be more effective when using algorithms that place greater emphasis on bias management [18]. One such algorithms is the Very Fast Decision Tree system [12]. VFDT is a decision-tree learning algorithm that dynamically adjusts its bias whenever new examples are available. It was designed to process thousands of examples per second using few computational resources, namely limited memory. The basic idea consists of using a small set of examples to select the splitting-test to incorporate in a decision tree node. It only makes a decision (i.e., adds a splitting-test in that node), when there is enough statistical evidence in favor of a particular test. This strategy guarantees model stability, controls overfitting, while it may achieve an increased number of degrees of freedom with increasing number of examples. Theoretically the Hoeffding trees are asymptotically nearly identical to that of a non-incremental learner using infinitely many examples [12].

We see that scalable incremental learning techniques are already available for selected base learners (mostly classifiers); however, the scope of learners needs to be expanded. Regression learners, that are particularly relevant for production industry applications, yet have not been scaled.

Another way to scale up the adaptive prediction system is to distribute the training process onto several machines. There are several different strategies to parallelize machine learning methods. We discuss briefly cloud computing, grid computing and MapReduce.

Hadoop MapReduce [10] is a programming model and software framework for writing applications that rapidly process large amounts of data in parallel on large clusters of com-

pute nodes. A MapReduce job divides the input dataset into independent subsets that are processed by map tasks in parallel. This step of mapping is then followed by a step of reducing tasks. These reduce tasks use the output of the maps to obtain the final result of the job.

S4 [40] and Storm [35] are distributed and scalable platforms that allows programmers to develop applications for processing continuous unbounded streams of data. For example, ensemble learning classifiers are easier to scale and parallelize than single classifier methods. They are the obvious candidate methods to implement using MapReduce, S4 or Storm techniques.

Cloud computing and *grid computing* are conceptually similar terms for denoting the use of computing resources being consumed like the electricity power grid. *Cloud computing* [39] is a service that gives computational power, and data storage without the user needing to know what hardware is used. The main characteristics of cloud computing are scalability and speed, to allow the deliverance of services in real-time. Main applications of cloud computing are internet services, as email, blog, and micro-blog websites.

Grid computing [39] is a computing infrastructure composed by a large number of computer devices, that can be located in different geographical places. The main difference of grid computing with supercomputers, is that supercomputers are optimized to have faster interprocessor interconnections, reducing the time and cost of moving the data between processors.

Cloud computing may use grid computing as hardware, and Hadoop MapReduce, S4 or Storm as software. A challenge on adaptive systems research is how to address scalability of systems combining hardware and software techniques trying to minimize the use of resources in an efficient way.

In the previous paragraphs, computational power is abundant and thousands of computers may work collectively and in parallel to solve problems. In the opposite side of the spectrum, data mining becomes ubiquitous. Computer power is cheap and widely available (PDAs, smart-phones, GPS devices, smart-meters, etc). Simple objects that surround us are gaining sensors, computational power, and actuators, and are changing from static, into adaptive and reactive systems. In these contexts, data mining algorithms will have to use limited computational resources, in terms of computations, memory, communications, and battery. The dissemination of these devices, users might request sheer amounts of data of interest to be streamed to their mobile devices. Storing, retrieving, and querying these huge amounts of data are infeasible due to resource limitations. Data stream mining can play an important role in helping mobile users in on-the-move decision making [18].

To address the challenges presented by massive and further increasing amounts of data research efforts need to be directed towards: (1) developing new incremental algorithms and transforming existing learning algorithms to operate in the incremental online mode; (2) developing techniques that would enable learning algorithms to operate within new hardware solutions, such as grid, cloud computing and parallel processing; (3) developing techniques that would be able to run data mining algorithms in resource-aware devices, like PDAs, mobile-phones, sensors. (4) developing anytime algorithms able to return an approximation of the correct answer, depending on the amount of computation they were able to perform.

3. DEALING WITH REALISTIC DATA

The mainstream adaptive learning algorithms concentrate on adaptation techniques. Typically it is assumed that data arrives already pre-processed or pre-processing filter is tied to the prediction algorithm, and that the feedback for driving adaptation (the ground truth) is immediately available after casting each prediction and before any new data arrives. The real streaming data application settings are often not that perfect. Therefore, algorithms need to be developed to work with such realistic data and learn from it in an automated way.

