THE IEEE
Intelligent Informatics
BULLETIN

IEEE COMPUTER SOCIETY

IEEE Computer Society
Technical Committee
on Intelligent Informatics

December 2009 Vol. 10 No. 1 (ISSN 1727-5997)

On-line version: http://www.comp.hkbu.edu.hk/~iib (ISSN 1727-6004)

The Technical Committee on Intelligent Informatics (TCII) of the IEEE Computer Society deals with tools and systems using biologically and linguistically motivated computational paradigms such as artificial neural networks, fuzzy logic, evolutionary optimization, rough sets, data mining, Web intelligence, intelligent agent technology, parallel and distributed information processing, and virtual reality. If you are a member of the IEEE Computer Society, you may join the TCII without cost. Just fill out the form at http://computer.org/tcsignup/.

The IEEE Intelligent Informatics Bulletin

Aims and Scope

The IEEE Intelligent Informatics Bulletin is the official publication of the Technical Committee on Intelligent Informatics (TCII) of the IEEE Computer Society, which is published once a year in both hardcopies and electronic copies. The contents of the Bulletin include (but may not be limited to):

1) Letters and Communications of the TCII Executive Committee

2) Feature Articles

3) R&D Profiles (R&D organizations, interview profile on individuals, and projects etc.)

4) Book Reviews

5) News, Reports, and Announcements (TCII sponsored or important/related activities)

Materials suitable for publication at the IEEE Intelligent Informatics Bulletin should be sent directly to the Associate Editors of respective sections.

Technical or survey articles are subject to peer reviews, and their scope may include the theories, methods, tools, techniques, systems, and experiences for/in developing and applying biologically and linguistically motivated computational paradigms, such as artificial neural networks, fuzzy logic, evolutionary optimization, rough sets, and self-organization in the research and application domains, such as data mining, Web intelligence, intelligent agent technology, parallel and distributed information processing, and virtual reality.

Publisher: *The IEEE Computer Society Technical Committee on Intelligent Informatics*
Address: *Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong (Attention: Dr. William K. Cheung; Email:william@comp.hkbu.edu.hk)*
ISSN Number: *1727-5997(printed)1727-6004(on-line)*
Abstracting and Indexing: *All the published articles will be submitted to the following on-line search engines and bibliographies databases for indexing—Google(www.google.com), The ResearchIndex(citeseer.nj.nec.com), The Collection of Computer Science Bibliographies (liinwww.ira.uka.de/bibliography/index.html), and **DBLP** Computer Science Bibliography (www.informatik.uni-trier.de/»ley/db/index.html).*

eCortex and the Computational Cognitive Neuroscience Lab at CU Boulder

BRIDGING THE GAP BETWEEN BIOLOGY AND COGNITION

I. INTRODUCTION

We founded eCortex, Inc. in 2006 to commercialize the scientific research conducted in the Computational Cognitive Neuroscience (CCN) laboratory at the University of Colorado in Boulder, headed by Dr. Randall C. O'Reilly. The company and the lab collaborate on much of their work, but each plays a somewhat different role in the overall research program.

For over 15 years, we have been asking two fundamental questions: "how does the brain think?" and "how can I capture that process in a computer program?" The brain is made of neurons, and at a high level our computer programs are straightforward implementations of standard "integrate and fire" equations that describe how neurons integrate information from, and send the results of the computation to, many thousands of other neurons.

What distinguishes our approach from others like it are the special equations (called Leabra – **l**ocal, **e**rror-driven & **a**ssociative, **b**iologically-**r**ealistic **a**lgorithm – suggestive of a balance of different learning forces, as in the Libra scale) that we use for getting our simulated neurons to learn in response to experience, as well as the kinds of biological and cognitive data we use to evaluate how our computer models are functioning. We configure the neurons in our models so that they capture the essential network circuitry and dynamic neural properties that are empirically observed in different brain areas, and then test the extent to which the models actually reproduce the kinds of learning and behavior that we know to occur in these brain areas.

We build these models using a software program called Emergent (http://grey.colorado.edu/emergent -- see Figure 1), which was developed in the CCN lab. Emergent is a comprehensive simulation environment for neural models, providing a graphical development environment, a highly flexible training and simulation engine, and powerful graphing and output capabilities. Emergent is available under the GPL open source license, and eCortex is its exclusive commercial licensee.

A historical example illustrates the power of our approach. It is now widely agreed that the hippocampus (a relatively old brain structure, in evolutionary terms, which is located inside the temporal lobes of the mammalian brain) is essential for forming much of what we typically think of as "memories" – the cataloging of daily events, facts, etc. In some of our earliest work with neural models, we showed that certain biological features of the hippocampus are critical for its ability to achieve this remarkable feat, and that these features are fundamentally in conflict with features characteristic of the cerebral cortex, where much of cognitive processing takes place (e.g., perception and language). Thus, we were able to clearly understand in explicit computational terms why the brain needs to have a specialized structure (the hippocampus) for "episodic" memories, in contrast to the relatively (but not entirely) homogeneous configuration of the cortex. In contrast to the hyper-specific mnemonic abilities of the hippocampus, the cortex excels at extracting generalities from among all the specific facts and events of our lives, and these "semantic" memories are essential for allowing us to behave sensibly when we confront new situations, where we have to apply our common-sense general world-knowledge.

II. THE COMMON-SENSE PROBLEM

More recently, we have been focusing on this common-sense ability, which many have argued is the most important differentiator between human and artificial intelligence. In addition to integrating over many particular experiences, human common sense is built upon a foundation of sensory and motor primitives that we learn early in childhood development. In essence, all

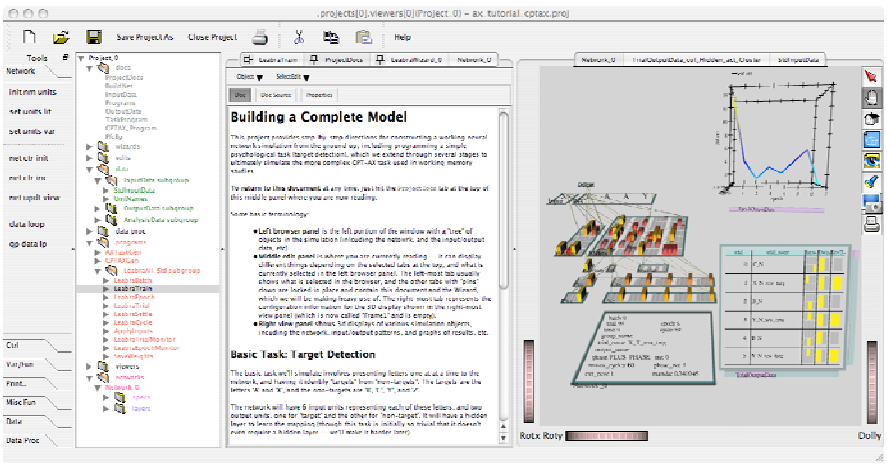


Figure 1: Screenshot of the emergent neural network simulation system, showing left-panel tree-browser of objects (network, control programs, documents, etc), middle editing panel, and right 3D visualization panel.

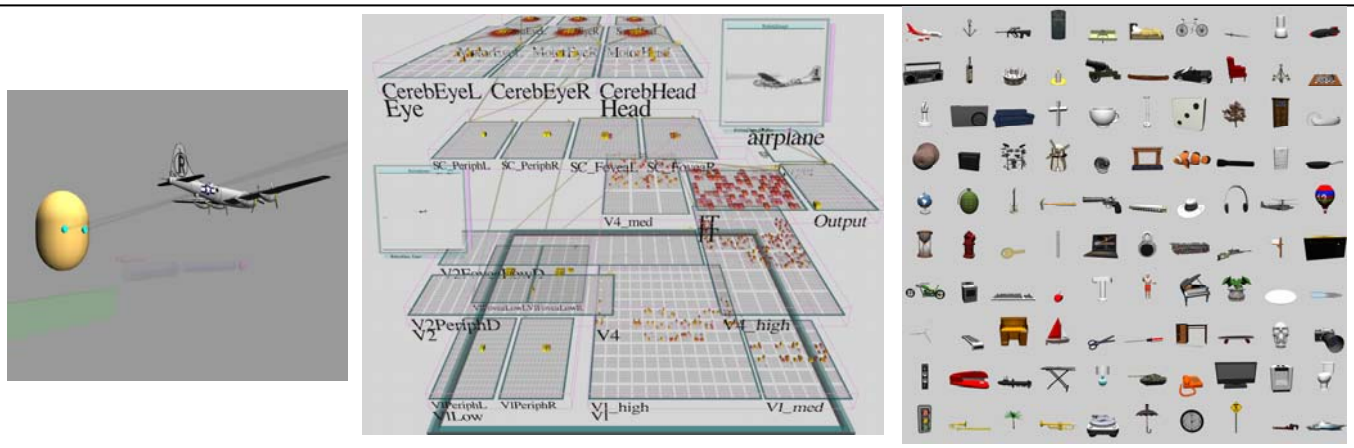


Figure 2: Our virtual agent (named "emer") shown fixating a 3D object within its virtual environment (left), which it can then name with high levels of accuracy, using biologically-based visual and motor pathways in its simulated brain (middle), and the 100 object categories used for visual object recognition testing (right).

abstract forms of cognition are "anchored" in concrete sensory-motor neural representations, and the particular "natural logic" that we learn so well in its concrete manifestations (i.e., an intuitive understanding of space, time, and everyday physics) can be leveraged when we launch off into new realms of understanding. In other words, all our knowledge and cognition is based on an analogy to something else, except that the buck stops at the eyeballs and the muscles.