In real data stream applications pre-processing is a very important step of data mining process, as real data often comes from complex environments and is often noisy, redundant, contains missing values. Data mining practitioners say (e.g. [7]) that data preparation takes 80 – 90% of a data mining project time, which means that modelling can take as little as 10%. In contrast, adaptive learning research concentrates on designing adaptive predictors, while data preparation and pre-processing steps are often overlooked. Although the problem of automating pre-processing is applicable to the traditional data mining settings, it is particularly relevant to the evolving streams scenario, as models need to be regularly updated. Automated on-line predictive models will give very limited benefits in practical applications if pre-processing still needs to be periodically updated manually. Therefore, research aiming at automated learning from streaming data, *must* automate data preparation and pre-processing steps as well. Moreover, as data is expected to evolve over time, pre-processing elements need to have adaptation mechanisms in line with predictive elements. Thus, research and development efforts are needed towards ability to predict from incomplete data, automating pre-processing and making it robust over time, taking into account suitability of data for prediction and specifying data collection process. In addition, evolving streams scenario may raise additional challenges because of the speed with which data arrives, thus pre-processing needs to be scalable as well as the main learning algorithms. Not only the operation of pre-processing filters needs to be scalable; the process of automatically building pre-processing filters online needs to be scalable. Scalability of operation has been to some extent addressed in image processing (incremental PCA techniques, e.g. [46]), while building adaptive pre-processing filters, as well as making this process scalable, to the best of our knowledge has not been addressed before.

In addition, in real data stream applications the true labels (the true values of target variables) often are not available immediately after casting the prediction, as assumed and required by majority of the adaptive learning algorithms. If in reality labels were arriving immediately, one could argue that the need for prediction in this application is limited, as in a few moments we would know the truth anyway.

In reality labels may arrive with a delay (e.g. bankruptcy in credit scoring prediction, laboratory test results in assessment of product quality in chemical production industry) that may range from a few hours to months or even years. Moreover, obtaining true labels often requires human efforts (e.g. confirming a fraud in credit card monitoring, confirming sentiment in text messages, performing laboratory test in assessment of product quality) and it is not realistic to expect to receive labels for all the incoming instances. While

in the traditional settings a set of labels for training can be arranged and collected retrospectively, in the evolving stream settings availability and reliability of labels online is a critical ingredient to make an adaptive learning possible. Timely and accurate feedback is essential for majority of the adaptation mechanisms, which rely on monitoring predictive performance over time and act if the performance degrades. Recently researchers started to address the issue of limited feedback in adaptive learning (e.g. [31, 33, 48]), while many more scenarios addressing selective feedback, dealing with noisy feedback and varying delays still need to be addressed and learning algorithms for reliable performance in those scenarios need to be developed.

Overall, to deal with the challenges presented by realistic real time data research efforts need to focus on developing algorithms for automating data mining process as a whole, including data preparation, preprocessing, prediction and the feedback loop.

4. IMPROVING USABILITY AND TRUST

Adaptive predictive systems are intrinsically parametrised. In most of the cases, setting these parameters, or tuning them is a difficult task. This in turn affects the usability of the systems negatively. In the panel discussion, the industry representatives have clearly indicated that it is strongly desired for the system to have as few user adjustable parameters as possible. That is particularly relevant for online learning from evolving streaming setting, since manual optimization of the parameters will not be accessible during continuous re-training of the model.

This obviously raises the issue of parameter setting and tuning. It is strongly desired to either mask the parameter setting to map to an interpretable outcome by the user, or to design the system with self-adjusting parameters.

On a related issue, adaptive predictive systems introduce approximation of the results. When dealing with stream mining techniques, this adds up another layer of approximation, by which most of the stream mining algorithms are attributed [16, 19]. This has a clear negative effect on the user's trust. Adding approximation layers from the user's viewpoint makes the system less reliable and less trustworthy.

For example, when using a large body of stream mining algorithms that are based on the *Hoeffding bound* in predictive systems [12, 25], it is extremely hard for the users to determine the settings for the *Hoeffding bound* parameters. Moreover, if these parameters adapt over time to respond to changes, it is important to determine how the tuning of parameters should be done. Furthermore, it is difficult to ensure that these settings and tuning of parameters result in trustworthy and reliable results. Finally, to respond to concept drift, new parameters may be introduced like sliding windows and dynamic ensemble [32]. Usability and trust are again affected, as the window size and participating classifiers in the ensemble are generally hard to set, making the system less usable. Also, using one of the techniques available for setting such parameters, or adapting them over time has a negative effect on the trust issue, as these system's changes bring uncertainty in the outcome.

Two advances/developments can lead to improving the usability and trust of adaptive predictive systems. First, self-adjustment and parameter masking, as previously mentioned,

would greatly help improving the system's usability. Second, wide deployment of such systems increases the trust in the outcome of these systems.

It is worth noting that we have in this section discussed the objective side of trust. In the following section, trust is discussed from its subjective aspect, as trusting the system is also related to understanding the rules that explain its behavior.

5. INTEGRATING EXPERT KNOWLEDGE

Most works on adaptive systems are concerned with the question of how to use learning machines in changing environments without considering the possibility of incorporating knowledge provided by experts or explaining the system's behavior using interpretable rules. As examples, [3, 30, 37, 42, 44, 45] all investigate how to use existing learning machines for creating adaptive systems without mentioning integration with expert knowledge.

Nevertheless, there is frequently a great deal of scepticism from practitioners when considering whether or not to adopt machine learning solutions. Two reasons for that are: (1) experts have valuable knowledge that may be able to improve or validate the system and (2) it is difficult to believe in a black box. When using traditional machine learning approaches, these issues have been dealt with by explaining the machines' behavior using rules. Such rules have shown to increase the acceptance of the learning system by users and experts, allowing verification on whether the reasoning mechanism is sound and possibly even improving its accuracy through rules insertion [34].

When considering adaptive systems, nevertheless, several additional challenges must be addressed. One of the first issues specific to this area is related to the timing of rules insertion and extraction. In the traditional machine learning approaches, a sequential process is frequently adopted to insert rules, learn from examples and extract rules. Rules are inserted or extracted only at the beginning and the end of the learning process. However, the continuous or incremental learning nature of adaptive systems makes it impossible to adopt such a scheme, as there is no ending point for the learning process. So, when to insert and extract rules? When to communicate with experts?

Another issue specific to this area is that the explanation of the system and the incorporation of expert knowledge involve not only how the adaptive system performs predictions, but also how it deals with changes. The expert needs to understand and believe that the system is really going to react to changes when they happen. So, the mechanism to deal with changes itself needs to be explained. Moreover, experts may have not only knowledge about how to make predictions considering the current concept, but also about when the environment is likely to present changes and what type of changes. So, how to integrate this knowledge into the system? What is the best way to represent it? Answers to these questions are not straightforward, as they may involve not only the use of rules, but also modifications in the way the system reacts to changes in order to accommodate and benefit from expert knowledge. Besides, the type of changes handling mechanism (e.g., weighted ensembles [30, 37] or drift detection methods [3, 37]) would highly influence any proposed mechanism.

In addition, as adaptive systems operate in changing envi-

ronments, it is important not only to explain the behavior of the system, but also what has changed in comparison to previous points in time, and how different the system became. It is worth noting that changes can happen more suddenly or more slowly, and can be large or small [36]. Explaining the system considering these points would help not to overload the expert with too many new rules. Research in this area would possibly involve not only comparisons between the current and previous states of the learning machines during and after changes, but also re-designing the learning machine itself to support such analyzes.

As we can see, there are several challenges regarding integration of expert knowledge that need to be tackled for bridging the gap between academic and real world application of adaptive systems.

6. TAKING INTO ACCOUNT DIFFERENT APPLICATION NEEDS

One of the challenges when talking about adaptive systems is the wide range of application areas. There are many problems to which the adaptive systems can be applied, e.g. classification, dynamic filtering, prediction and clustering. In practice these problems are present in many areas such as finance, industry, medicine, software, etc. Although all of the adaptive systems cope with the information available in different application areas and use the same methods to solve the problems there is no one adaptive learning system that can be applied in all situations. One of the challenges for future adaptive systems is to develop a software solution that will be able to train and deploy an appropriate model regardless of the application area. Streaming data in different applications have not only different format and size but also frequency with which the data points arrive to the system. This makes the issue of developing such multi-purpose software even more complex than in the case of processing data that has been already gathered in the database. From the software engineering perspective, coping with inputs that can significantly differ and that come into the system in a stream is a great challenge. Additional effort is required to dynamically manage the access to both memory and CPU, to be able to process the online data.