Our approach to this problem is to apply the powerful Leabra learning mechanisms to fundamental sensory-motor tasks, such as visual object recognition, eye and head gaze control for scanning the environment, and manually reaching and interacting with objects in the world. To facilitate rapid development of this type of model, we have built a virtual simulation environment and incorporated a simulated robotic agent (Figure 2) within the Emergent software.

The virtual agent, named "emer," learns to recognize 100 different object categories with high levels of accuracy and, crucially, can then generalize this knowledge to novel object exemplars from the same categories, with 92.8% average generalization accuracy. Unlike other object recognition models, this ability emerges out of a set of neural processing units that all use the same learning algorithm, operating over hierarchically-organized neural layers. This network learns to break down the recognition problem over these layers, resulting in what can be described as a sensible "divide and conquer" approach. This outcome, which was not pre-wired, enables the network to handle the conflicting problems of distinguishing different object categories, while also collapsing across all of the spatial and other variability of objects within a category. Furthermore, the network is fully bidirectional (as is the brain). We have demonstrated that this architectural feature enables the model to deal with noisy or partially-occluded images much more robustly than the purely feed-forward models that are prevalent in the literature.

One goal of eCortex is to commercialize this robust object recognition ability. Under a U.S. Navy SBIR (Small Business Innovation Research) project, we applied the model to the problem of distinguishing floating objects in variable seas, and found it to be highly successful at this task. Other applications are under development, and we are always interested in new challenges, so please contact us if you have a problem that could be solved with this visual object recognition capability.

The motor abilities of the emer agent depend on specialized learning mechanisms and neural architecture associated with the cerebellum, in addition to the above more general-purpose cortical learning mechanisms. Together with its object recognition abilities, emer has a solid sensory-motor foundation for subsequent cognitive learning upon which to build. Importantly, we have found that simply by virtue of having a (simulated) body and a "point of view" within a 3D (simulated) environment, emer obtains a large quantity of highly informative training signals to shape sensory-motor learning. For example, the simple act of visually fixating an object provides several inputs that can be considered training signals: the fixation operates as a self-correcting feedback loop, and with slight extension, provides feedback information for the process of reaching for objects. Fixations also can naturally inform the visual system about what is figure versus ground. All of these natural training signals could be artificially generated in one way or another, but the fact that they come "for free" through normal everyday interactions with the environment suggests that they are an important component of human learning. Furthermore, any classification of these training signals is bound to be incomplete and less "holistic" than the interaction as a whole, and will miss the integration of the signal, wherein we suspect much of common-sense arises. This is just one example of how *embodiment* (having a physical body in a 3D world) is essential for developing common-sense knowledge, and one step along the longer path toward imbuing an artificial system with robust human-like intelligence.

III. Executive Control and Abstract Cognition

Another major focus of our research is on the prefrontal cortex (PFC) and basal ganglia (BG) (Figure 3), which are brain

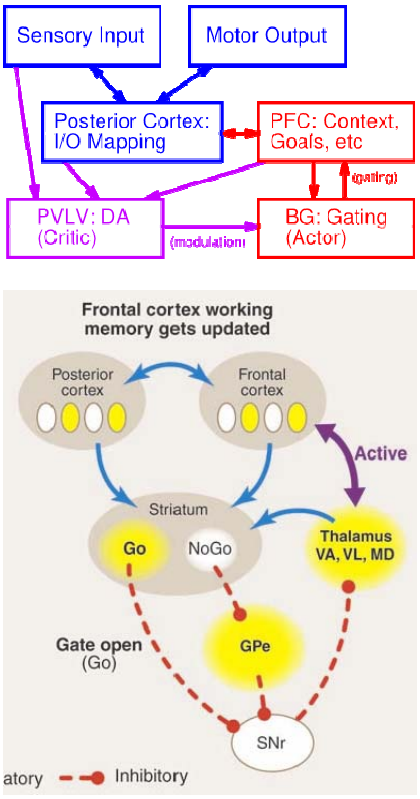


Figure 3. **P**refrontal-cortex **B**asal ganglia **W**orking **M**emory (PBWM) architecture for simulating executive control tasks, based on relevant brain areas (top), and dynamics of interaction between PFC and BG in updating working memory.

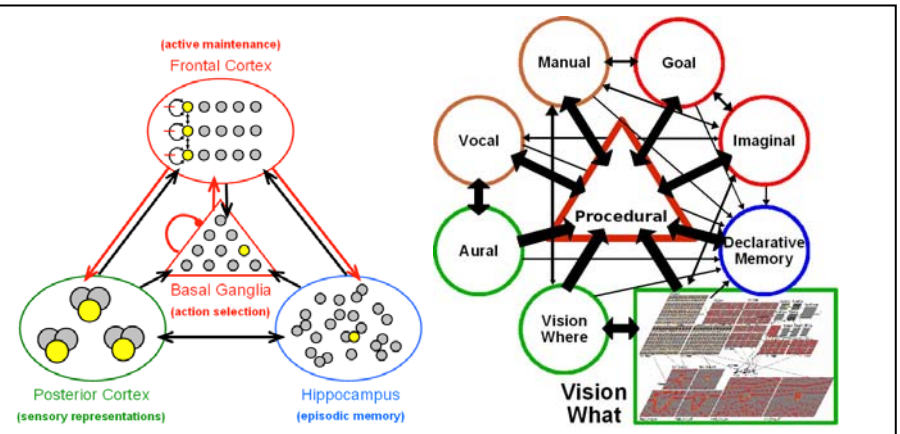


Figure 4. **S**ynthesis of **ACT**-R and **L**eabra (SAL) – Leabra brain-scale architecture (PBWM plus hippocampus) (left) has same general structure as ACT-R system (right) – right panel also shows one version of SAL where ACT-R provides control over a Leabra-based visual model. Also in progress are more synthetic models combining best features from each architecture.

systems that play a central role in *executive control* – our ability to overcome habitual or inappropriate behaviors, and "stay on task". In effect, the PFC/BG system is crucial for enabling our sense of "free will" – without these systems, behavior and thoughts become almost entirely driven by the immediate environment, and people lose the ability to initiate new actions based on internal plans and goals. The PFC/BG system has unique neural circuitry that enables it to function somewhat like a computer logic gate, where *control signals* can operate on *content signals* in a systematic fashion. This, combined with powerful reinforcement-learning mechanisms based on the neuromodulator dopamine, enables the system to learn to operate according to internal rules and plans, and to maintain elaborate internal context that renders behavior more independent of immediate environmental influences.

Around the time that eCortex was formed, we began a collaboration with the ACT-R group at Carnegie Mellon University to develop a **S**ynthesis of **ACT**-R and **L**eabra (SAL), which integrates the best ideas from both frameworks (Figure 4). ACT-R is a popular cognitive modeling environment and architecture that is cast at a higher level of abstraction than Leabra, and can readily perform abstract cognitive tasks like solving algebra problems or operating complex equipment like airplanes or air traffic control stations. Nevertheless, ACT-R reflects a remarkably similar conception of the overall cognitive architecture, in particular with respect to the function of the PFC/BG system and reinforcement learning, so there is great potential for synthesis and cross-fertilization from these different levels of analysis.

The ultimate goal of this collaboration is to develop a system that has the more robust and fine-grained learning mechanisms of Leabra, with the higher-level planning and execution abilities of ACT-R. Given the widespread adoption of ACT-R for practical applications in many arenas, from military to education, this could be an important development.

eCortex is positioned to commercialize components of this research as it transitions to the application stage. The company's efforts center around commercialization opportunities and applications of the research performed in the CCN lab. Application-oriented work often requires a broader set of skills and more complex management and organization than can be realistically accomplished in a research laboratory. Furthermore, in the short run, applications can be a distraction to the deeper scientific research efforts. Nevertheless, lessons learned with models in application areas feed into the lab's research projects at appropriate intervals and inform the research. The true test of a model and modeling approach is whether it works, and applications provide a powerful test environment to that end.

Contact Information

David J. Jilk,
CEO, eCortex, Inc.:
djilk@e-cortex.com

Dr. Randall C. O'Reilly,
CTO, eCortex, Inc. and
head of CCN Lab:
randy.oreilly@colorado.edu

http://www.e-cortex.com

Recommender Systems in the Web 2.0 Sphere

BY DIETMAR JANNACH AND MARKUS ZANKER

Recommender Systems (RS) are software solutions that help users deal with the information overload and find the information they need. From a technical perspective, RS have their origins in different fields such as information filtering and data mining and are built using a broad array of statistical methods and algorithms. Since their beginnings in the early 1990s, boosted by the enormous growth of the Web, RS have been widely applied in e-commerce settings, with recommendations from the book e-tailer Amazon.com being probably the most prominent example [4]. Systems like this point an individual online visitor to additional interesting items, usually by analyzing their shopping behavior or that of the wider user community. The success of RS is based on the fact that personalized item recommendations can measurably increase overall sales figures on e-commerce sites as shown by a recent study [1], [2].

Fig. 1. Pasadena City Hall (ITWP'09)

The collaborative recommendation paradigm itself can be seen as one of the pioneering applications of what has now come to be known as "Web 2.0" where users are not only consumers of information but also contribute in a democratic way and actively shape the Web by themselves. Thus, the metaphor of a *Social Web* commonly stands for (participatory) media applications like blogs, Wikipedia and other forms of content annotation and sharing as well as social and trust-based networks. In a similar way, collaborative filtering applications are about sharing opinions on items and benefitting from the ratings and recommendations of other users in a community. Consequently, the abundance of user-generated data that is now available impacts current recommender systems research and practices in a number of different ways. Consider for instance that the prediction accuracy of any RS mostly depends on the amount and quality of information the system has about the customer. In the worst case, the system only has information about the customer's previous purchases or ratings and this number of purchases might be rather low. In the Web 2.0 era, however, more information about customers like demographics or social relationships may be available which can be exploited by a RS. In Social Web platforms for example, users are often rather willing to reveal personal information, e.g., about their hobbies, book or film preferences or favorite Web sites. Furthermore, such networks consist of explicit *trust* relationships between users that can be relevant for the recommendation process. Note that in addition to more in-depth information about users, more information about the items themselves is also available. Users are often willing to write detailed product reviews or add comments to bookmarks or tag resources, thus providing data that can be exploited by content-based recommendation mechanisms.

In addition to aspects surrounding the exploitation of additional knowledge sources, Web 2.0 also opens new application opportunities for RS technology. While typical Web 2.0 content such as blogs or bookmarks can be recommended to users with the help of classical RS algorithms, the recommendation of contacts on a Social Web platform or the recommendation of tags for resources often requires the development of new approaches. Viewed more generally, there seem to be many opportunities where RS can help to stimulate participation and sustained membership in Social Web applications.

The recommender systems research community is currently very active. Aside from the different workshops held at major conferences, the newly established ACM Conference on Recommender Systems has already received nearly 200 paper submissions in 2009. In general, recommendation in the Web 2.0 sphere is one of the major topics at all events related to recommender systems research. In this report, we will summarize the issues that were discussed in this context on two focused workshops held in 2009.

I. ITWP'09

The one-day workshop on *Intelligent Techniques for Web Personalization & Recommender Systems* was held on July 11 at IJCAI'09 in Pasadena and was the seventh in a series of successful events held at major Artificial Intelligence conferences since 2001. At this workshop, Recommender Systems were discussed in the context of the more general problem of Web personalization and therefore viewed as a special way of tailoring the Web experience to individual users. Overall, the main goals of the workshop were to bring together people from the different fields, foster the exchange of ideas and discuss current topics in the area.

Paper submissions from 15 different countries were received out of which less than 40% were accepted for full presentation at the workshop. The workshop was organized in four technical sessions in which the seven full papers and the three short papers were presented. With respect to Web 2.0 recommender systems, recent research results were presented in particular in the area of intelligent tag recommendation in folksonomies, the simultaneous exploitation of different information sources for a given recommendation task and the in-

corporation of social and semantic information in a hybrid recommender system.

In addition to the technical paper presentations, the workshop also featured an invited talk given by Barry Smyth of University College Dublin. In his talk on "Personalization and Collaboration in Social Search" Barry Smyth focused on the HeyStaks [5] collaborative Web search system in which users are connected in a social network and can recommend interesting search results to each other. Overall, his talk demonstrated the opportunities for combining approaches from Web search, personalization, recommender systems and the Social Web.

The 2009 workshop ended with an open discussion on current challenges and future developments in the field.

II. RSWEB'09

Due to the increasing interest in recommender systems in the Web 2.0 sphere[1], this year's ACM Recommender Systems conference program also included a dedicated workshop on Recommender Systems & the Social Web (RSWEB'09) for the first time. The workshop was held on October 25 in New York and received more than 20 submissions from 10 different countries and accepted around 50% for full presentation.

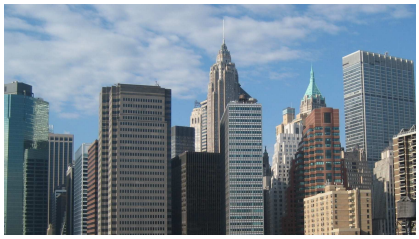


Fig. 2. New York skyline (RSWEB'09).

The one-day workshop consisted of both technical paper sessions, which were scheduled in a way that allowed ample time for discussions, as well as of more informal break-out sessions for brainstorming and discussion on specific subtopics. The papers submitted to the workshop covered a variety of topics in the context of Social Web recommender systems, which can be grouped into the following broad categories.

- Trust-based recommendation: Using trust-statements and explicit relationships in social networks to find similar neighbors and improve recommendation accuracy and coverage.
- Issues in tag recommendations: Improving recommendation accuracy through graph-based and hybrid algorithms; generating appropriate tags from document content.
- Web 2.0 content: Recommendation on social media sites; knowledge-based preference elicitation.

The breakout sessions were devoted to topics such as "What kind of additional knowledge can be leveraged to make recommendations more accurate?" or "To which problems of Web 2.0 and Social Web systems can recommender systems technology be applied and how?". Overall, the workshop raised strong interest in the research community leading to the situation that not all requests for invitations to the workshop could be satisfied. The timeliness of the topic was also demonstrated by the fact that also at the main conference more than a fourth of the accepted long papers dealt with recommender systems technology in the context of the Social Web.

To summarize, the question of how recommender systems technology can be applied to and is influenced by the developments in Web 2.0, the Social Web and also the Semantic Web is one of the main topics in recommendation research in 2009 and will also continue to be so for the coming years. In that context, the ITWP and RSWEB workshops served as an inspiring platform for the exchange of ideas and discussion among researchers working on all aspects of recommender systems in the Web 2.0 era and should be repeated in 2010.

REFERENCES

[1] M. Benjamin Dias, Dominique Locher, Ming Li, Wael El-Deredy, and Paulo J.G. Lisboa. The value of personalised recommender systems to e-business: a case study. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 291–294, Lausanne, Switzerland, 2008.

[2] Dietmar Jannach and Kolja Hegelich. A case study on the effectiveness of recommendations in the mobile internet. In *Proceedings of the 3rd ACM Recommender Systems Conference*, New York, NY, USA, pages 205–208, 2009.

[3] Dietmar Jannach, Markus Zanker, and Joseph A. Konstan. Special issue on recommender systems. *AI Communications*, 21(2-3):95–96, 2008.

[4] G. Linden, B. Smith, and J. York. Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.

[5] Barry Smyth, Peter Briggs, Maurice Coyle, and Michael O'Mahony. Google shared. a case-study in social search. In *Proc. 17th International Conference on User Modeling, Adaptation, and Personalization*, pages 283–294, 2009.

Dietmar Jannach is a professor in Computer Science at TU Dortmund, Germany and chair of the e-Services Research Group. His main research interests lie in the application of artificial intelligence and knowledge-based systems technology to real-world problems in particular in e-business environments. He has authored numerous papers on intelligent sales support systems such as recommender systems or product configurators. Dietmar Jannach was also one of the co-founders of ConfigWorks GmbH, a company focusing on next-generation interactive recommendation and advisory systems. He was a co-chair and organizer of the ITWP workshop at IJCAI'09 and the ACM RecSys'09 Workshop on RS and the Social Web.

Markus Zanker is an assistant professor at the Department for Applied Informatics at the University of Klagenfurt, Austria and is a co-founder and director of ConfigWorks GmbH, a provider of interactive selling solutions. He received his MS and doctorate degree in computer science and MBA in business administration from Klagenfurt University. His research interests lie in the area of knowledge-based systems, in particular in the fields of interactive sales applications such as product configuration and recommendation. Markus Zanker will be a program co-chair at the 4^{th} ACM Conference on Recommender Systems to be held in Barcelona, Spain in 2010.

[1]See also the recent AI Communications Special Issue on Recommender Systems [3] that featured several papers on this topic.

Behavior Informatics: An Informatics Perspective for Behavior Studies

Longbing Cao, *Senior Member, IEEE* and Philip S. Yu, *Fellow, IEEE*

***Abstract*—Behavior is increasingly recognized as a key entity in business intelligence and problem-solving. Even though behavior analysis has been extensively investigated in social sciences and behavior sciences, in which qualitative and psychological methods have been the main means, nevertheless to conduct formal representation and deep quantitative analysis it is timely to investigate behavior from the informatics perspective. This article highlights the basic framework of *behavior informatics*, which aims to supply methodologies, approaches, means and tools for formal behavior modeling and representation, behavioral data construction, behavior impact modeling, behavior network analysis, behavior pattern analysis, behavior presentation, management and use. Behavior informatics can greatly complement existing studies in terms of providing more formal, quantitative and computable mechanisms and tools for deep understanding and use.**

***Index Terms*—Behavior, Behavior Informatics.**

I. Introduction

While behavior has been intensively studied in social sciences and behavioral sciences, the current research methodologies and approaches are derived mainly from the social and psychological aspects. Behavioral sciences [7], [11], [9] abstract empirical data to investigate the decision processes and communication strategies within and between organisms in a social system [2]. This involves fields like psychology and social neuroscience (psychiatry), and genetics among others. Qualitative analysis and experiments followed by psychological explanation and reasoning are mainly conducted on human and animal behavior.

Behavioral sciences include two broad categories [2]: neural-decision sciences and social-communication sciences. Decision sciences involve those disciplines primarily dealing with the decision processes and individual functioning used in the survival of an organism in a social environment. These include psychology, cognitive science, organization theory, psychobiology, management science, operations research (not to be confused with business administration) and social neuroscience. On the other hand, communication sciences include those fields which study the communication strategies used by organisms and the dynamics between organisms in an environment. These include fields like anthropology, organizational behavior, organization studies, sociology and social networks.