The next challenge from the user interface modelling perspective is how to represent such dynamic processes as: (i) pre-processing of streaming data, (ii) training and running the predictive models that adapt over time to new incoming data, (iii) and presenting the results of the analyses for different applications in a meaningful manner that it could be easily understood by the end user. In addition, user interface should enable different people to adapt their views to their needs, e.g. in a way that one system can be used in predicting financial markets and another in medical diagnostics.

Another issue connected with the application of adaptive systems is: how to use the results that these kind of systems produce. Should the result be treated in terms of recommendation, decision support or rather as input for further processes without human intervention? Should adaptive systems assist in decision making process or rather make decisions instead of a human being?

The recommendations are one of the ways to support people in their decisions by suggesting the things one would like to buy, watch, read, do, etc [1, 8, 38]. Most widely ap-

plied recommendation methods are statistical analysis (e.g. ratings), demographic filtering, content-based filtering, and collaborative filtering. The first enumerated method is not personalized and the rest of them are personalized. Recommendations are created based on the information about users and their activities.

Decision support systems (DSS) are used to help people to make decisions and suggest the solutions to their problems but the final decision in most cases is made by an expert. The major application for DSS is creating, manipulating and optimizing of simulation models, accessing and analysing large databases, containing both historical and real-time data, and supporting individual and group decision making [2, 14]. Model-driven, data-driven, communication-driven, document-driven, knowledge-driven and Web-based DSSs are groups of systems that support decision making using different types of approaches.

Though a lot of research still needs to be done, both recommender systems and decision support systems can be and nowadays usually are developed as systems that adapt to the changing input data. Both individuals and organizations widely use these systems to obtain guidelines in their decision making processes. However, adaptive systems that can make autonomous decisions without consulting them with users, are still at the beginning of their deployment path, e.g. autonomous vehicles [4]. People are reluctant to let computers take full control, they do not trust the machines and do not want to be excluded from the process of decision making. This is mainly due to the fact that decision making process is tightly connected with the concept of uncertainty that is a state of limited information or knowledge [43]. Peoples perception is that if they have difficulties with coping with the changing situations and predicting the consequences of some actions, then how any artificial system can properly react to the evolving, online data? In decision theory decision problems can be divided into: (i) decisions under certainty, (ii) decisions with risk, and (iii) decisions under strict uncertainty [15]. In the first situation one can make fully informative decision. In the second case, not all information is certain but one can infer a probability distribution of possible outcomes. The last situation refers to the situation when no information is available. From the computer science perspective only the first decision problem can be truly solved e.g. using expert systems. The rest of the problems are connected with some level of uncertainty and in such a case people are not keen to give the control to the machine. Thus, complex adaptive systems that are currently developed are just used to support the decision not to make them.

The concept of a complex adaptive system is that it is able to autonomously adapt their behaviour to changing environment [23]. The issues connected with this are security and safety of such systems, especially in the context of safety critical applications as well as trust (as discussed in Section 4), i.e. are the systems trustworthy to the extent that people would allow them to make autonomous decisions? These issues, although common for all predictive systems, are much harder to overcome when the evolving, streaming data instead of data previously gathered in database are processed. This is connected with the fact that it cannot be fully predicted what data will come into the system in the next step and what would be the consequence of this new data point entering the system. This introduces another

uncertainty level what causes additional users concerns in regard to how trustworthy adaptive predictive systems are. The future research will focus on addressing the issue of building adaptive systems that can be trusted and reliable. In the same time the work needs to be done to overcome the psychological issue that even if the system can be trusted, people will not necessarily be keen to trust it.

7. MOVING FROM ADAPTIVE ALGORITHMS TOWARDS ADAPTIVE TOOLS

With the explosion of generated and stored data there have been also a lot of effort dedicated to developing software tools that can take advantage of such data in various businesses. According to Davenport and Harris [9] it is now virtually impossible to differentiate yourself from the competitors on the product alone and the companies now compete on advanced analytics. Those who are able to use advanced data collection and analysis in their decision making processes have been shown to seize the lead in their fields. So it can be firmly said that complex predictive modelling has left its childhood stage in the academic bubble and started to be a success story in a wide range of enterprises.

With the increasing number of available analytical tools in this area, it is no longer only experts who can generate useful predictions from the vast amounts of data but more and more sophisticated user interfaces aided by automation mechanisms help non-expert users to exploit and extract knowledge from their data.