This work is sponsored in part by Australian Research Council Discovery Grants (DP0988016, DP0773412, DP0667060) and ARC Linkage Grant (LP0989721, LP0775041).

Longbing Cao is with the Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia. E-mail: lbcao@it.uts.edu.au. Philip S Yu is with the Department of Computer Science, University of Illinois at Chicago. E-mail: psyu@cs.uic.edu.

With the emergence of new behavioral data, for instance, web usage, vehicle movements, market dynamics, ubiquitous transactional data recorded in computerized software systems, and agentized behavior, behavioral data including human behavior is largely electronically recorded. Behavioral sciences cannot support the formal representation and deep understanding of such behavioral data.

With the increasing needs and focus on social network analysis and social computing, it is very timely to develop behavior representation and analysis from the informatics perspective. Behavior informatics (including analytics, BI or BIA) is proposed for and aimed at the development of effective methodologies, approaches, tools and applications for formal and quantitative behavior representation and modeling, and deep analysis of behavior networks, impacts and patterns. This differentiates the aims and tasks of behavior informatics from those of behavioral sciences. This article outlines the area of behavior informatics. Behavior informatics [1] has the potential for designing and supplying new and practical mechanisms, tools and systems for deep behavior understanding and use. This will greatly complement behavioral sciences and behavior studies in social sciences. It can be widely used in many areas and domains, including understanding the Internet network, human community behavior and its evolution in the Internet, the deep understanding of human, animal, agentized and computerized organism behavior, and in widespread domains such as counter-terrorism, crime prevention, network analysis, intrusion detection, fraud and risk control, intelligent transport systems, trading agents, market dynamics, e-commerce, and financial transactions.

In fact, many researchers have started to develop deep analysis techniques for understanding behavior-related data in relevant domains. Typical examples include sequence analysis [15], event mining, crime mining, and activity mining [3], [4] and monitoring [8], as well as specific methods proposed to handle intrusion detection [13], fraud detection, outlier detection, customer relationship management [10], web usage mining [12], and so on. Behavior informatics is a scientific field consolidating these efforts and further studies on open issues toward a systematic and rigorous formalization and mechanism for behavior representation, analysis, presentation and use. With the power of behavior informatics, many traditional methods and domains can be further investigated from the behavioral perspective. In [14], facial behavioral data is analyzed, combined with facial expression information, which has shown great opportunities for expanding facial recognition capabilities and performance by considering facial behavior. [5] further reports the use of behavior informatics

in deeply analyzing microstructure-based trading behavior in stock markets, which has demonstrated very impressive advantages compared to traditional methods in understanding low-level driving forces of exceptional market dynamics. The remainder of this article is organized as follows. Section II describes what behavior informatics is. Section III argues why we need behavior informatics. The theoretical underpinnings are discussed in Section IV. Section V lists various research issues related to behavior informatics. We conclude this paper in Section VI.

II. What Is Behavior Informatics?

Behavior Informatics is a scientific field which aims to develop methodologies, techniques and practical tools for representing, modeling, analyzing, understanding and/or utilizing symbolic and/or mapped behavior, behavioral interaction and networking, behavioral patterns, behavioral impacts, the formation of behavior-oriented groups and collective intelligence, and behavioral intelligence emergence. In more detail, behavior informatics addresses the following key aspects.

- Behavioral data: In preparing behavioral data, behavioral elements hidden or dispersed in transactional data need to be extracted and connected, and further converted and mapped into a behavior-oriented feature space, or *behavioral feature space*. In the behavioral feature space, behavioral elements are presented in behavioral itemsets. Figure 1 illustrates the mapping and conversion from transactional data to behavioral data.
- Behavioral representation and modeling: The goal is to develop behavior-oriented specifications for describing behavioral elements and the relationships amongst the elements. The specifications reshape the behavioral elements to suit the presentation and construction of behavioral sequences. Behavioral modeling also provides a unified mechanism for describing and presenting behavioral elements, behavioral impact and patterns.
- Behavioral impact analysis: For analyzing behavioral data, we are particularly interested in those behavioral instances that are associated with having a high impact on business processes and/or outcomes. Behavioral impact analysis features the modeling of behavioral impact.
- Behavioral pattern analysis: There are in general two ways of conducting behavioral pattern analysis. One is to discover behavioral patterns without the consideration of behavioral impact, the other is to analyze the relationships between behavior sequences and particular types of impact.
- Behavioral intelligence emergence: To understand behavioral impact and patterns, it is important to scrutinize behavioral occurrences, evolution and life cycles, as well as the impact of particular behavioral rules and patterns on behavioral evolution and intelligence emergence (for instance, the emergence of swarm intelligence from a group of interactive agents). An important task in behavioral modeling is to define and model behavioral rules, protocols and relationships, and their impact on behavioral evolution and intelligence emergence.

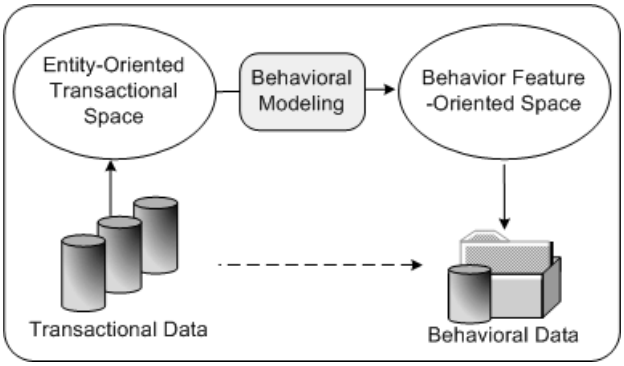


Fig. 1. From Transactional Data to Behavioral Data

- Behavioral network: Multiple sources of behavior may form into certain behavioral networks. Particular human behavior is normally embedded into such a network to fulfill its roles and effects in a particular situation. Behavioral network analysis seeks to understand the intrinsic mechanisms inside a network, for instance, behavioral rules, interaction protocols, convergence and divergence of associated behavioral itemsets, as well as their effects such as network topological structures, linkage relationships, and impact dynamics.
- Behavioral simulation: To understand all the above mechanisms that may exist in behavioral data, simulation can play an important role for observing the dynamics, the impact of rules/protocols/patterns, behavioral intelligence emergence, and the formation and dynamics of social behavioral networks.
- Behavioral presentation: From analytical and business intelligence perspectives, behavioral presentation seeks to explore presentation means and tools that can effectively describe the motivation and interest of stakeholders on the particular behavioral data. Besides the traditional presentation of patterns such as associations, visual behavioral presentation is a major research topic, and it is of high interest to analyze behavioral patterns in a visual manner.

In essence, the purpose of Behavior Informatics is to deliver technologies and tools for understanding behavior and social behavior networks. In this sense, we also call it *behavioral computing*.

III. Why Behavior Informatics?

First of all, deep and quantitative behavior analysis cannot be supported by methodologies and techniques in traditional behavioral sciences. In understanding and solving many issues and problems, *behavior* emerges as a key component, in both artificial societies (such as computerized business-support systems) and human societies. Behavior connects to many entities and objects in businesses, such as business objects, behavior subjects and objects, causes, impacts, scenarios and constraints. In addition, multiple relevant behavior instances make up a social behavior network, which involves social and organizational factors, and collective intelligence. Therefore, it is highly likely that behavior-oriented analysis can provide extra information, in particular regarding interior principles,

causes and impact about the formation and movement of exterior business objects and appearances.

In current business management information systems, the above behavior-related factors are normally hidden in transactional data. Transactional data is usually entity-oriented, and entities are connected through keys, which form a *transactional entity space*. In such transactional entity spaces, behavioral elements are dispersed and hidden in multiple transactions with weak or no direct linkages. An example would be the trading transactions recorded in stock markets, in which an investor's trading behaviors, such as buy quote, sell quote, trade, withdrawal etc., are separately recorded into different tables, while they are actually closely related to each other. We certainly lose the full picture of an investor's overall behavior if we only look at any single aspect of them rather than putting them together. Therefore, in general, behavior is *implicit* and often *dispersed* in transactional data. It is not effective to straightforwardly analyze the interior driving force of human behavior on normal transactional data. To effectively understand such driving forces, we need to make behavior *explicit* for further behavior-oriented pattern analysis. For the example of trading behavior, if we consider the coupling relationships amongst quotes, trades, withdrawals etc. regulated by trading rules and market mechanisms, and analyze the coupled multiple behavior sequences, it is very likely that we can generate a much more informative and natural picture of trading behaviors. For this purpose, we extract quotes, trades, withdrawals etc. behavioral elements, and their properties including timepoints, prices and volumes when we detect exceptional trading behavior [5], [6].

As addressed above, the presentation of behavioral data differentiates from that of normal transactional data. To effectively understand and analyze behavior and its impact, it is essentially important to squeeze out behavioral elements from transactions, and to map behavior-oriented elements in transactional data into a behavior-oriented feature space to form the behavioral data. Such extrusion and transformation from transactional space to behavioral space makes a behavior shift from implicit to explicit for more effective analysis of behavior patterns and impacts. To support the mapping from transactional space to behavioral space, it is vitally important to build formal methods and workable tools for behavior representation, processing and engineering, namely the sciences of Behavior Informatics. Even though general data preprocessing on behavior element-oriented data is helpful, it is not effective enough nor sophisticated enough to mine such data for explicit behavior patterns and impact. Straightforward behavioral data is expected in order to cater for behavior analytics smoothly. Further, to mine for behavior and impact patterns, new issues and corresponding techniques have to be addressed.