Recent Forrester's report [29] on Predictive Analytics and Data Mining Solutions provides detailed information on a number of very advanced and comprehensive tools offered by such well known vendors as IBM, SAS Institute, KXEN, Oracle and Portrait Software who were found to be the leaders in the predictive analytics market. According to the report author, James Kobielus, some of the smaller strong performing vendors have established themselves as innovators in functionality in such key areas as the wizard-driven development automation, multi-business scenario modelling, interactive visualization, content analytics, sentiment analysis, social network analysis, in-stream analytics, and open-source modelling languages.

As comprehensive as the currently available software platforms and tools are there are numerous challenges that keep being highlighted and which are notoriously difficult to overcome. Some of them are further discussed in James Kobielus' Blog [28] on Advanced Analytics Predictions for 2010 and are closely linked to the issues of adaptivity of methods and interfaces, automation of analysis and models generation and abstracting out from the details of sophisticated underlying data analysis algorithms and setting of their parameters for non-expert users.

While the research into learning and adaptation algorithms is vigorously pursued in academia, the level of trust and robustness required for highly adaptive tools to be used by non-expert users and/or to be deployed in unfamiliar business settings with limited human supervision is still missing. There is an inherent conflict between the desire for the tools and algorithms to be adaptive/learning/autonomous and the releasing of the control by humans present in the decision making loop which was mentioned in previous sections. As already discussed in [17], it is felt that one of the

main challenges to overcome by the developers of adaptive algorithms is to prove the robustness, accuracy, safety and stability of their performance under a wide set of conditions requiring adaptation over time.

8. FUTURE RESEARCH IN ADAPTIVE SYSTEMS

We identified six current challenges for adaptive systems and discussed the most urgent research directions to be taken to bring adaptive systems closer to practical use. From scalability perspective research should focus on incremental, resource aware algorithms and parallelisation techniques for adaptive algorithms. The demand for incremental learning has been quickly growing and has become crucial in the light of popularity of intelligent mobile devices and unprecedented accumulation of data. From the resource perspective, computational power is becoming a tradable commodity on the market, thus it is essential to optimize the amount of computational power that is required in order to balance benefits and costs of adaptivity.

From real time data perspective we need to focus on developing adaptive data mining processes that would fully integrate data mining steps from data preparation to the feedback. Now, when more and more adaptive predictive algorithms become available, this need is urgent, as these algorithms cannot be put into real time use without automating the full data mining process. Thus, to deal with the challenges presented by realistic, real time data, research efforts need to focus on automating data mining process as a whole, including data preparation, preprocessing, prediction and the feedback loop.

Transparency of the methods to the users was important in off-line data mining algorithms. However, transparency is now essential in data stream settings, as not only the predictors, but more importantly, the adaptation mechanisms need to be transparent to the users. Off-line models could be validated before putting into use, adaptive systems need to work in an autonomous way with minimum amount of human intervention, thus, opportunities for re-validation are extremely limited. Thus, not only predictors, but also adaptation mechanisms need to be transparent. Research should take into account not only trust from the users, but also reliability in safety critical applications, as well as possible adversary actions that could drive an adaptive system to adapt towards an undesired state. Thus research efforts are needed towards moving from the 'black box' to the 'white box' algorithms, as well as establishing mechanisms for monitoring and controlling adaptation.

In addition, research efforts needed to be put towards usability of adaptive systems in terms of system tuning and parameter setting. While in stationary environments domain experts still could afford to spend time on tuning the parameters, with increasing amounts of data that becomes not only costly but also nearly infeasible. Therefore, self-adjustment and parameter masking in adaptive systems would greatly help improving the system's usability. The task is even more challenging, given that tuning needs to be transparent.

Furthermore, with massive amounts of data it is becoming crucial to balance the role of experts and learning machines in the use of expert feedback about the system performance (internal) as well as about environment changes (external). Firstly, the trade-off between costs and benefits while using

experts need to be taken into account. Secondly, interfaces for human-computer interaction for efficient and effective use of feedback need to be developed.

The variety of data types and sources calls for specialized algorithmic solutions for different applications. The expansion of the digital universe presents us with nearly unlimited variations in data mining tasks. On one hand, specialized solutions are also required, on the other hand, generalizations and systematic descriptions of the application tasks are also required in order not to reinvent the same techniques again and again.