As a result, with the development of foundations and technical tools for behavior informatics, it is possible for us to understand and scrutinize business processes, problems and potential solutions from a perspective different from the traditional ones of target behavior and behavioral network perspective. In fact, due to the intrinsic integration of behavior and its subjects and objects, the in-depth understanding of behavior can actually promote a much deeper understanding of the roles and effects of comprehensive factors surrounding a business problem, for instance, human demographics, human actions, environment and behavioral impact. With such a capability, behavior informatics is likely to further expand the opportunities of problem-solving, and stimulate promising prospects. Behavior informatics can complement classic behavioral analytical methods. This makes it possible to more effectively understand, model, represent, analyze and utilize behavior and social behavior networks toward more comprehensive and effective problem solving and understanding. This includes but is not limited to behavior understanding, exceptional behavior analysis, taking advantage of opportunities, behavior pattern analysis, behavior impact analysis, and cause-effect analysis.

IV. Theoretical Underpinnings

Behavior Informatics is a multidisciplinary research field. Its theoretical underpinnings involve analytical, computational and social sciences as shown in Figure 2. We interpret the theoretical infrastructure for behavior informatics from the following perspectives: (1) Methodological support, (2) Fundamental technologies, and (3) Supporting techniques and tools. From the methodological support perspective, behavior informatics needs to draw support from multiple fields, including information sciences, intelligence sciences, system sciences, cognitive sciences, psychology, social sciences and sciences of complexities. Information and intelligence sciences provide support for intelligent information processing and systems. System sciences furnish methodologies and techniques for behavior and behavioral network modeling and system simulation, and the large scale of a behavior network. Cognitive sciences incorporate principles and methods for understanding human behavior belief, and the intention and goal of human behavior. Psychology can play an important role in understanding human behavior motivation and evolution. The social sciences supply foundations for conceiving the organizational and social factors and business processes that surround behavior and are embedded in behavior networks. Areas such as economics and finance are also important for understanding and measuring behavior impact. Methodologies from the science of complexities are essential for group behavior formation and evolution, behavior self-organization, convergence and divergence, and behavior intelligence emergence.

Fundamental technologies are necessary for behavioral modeling, pattern analysis, impact analysis, and behavior simulation. To support behavior modeling, technologies such as user modeling, formal methods, logics, representation, ontological engineering, semantic web, group formation and cognitive science are essentially important. They can not only represent behavioral elements, but also contribute to the mapping from the transactional entity space to the behavioral feature space. The modeling of behavior impact needs to refer to technologies in areas such as risk management and analysis, organizational theory, sociology, psychology, economics and finance. For the analysis of behavioral patterns, technologies such as data mining and knowledge discovery,

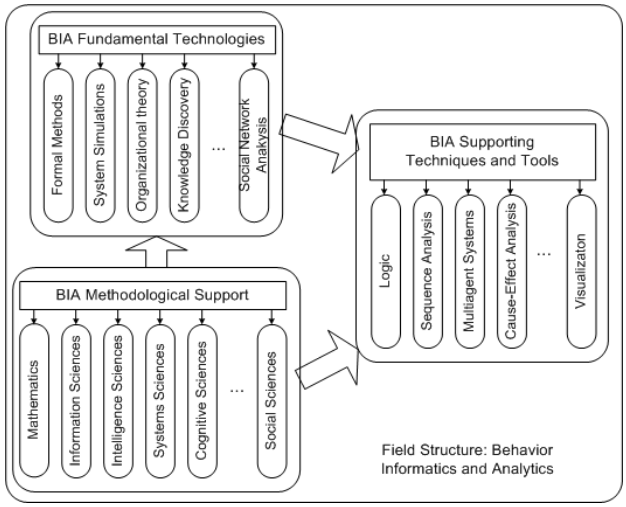


Fig. 2. Field Structure of Behavior Informatics

artificial intelligence and machine learning can contribute a great deal. In simulating behavior, behavioral impact and behavior networks, we refer to techniques and tools in fields like system simulation, artificial social system, open complex systems, swarm intelligence, social network analysis, reasoning and learning. The presentation of behavior evolution and behavior patterns can benefit from areas of visualization and graph theory. In addition, the scale and complexity related to behavioral data used to be a critical issue in social science studies. We now have the ability to collect a huge amount of data in a continuous fashion (just think about Facebook). An analogy is bioinformatics. Even the simulation can produce a large amount of data to predict behavior changes over a long period of time. The studies on complex sequence analysis can provide effective tools for handling complex behavior. Adaptive and active learning offers capabilities for dealing with behavior changes in a dynamic and online environment. From the operationalization aspect, behavior informatics needs to develop effective techniques and tools for representing, modeling, analyzing, understanding and/or utilizing behavior. This involves many specific approaches and means. For instance, several methods such as algebra and logics may be useful for modeling behavior. Behavior pattern analysis may involve many existing tools such as classification and sequence analysis, as well as the development of new approaches. To simulate behavior impact, one may use agent-based methods for cause-effect analysis, while for presenting behavior, visualization techniques may be useful.

V. Research Issues

As behavior informatics is at its beginning stage, many open issues are worthy of systematic investigation and case studies from aspects such as *behavioral data*, *behavior modeling and representation*, *behavioral impact analysis*, *behavioral pattern analysis*, *behavior presentation*, and *behavior simulation*. We further expand these by listing some key research topics for each of the above research issues, although certainly there may be other issues.

(1) *Behavioral Data*: In many cases, it may be necessary to convert normal transactional data into a behavior-oriented feature space, in which behavior elements consist of the major proportion of the dataset.
- Behavioral data modeling
- Behavioral feature space
- Mapping from transactional to behavioral data
- Behavioral data processing
- Behavioral data transformation

(2) *Behavior Modeling*: The building of behavior models will enable the understanding of interaction, convergence, divergence, selection, decision, and evolution of behavior sequences and behavior networks. To achieve this, modeling language, specifications and tools need to be developed to understand behavior dynamics.
- Behavior model
- Behavior interaction
- Collective behavior
- Action selection
- Behavior convergence and divergence
- Behavior representation
- Behavioral language
- Behavior dynamics
- Behavioral sequencing

(3) *Behavior Pattern Analysis*: This is the major focus of behavior informatics, namely to identify patterns in behavior sequences or behavior networks. For this, we need first to understand behavior structures, semantics and dynamics in order to further explore behavior patterns. We then need toinvestigate pattern analytical tasks such as detection, prediction and prevention through approaches like correlation analysis, linkage analysis, clustering and combined pattern mining.
- Emergent behavioral structures
- Behavior semantic relationship
- Behavior stream mining
- Dynamic behavior pattern analysis
- Dynamic behavior impact analysis
- Visual behavior pattern analysis
- Detection, prediction and prevention
- Customer behavior analysis
- Behavior tracking
- Demographic-behavioral combined pattern analysis
- Cross-source behavior analysis
- Correlation analysis
- Social networking behavior
- Linkage analysis
- Evolution and emergence
- Behavior clustering
- Behavior network analysis
- Behavior self-organization
- Exceptions and outlier mining

(4) *Behavior Simulation*: Simulation can play an essential role in the deep understanding of behavior working mechanisms, interaction amongst behavior instances, dynamics and the formation of behavior group and behavior intelligence emergence, etc. For example, simulation

can be conducted on large-scale behavior networks, convergence and divergence, evolution and adaptation of behavior through setting up artificial and computation-oriented behavior systems.

- Large-scale behavior network
- Behavior convergence and divergence
- Behavior learning and adaptation
- Group behavior formation and evolution
- Behavior interaction and linkage
- Artificial behavior system
- Computational behavior system
- Multi-agent simulation

(5) *Behavior Impact Analysis*: Behavior that has a high impact on business is our major interest. To analyze the behavior impact, techniques such as impact modeling, measurements for risk, cost and trust analysis, the transfer of behavior impact under different situations, exceptional behavior impact analysis would be very helpful. The analytical results will be utilized for detection, prediction, intervention and prevention of negative behavior or for opportunity use if positive cases are identified.

- Behavior impact analysis
- Behavioral measurement
- Organizational/social impact analysis
- Risk, cost and trust analysis
- Scenario analysis
- Cause-effect analysis
- Exception/outlier analysis and use
- Impact transfer patterns
- Opportunity analysis and use
- Detection, prediction, intervention and prevention

(6) *Behavior Presentation*: The presentation of dynamics of behavior and behavior networks in varying aspects would assist with the understanding of behavior lifecycle and impact delivery; for instance, rule-based behavior presentation, visualization of behavior network, and visual analysis of behavior patterns.

- Rule-based behavior presentation
- Flow visualization
- Sequence visualization
- Parallel visualization
- Dynamic group formation
- Dynamic behavior impact evolution
- Visual behavior network
- Behavior lifecycle visualization
- Temporal-spatial relationship
- Dynamic factor tuning, configuration and effect analysis
- Behavior pattern emergence visualization
- Distributed, linkage and collaborative visualization

Figure 3 further illustrates major research tasks/approaches and the relations among the above key research components. Behavioral data is extracted from behavior-relevant applications, and then converted into behavioral feature space. When the behavioral data is ready, behavior pattern analysis and impact analysis are conducted on the data. To support behavior pattern analysis and impact analysis effectively, behavior simulation and modeling can provide fundamental results about behavior dynamics and relevant businesses and tools for knowledge discovery. Besides supplying another point of view for behavior analysis, behavior presentation contributes techniques and means to study behavior.