Finally, there is a need for integrated software tools, that could be deployed and operate autonomously, provide robust and reliable performance over long periods of time without human interaction. Research and software development need to join forces to move from research tools towards applicable tools and decision support systems for industry. That is essential in data stream settings since the amounts of data and operational settings limit the possibilities and feasibility of manual support.

Adaptive learning systems have advanced a lot during the last decade, but there is still long way to go before they become commonly adopted in applications. We hope that our discussion will add to the awareness and interest towards these research problems.

9. ACKNOWLEDGEMENTS

The INFER-APS workshop leading to this discussion was organized within INFER project that has received funding from the European Commission within the Marie Curie Industry and Academia Partnerships and Pathways (IAPP) programme under grant agreement no. 251617. I.Žliobaitė, B.Gabrys and K.Musial research is funded by this grant, which is greatly acknowledged. L.Minku research is funded by EPSRC grant no. EP/D052785/1. J.Gama thanks to the financial support of the project Knowledge Discovery from Ubiquitous Data Streams (PTDC/EIA-EIA/098355/2008).

10. REFERENCES

- [1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
- [2] D. Arnott and G. Pervan. A critical analysis of decision support systems research. *Journal of Information Technology*, 20(2):67–87, 2005.
- [3] M. Baena-Garcia, J. D. Campo-Avila, R. Fidalgo, and A. Bifet. Early drift detection method. In *Proc. of the 4th ECML PKDD Int. Workshop on Knowledge Discovery From Data Streams (IWKDD)*, pages 77–86, 2006.
- [4] M. Bajracharya, M. Maimone, and D. Helmick. Autonomy for mars rovers: Past, present, and future. *Computer*, 41(12):44–50, 2008.
- [5] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer. Data stream mining: A practical approach. Technical report, University of Waikato, 2011.
- [6] A. Bifet, G. Holmes, and B. Pfahringer. Leveraging bagging for evolving data streams. In *Proc. of the 2010 European conf. on Machine learning and knowledge discovery in databases (ECML PKDD)*, pages 135–150, 2010.
- [7] T. Breur. Toms ten data tips, 2007.
- [8] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User Adapted Interaction*, 12(4):331–370, 2002.
- [9] T. H. Davenport and J. G. Harris. *Competing on analytics : the new science of winning*. Boston, Mass. : Harvard Business School Press, 2007.
- [10] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Commun. ACM*, 51:107–113, 2008.
- [11] T. G. Dietterich. Machine learning research: Four current directions. *AI Magazine*, 18(4):97–136, 1997.
- [12] P. Domingos and G. Hulten. Mining high-speed data streams. In *Proc. of the ACM 6th Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 71–80, 2000.
- [13] J. Doyle and T. Dean. Strategic directions in artificial intelligence. *ACM Computing Surveys*, 28:653–670, 1996.
- [14] S. Eom. *Decision Support Systems Research (1970-1999): A Cumulative Tradition and Reference Disciplines*. Edwin Mellen Press, Lewiston, New York, 2002.
- [15] S. French. *Decision Theory: An Introduction to the Mathematics of Rationality*. Ellis Horwood Halsted Press, New York, 1986.
- [16] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy. Mining data streams: a review. *SIGMOD Rec.*, 34:18–26, 2005.
- [17] B. Gabrys. Do smart adaptive systems exist? introduction. In B. Gabrys, K. Leiviska, and J. Strackeljan, editors, *Do Smart Adaptive Systems Exist?*, volume 173 of *Studies in Fuzziness and Soft Computing*, pages 1–17. Springer, 2005.
- [18] J. Gama. *KnowledgeDiscovery from Data Streams*. Data Mining and Knowledge Discovery. Chapman & Hall CRC Press, Atlanta, US, 2010.
- [19] J. Gama, M. M. Gaber, and S. Krishnaswamy. Data stream mining: from theory to applications and from stationary to mobile. A tutorial at the ACM 25th Symposium on Applied Computing SAC 2010, March 2010.
- [20] J. Gama and P. Kosina. Learning decision rules from data streams. In *Proc. of the 22nd Int. Joint Conf. on Artificial Intelligence(IJCAI)*, pages 1255–1260, 2011.
- [21] J. Gantz and D. Reinsel. The 2011 IDC digital universe study: Extracting value from chaos. June 2011.
- [22] D. J. Hand. Classifier technology and the illusion of progress. *Statistical Science*, 21:1–14, 2006.
- [23] J. Holland. *Hidden Order: How Adaptation Builds Complexity*. Basic Books, New York, USA, 1996.