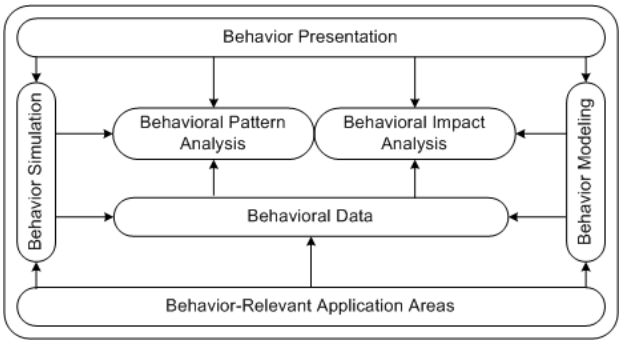


Fig. 3. Research Map of Behavior Informatics

VI. Conclusion

Behavioral sciences mainly explore the activities of and interactions among humans and animals in the natural world. For this study, qualitative, empirical, experimental and psychological methodologies and tools are generally used. With the increasing emergence of computerized and agentized behavioral data, behavioral sciences do not provide such methodologies, methods and means for formal representation and reasoning, or deep and quantitative analysis of behavior networks, impacts and patterns, from either individual or group perspectives. For this purpose, behavior informatics is proposed. Behavior informatics is essential for dealing with many behavior-related problems crossing widespread domains and areas. Typical driving forces come from Internet networks and activities, financial market dynamics, e-commerce and online businesses, human community activities and interactions, and customer relationship management.

This article highlights the framework of behavior informatics, explaining its main concepts, driving forces, theoretical underpinnings, and research issues. As a new and promising field, great efforts are expected to follow on every aspect, from formal modeling, pattern analysis, impact analysis, network analysis, behavior presentation, to behavior management and use, from fundamental, technical and practical perspectives.

References

[1] www.behaviorinformatics.org, www.behavioranalytics.org.
[2] http://en.wikipedia.org/wiki/behavioral_sciences
[3] Cao, L., Zhao, Y. and Zhang, C. Mining impact-targeted activity patterns in imbalanced data, IEEE Trans. on Knowledge and Data Engineering, Vol. 20, No. 8, pp. 1053-1066, 2008
[4] Cao, L., Zhao, Y., Zhang, C. and Zhang, H. Activity mining: from activities to actions, International Journal of Information Technology & Decision Making, 7(2), pp. 259 - 273, 2008.
[5] Cao, L. and Ou, Y. Market microstructure patterns powering trading and surveillance agents. Journal of Universal Computer Sciences, 2008 (to appear).
[6] Cao, L., Ou, Y., Luo, D. and Zhang, C. Adaptive detection of abnormal behavior in multiple trading activity streams, Technical Report, 2008.

[7] Devereux, G. From anxiety to method in the behaviorial sciences, The Hague, Mouton & Co, 1967.
[8] Fawcett, T. and Provost, F. Activity monitoring: noticing interesting changes in behavior, KDD'99, 53 - 62.
[9] Klemke, E., Hollinger, R. and Kline, A. (eds) Introductory readings in the philosophy of science. Prometheus Books, New York, 1980.
[10] Mozer, M., Wolniewicz, R., Grimes, D., Johnson, E. and Kaushanky, H. Predicting subscriber dissatisfaction and improving retention in wireless telecommunications industry. IEEE Transactions on Neural Networks, 11, pp. 690-696, 2000.
[11] Smelser, N. and Baltes, P. (eds) International encyclopedia of the social & behavioral sciences, 26 v. Oxford, Elsevier, 2001.
[12] Srivastava, J., Cooley, R., Deshpande, M. and Tan, P. Web usage mining: discovery and applications of usage patterns from Web data, ACM SIGKDD Explorations Newsletter, v.1 n.2, 2000.
[13] Stolfo, S., Hershkop, S., Hu, C., Li, W., Nimeskern, O. and Wang, K. Behavior-based modeling and its application to Email analysis, ACM Transactions on Internet Technology (TOIT), 6(2), 187-221, 2006.
[14] Tsai, P., Cao, L., Hintz, T. and Jan, T. A bi-modal face recognition framework integrating facial expression with facial appearance, Pattern Recognition Letter, 30(12): 1096-1109, 2009.
[15] Weiss, G. and Hirsh, H. Learning to predict rare events in event sequences, KDD-98, pp. 359-363, 1998.

Adaptive Anomaly Detection of Coupled Activity Sequences

Yuming Ou, Longbing Cao and Chengqi Zhang

***Abstract*—Many real-life applications often involve multiple sequences, which are coupled with each other. It is unreasonable to either study the multiple coupled sequences separately or simply merge them into one sequence, because the information about their interacting relationships would be lost. Furthermore, such coupled sequences also have frequently significant changes which are likely to degrade the performance of trained model. Taking the detection of abnormal trading activity patterns in stock markets as an example, this paper proposes a Hidden Markov Model-based approach to address the above two issues. Our approach is suitable for sequence analysis on multiple coupled sequences and can adapt to the significant sequence changes automatically. Substantial experiments conducted on a real dataset show that our approach is effective.**

***Index Terms*—Multiple coupled sequences, Anomaly, HMM, Adaptation, Stock market.**

I. Introduction

TYPICAL sequence analysis [4], [9], [7], [1], [10] mainly focuses on identifying patterns on one sequence. However, dealing with the real-life problems, we often have to face multiple interacting sequences rather than only one single sequence. For example, in stock markets there are three coupled sequences including buy orders, sell orders and trades by matching orders from both buy and sell sides. These three sequences are coupled with each other in terms of many aspects such as timing, price and volume. The interacting relationships among them contain rich information which is very valuable to stock market surveillance. As price manipulators may deliberately place their buy orders and/or sell orders and indirectly affect the trade price through manipulating the interaction between them, the interaction is an important clue to identifying stock price manipulations. If we study the three sequences separately or simply merge them into one sequence, the valuable information about their interacting relationships would be of course lost.

In real-life applications, we also often face another issue that is the significant changes in sequences. For instance, the trading activities in stock markets change frequently due to the investors' sentiment and the external market environment, resulting in the potential significant changes in the three coupled sequences. Thus it is necessary for sequence analysis methods to identify the significant changes and adapt to the new environment.

Yuming Ou is with the Faculty of Engineering and Information Technology, University of Technology Sydney.

Longbing Cao is with the Faculty of Engineering and Information Technology, University of Technology Sydney.

Chengqi Zhang is with the Faculty of Engineering and Information Technology, University of Technology Sydney.

In this paper, we employ agent technology to develop a pattern mining system to detect abnormal trading activity patterns in the three coupled sequences including buy orders, sell orders and trades. The system uses six Hidden Markov Model(HMM)-based models to model the trading activity sequences in different ways: three standard HMMs for modeling single sequences respectively; an integrated HMM model combining all individual sequence-oriented HMMs; a Coupled HMM reflecting coupled relationships among sequences; and an Adaptive Coupled HMM to automatically capture the significant changes of activity sequences. The above six HMM-based models compete with each other. The outputs generated by the best model are used as the final outputs of system.

The rest of this paper is organized as follows. We present the system framework in Section II. After Section III introduces the modeling of trading activity sequences by HMM-based methods, Section IV provides the approach to identify abnormal activity patterns using six HMM-based models. The model selection and evaluation are introduced in Section V, and the experimental results are given in Section VI. Finally, Section VII concludes this paper.

II. Agent-Based Framework for Discovering Abnormal Patterns in Coupled Sequences

To make our system autonomous, we use agent technology to build the system. As shown in Figure 1, the system consists of the following main agents: Activity Extraction Agent, Anomaly Detection Agent, Change Detection Agent, Model Adjusting Agent, and Planning Agent. They collaborate with each other to find out the best model, and then deploy this best one for activity pattern discovery. In particular, to adapt to the source data dynamics, Change Detection Agent detects changes in the outputs of CHMM, and then the Planning Agent triggers the adjustment and retraining of the CHMM model (More details are introduced in Section III-C).

III. Modeling Activity Sequences by Hidden Markov Model-Base Methods

In this section, we first introduce the approaches to build Hidden Markov Model (HMM) [8], [5] for single activity sequence and Coupled Hidden Markov Model (CHMM) [2], [6] for multiple coupled activity sequences respectively, and then improve the CHMM by adding an automatically adaptive mechanism to it to create an Adaptive CHMM (ACHMM).

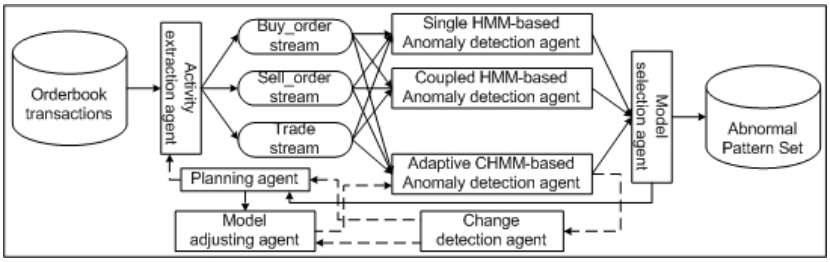


Fig. 1. Agent-Based Framework for Identifying Abnormal Activity Patterns

A. Modeling Single Activity Sequence by HMM

To model the single activity sequences including buy-order, sell-order and trade sequences separately, we build three models *HMM-B*, *HMM-S*, and *HMM-T* for them based on the standard HMM. The hidden states and observation sequences of the three models are defined as follows:

- In model *HMM-B*, the hidden states S^{buy} represent the investors' belief, desire and intention (BDI) on buy side, S^{buy} ={*Positive Buy, Neutral Buy, Negative Buy*}. In model *HMM-S*, the hidden states S^{sell} denote the investors' BDI on sell side, S^{sell} ={*Positive Sell, Neutral Sell, Negative Sell*}. In model *HMM-T*, the hidden states S^{trade} stand for the market states, S^{trade} ={*Market Up, Market Down*}. The exact values of the hidden states are unknown, while they can change from one to another with particular probabilities. For example, we cannot know the investors' BDI is *Positive Buy*, *Neural Buy* or *Negative Buy* actually.
- The observation sequences IA^{buy}, IA^{sell} and IA^{trade} stand for the activity sequences of buy-order, sell-order and trade respectively. The values of these activity sequences of buy-order, sell-order and trade can be observed. In the following, we will detail the method for constructing trading activity sequences.

The construction of trading activity sequences is based on two concepts: $activity$ (A) and $interval\ activity$ (IA), which involve human intention information including prices and volumes in stock markets.

Definition 1: Activity (A) is an action (a) and it is associated with BDI information (represented in p and v).

$$A = (a, p, v) \tag{1}$$

$$a = \begin{cases} buy\ order, & at\ time\ t \\ sell\ order, & at\ time\ t \\ trade, & at\ time\ t \end{cases} \tag{2}$$

$$p = \begin{cases} trade\ price, & of\ trade\ at\ time\ t \\ order\ price, & of\ buy\ or\ sell\ order\ at\ time\ t \end{cases} \tag{3}$$

$$v = \begin{cases} trade\ volume, & of\ trade\ at\ time\ t \\ order\ volume, & of\ buy\ or\ sell\ order\ at\ time\ t \end{cases} \tag{4}$$

Definition 2: Interval Activity (IA) represents the actions and BDI information associated with the activity sequence taking place during a window l (the window size is denoted by w).

$$IA_l = (A_l, P_l, V_l, W_l) \tag{5}$$

which is calculated as follows:

$$A_l = \{A_{l1}, A_{l2}, \ldots, A_{ln}\} \tag{6}$$

$$P_l = \frac{\sum_{i=1}^{n} p_i}{W_l} \tag{7}$$

$$V_l = \frac{\sum_{i=1}^{n} v_i}{W_l} \tag{8}$$

$$W_l = n \tag{9}$$

where n is the number of activities in the window l.

In stock markets, orders normally do not last for more than one day. Order are placed by investors after market opens and are expired after market closes if they have not been traded. Trades are also based on the orders placed on the same day only. This market mechanism indicates that all orders and trades on a same day are closely related. Thus we construct the sequences for buy order, sell order and trades respectively by grouping the IAs that fall into a same trading day together.

B. Modeling Multiple Coupled Activity Sequences by CHMM

In order to reflect the interacting relationship among the three activity sequences, we use a CHMM consisting of three chains of HMM to model the buy-order, sell-order and trade processes together. As shown in Figure 2, the circles denote the hidden states of the three processes while the squares stand for their observation sequences. The three chains are fully coupled with each other reflecting their interactions.

C. Adapting to Significant Activity Sequence Changes

In order to adapt the significant changes that often exist in trading sequences, we involve multi-agent technology to enhance the CHMM, and form an agent-based adaptive CHMM (ACHMM).

The adaptation of ACHMM is mainly based on the detection of change between the current outputs of model and the current benchmark. The current benchmark is defined as the outputs generated after the last update of model. Three agents contribute to the adaption: Change Detection Agent, Model Adjusting Agent and Planning Agent. The Change Detection Agent checks whether there is a significant difference between the current outputs and the current benchmark based on statistical test methods, for instance, t test. The significant

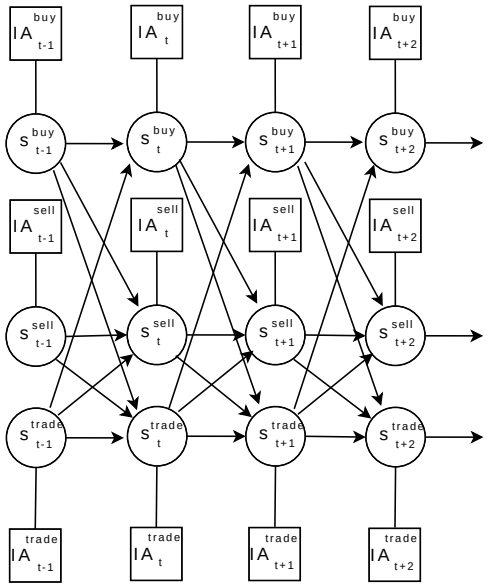


Fig. 2. CHMM for Modeling Multiple Coupled Activity Sequences

difference suggests that there is a change in the trading activities and the CHMM cannot model the trading activities properly, therefore the model needs to be updated. Once the change is detected, the Planning Agent will receive a notice and trigger the Model Adjusting Agent to retrain the model. The outputs generated after this update are the new current benchmark.

IV. Identifying Abnormal Trading Activity Sequences

After the models discussed above are trained, they can be used to identify abnormal trading activity sequences. The basic idea is to calculate the distances from the test sequences to the centroid of model. If the distance of a sequence is larger than a user-specified threshold, then the sequence is considered to be abnormal.

The formulas to compute the centroid (μ) and radius (σ) of model $\mathcal{M} \in \{$*HMM-B, HMM-S, HMM-T, IHMM, CHMM, ACHMM*$\}$ are as follows:

$$\mu = \frac{\sum_{i=1}^{K} Pr(Seq_i|\mathcal{M})}{K} \tag{10}$$

$$\sigma = \sqrt{\frac{1}{K}\sum_{i=1}^{K} Pr(Seq_i|\mathcal{M})) - \mu} \tag{11}$$

where Seq_i is a training sequence and K is the total number of training sequences.

The distance $Dist_i$ from a test sequence $Seq_i^{'}$ to model $\mathcal{M}$ is calculated by the following formula:

$$Dist_i = \frac{\mu - Pr(Seq_i^{'}|\mathcal{M})}{\sigma} \tag{12}$$

Consequently, $Seq_i^{'}$ is an abnormal sequence, if it satisfies:

$$Dist_i > Dist_{max} \tag{13}$$

where $Dist_{max}$ is a given threshold.

V. Model Selection and Evaluation

There are six HMM-based models for modeling the trading activity sequences in the system, including: 1) *HMM-B*: an HMM on buy-order sequences only; 2) *HMM-S*: an HMM on sell-order sequences only; 3) *HMM-T*: an HMM on trade sequences only; 4) *IHMM*: an integrated HMM combining *HMM-B*, *HMM-S* and *HMM-T*. The probability of *IHMM* is the sum of the probability values of the three models. This model does not consider the interactions among the three processes; 5) *CHMM*: a Coupled HMM for trade, buy-order and sell-order sequences, considering their interactions; and 6) *ACHMM*: an Adaptive Coupled HMM which is able to adapt to the significant changes in sequences automatically.

The selection of the best model amongst these candidates is conducted by the Model Selection Agent. The selection policies conducted by the agent are as follows.

Policy 1 selectBestModel

Rule 1: Select the X ($X > 1$) best candidate models by evaluating the technical performance;

Rule 2: Select the best model from the X ($X > 1$) best models by checking business performance.

The technical performance evaluation of model is based on the following metrics: *accuracy, precision, recall, specificity*. These four technical metrics measure the quality of the models. Furthermore, we introduce a business metric widely used in capital markets to evaluate the business performance of model. This business metric is *return (R)* [3], which refers to the gain or loss for a single security or portfolio over a specific period. It can be calculated by

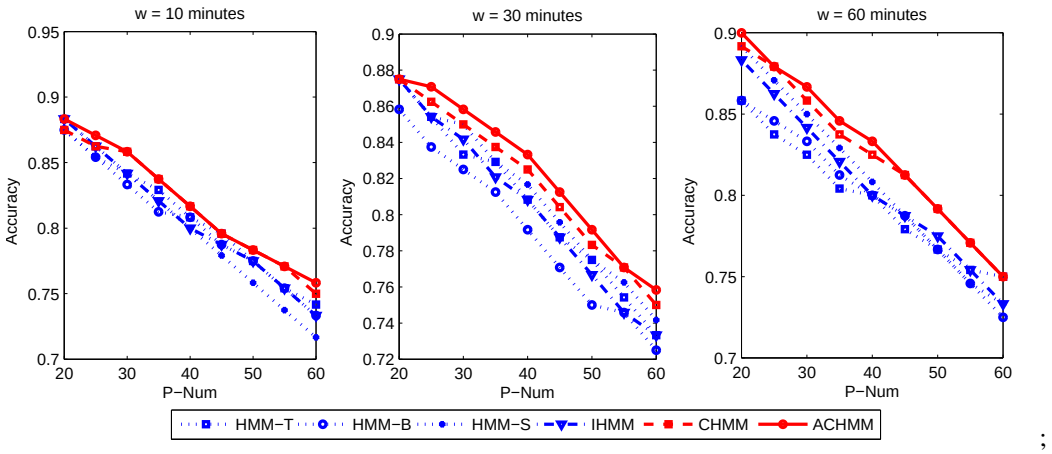
$$R = ln\frac{p_t}{p_{t-1}} \tag{14}$$

where p_t and p_{t-1} are the trade prices at time t and t-1, respectively. Empirically, the trading days with exceptional patterns are more likely to incur higher daily *return* than those without exceptional patterns.