- [24] G. Hulten and P. Domingos. Catching up with the data: research issues in mining data streams. In *Proc. of Workshop on Research Issues in Data Mining and Knowledge Discovery*, Santa Baraba, USA, 2001.
- [25] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *Proc. of the 7th ACM SIGKDD int. conf. on Knowledge discovery and data mining (KDD)*, pages 97–106, 2001.
- [26] E. Ikonomovska, J. Gama, and S. Dzeroski. Learning model trees from evolving data streams. *Data Mining and Knowledge Discovery*, 23:128–168, 2011.
- [27] P. Kadlec, R. Grbic, and B. Gabrys. Review of adaptation mechanisms for data-driven soft sensors. *Computers & Chemical Engineering*, 35(1):1–24, 2011.
- [28] J. Kobielius. Advanced analytics predictions for 2010. Technical report, 2009.
- [29] J. Kobielius. The forrester wave: Predictive analytics and data mining solutions, Q1 2010. Technical report, Forrester Research, Inc., 2010.
- [30] J. Z. Kolter and M. A. Maloof. Dynamic weighted majority: An ensemble method for drifting concepts. *Journal of Machine Learning Research*, 8:2755–2790, 2007.
- [31] L. Kuncheva and J. Sanchez. Nearest neighbour classifiers for streaming data with delayed labelling. In *Proc. of the 2008 8th IEEE Int. Conf. on Data Mining (ICDM)*, pages 869–874, 2008.
- [32] L. Kuncheva and I. Zliobaite. On the window size for classification in changing environments. *Intelligent Data Analysis*, 13(6):861–872, 2009.
- [33] P. Lindstrom, S. J. Delany, and B. M. Namee. Handling concept drift in a text data stream constrained by high labelling cost. In *Proc. of the 23rd Int. Florida Artificial Intel. Research Society Conference (FLAIRS)*, 2010.
- [34] T. Ludermir, M. Souto, and W. Oliveira. On a hybrid weightless neural system. *International Journal of Bio-Inspired Computation*, 1:93–104, 2009.
- [35] N. Marz. Storm: Distributed and fault-tolerant real-time computation. 2011.
- [36] L. L. Minku, A. White, and X. Yao. The impact of diversity on on-line ensemble learning in the presence of concept drift. *IEEE TKDE*, 22:730–742, 2010.
- [37] L. L. Minku and X. Yao. DDD: A new ensemble approach for dealing with concept drift. *IEEE TKDE*, page 16p. (in press), 2012.
- [38] B. Montaner, M.; Lopez and J. De la Rosa. A taxonomy of recommender agents on the internet. *Artificial Intelligence Review*, 19:285–330, 2003.
- [39] J. Myerson. Cloud computing versus grid computing. IBM developerWorks Article, <http://www.ibm.com/developerworks/web/library/wa-cloudgrid/>, 2009.
- [40] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari. S4: Distributed stream computing platform. In *ICDM Workshops*, pages 170–177, 2010.
- [41] P. P. Rodrigues, J. Gama, and J. P. Pedroso. Hierarchical clustering of time series data streams. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):615–627, 2008.
- [42] M. Scholz and R. Klinkenberg. Boosting classifiers for drifting concepts. *Intelligent Data Analysis*, 11(1):3–28, 2007.
- [43] D. Skinner. *Introduction to Decision Analysis*. Probabilistic Publishing, Gainesville, 3rd edition, 2009.
- [44] W. Street and Y. Kim. A streaming ensemble algorithm (SEA) for large-scale classification. In *Proc. of the 7th ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 377–382, 2001.
- [45] H. Wang, W. Fan, P. S. Yu, and J. Han. Mining concept-drifting data streams using ensemble classifiers. In *Proc. of the 9th ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 226–235, 2003.
- [46] H. Zhao, P. C. Yuen, and J. T. Kwok. A novel incremental principal component analysis and its application for face recognition. *IEEE transactions on systems man and cybernetics Part B Cybernetics a publication of the IEEE Systems Man and Cybernetics Society*, 36(4):873–886, 2006.
- [47] I. Zliobaite. Learning under concept drift: an overview. Technical report, Vilnius University, 2009.
- [48] I. Zliobaite, A. Bifet, B. Pfahringer, and G. Holmes. Active learning with evolving streaming data. In *Proc. of the European Conf. on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pages 597–612, 2011.