VI. Experimental Results

Our system is tested on a real dataset from a stock exchange, which covers 388 trading days from June 2004 to December 2005. In the dataset, there are some trading days associated with alerts that are generated by the surveillance system used in that stock exchange. These alerts can be used to label the data, that is, the data with alerts is labelled as true anomalies. After labelling the data, we divide the whole dataset into two parts: one is for training models and another is for testing models.

Model *HMM-B*, *HMM-S*, *HMM-T* and *IHMM* are trained by the standard Baum-Welch algorithm [8] respectively, while model *CHMM* and *ACHMM* are trained by the algorithm proposed in [2], which is similar to the Baum-Welch algorithm, respectively. After these six models are trained, they are tested on the test data. The Model Select Agent will choose the best model in terms of their technical and business performance as presented in Section V.



;

Fig. 3. Technical Performance of Six Systems: Accuracy

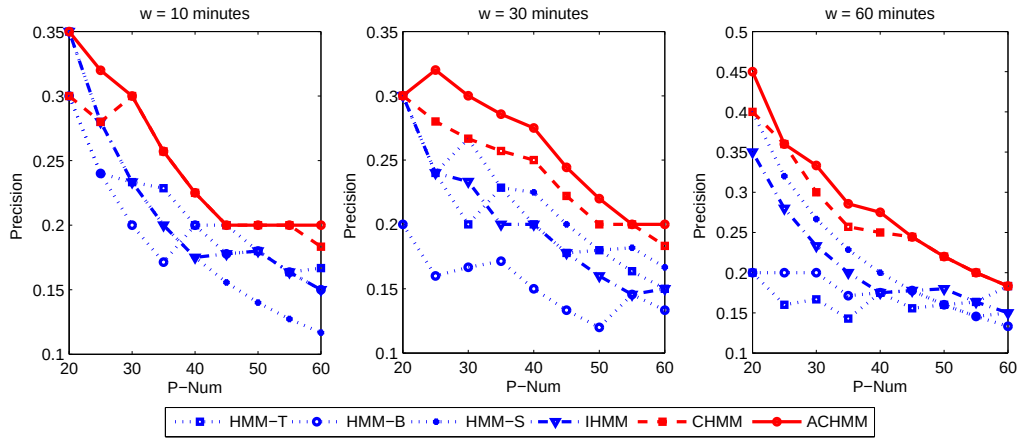


Fig. 4. Technical Performance of Six Systems: Precision

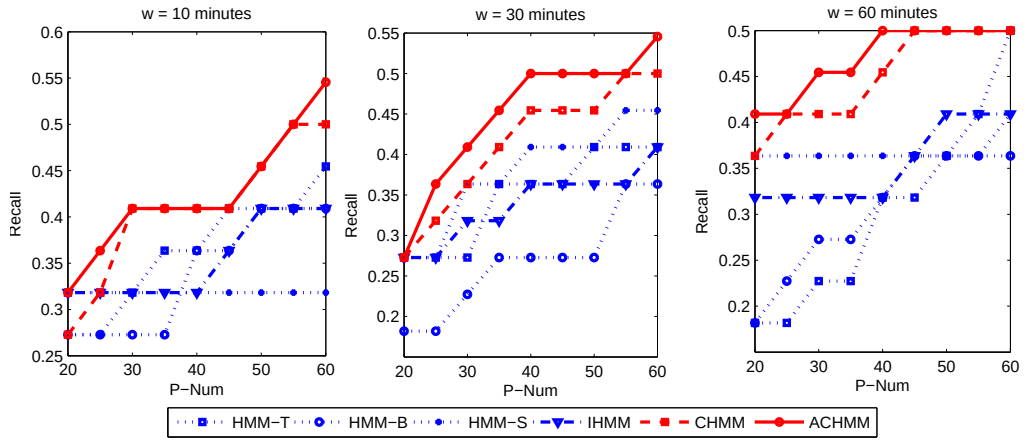


Fig. 5. Technical Performance of Six Systems: Recall

Figures 3, 4, 5 and 6 show the technical performance of the six models, where x axis (P-Num) stands for the number of detected abnormal activity patterns, and y axis represents the values of technical measures. Clearly *ACHMM* outperforms the other five models under different window sizes (w).

In terms of the business performance, Figure 7 shows the business performance of the six models, where y axis represents the values of average daily *return* of trading days in where abnormal activity patterns are detected. We can see that *ACHMM* also outperforms the other five models under different w.

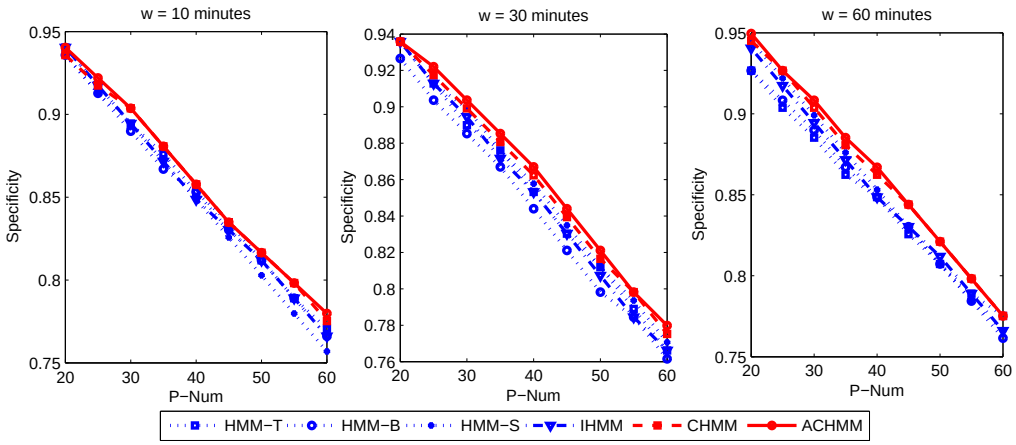


Fig. 6. Technical Performance of Six Systems: Specificity

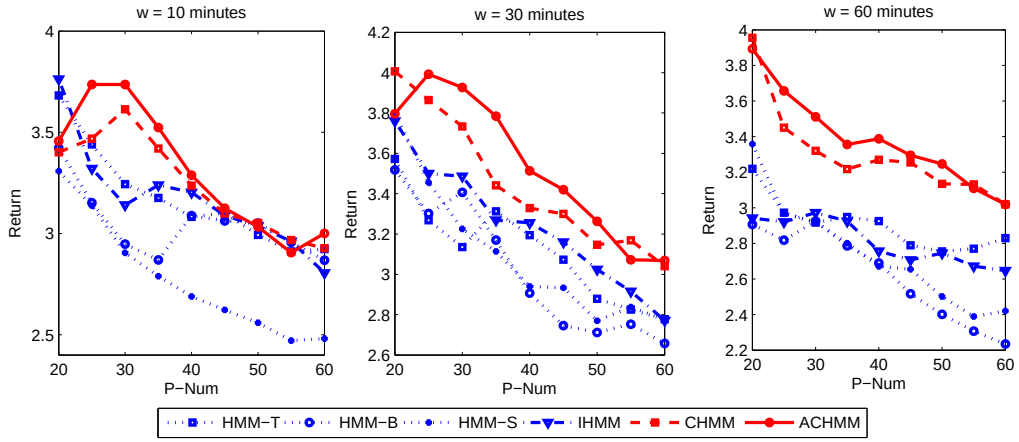


Fig. 7. Business Performance of Six Systems: Return

VII. Conclusion and Future Work.

Many real-life applications, such as detecting abnormal trading activity patterns in stock markets, often involve multiple coupled sequences. Typical existing methods mainly pay attention to only one single sequence. As the interacting relationships among the multiple coupled sequences contain valuable information, it is unreasonable to study the multiple coupled sequences separately by those existing methods. Furthermore, in practice such coupled sequences change frequently, which greatly challenges the trained model.

Taking the detection of abnormal trading activity patterns in stock markets as an example, this paper proposed a HMM-based approach to address the above two issues widely existing in real-life applications. Our approach caters for the sequence analysis on multiple coupled sequences and also can be used under the circumstances in which sequences change frequently. Substantial experiments conducted on a real dataset show that our approach is effective.

Our further work is on generalizing our approach for dealing with other application problems, investigating the update of existing sequence analysis methods for analyzing multiple coupled sequences, and comparing them with our HMM-based models.

References

[1] J. Ayres, J. Flannick, J. Gehrke and T. Yiu, *Sequential Pattern mining using a bitmap representation*, SIGKDD02, pp. 429–435, 2002.

[2] M. Brand, *Coupled hidden Markov models for modeling interacting processes*, Tech. Rep., MIT Media Lab, 1997.

[3] S. J. Brown and J. B. Warner, *Using daily stock returns: the case of event studies*, Journal of Financial Economics, vol. 14, pp. 3–31, 1985.

[4] G. Dongand and J. Pei, *Sequence Data Mining*, Springer, 2007.

[5] R. Durbin, S. R. Eddy, A. Krogh and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, 1999.

[6] N. M. Oliver, B. Rosario and A. P. Pentland, *A Bayesian computer vision system for modeling human interactions*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, pp.831–843, 2000.

[7] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M. C. Hsu, *Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth*, ICDE2001, pp. 215-226, 2001

[8] L. R. Rabiner, *A tutorial on hidden Markov models and selected applications in speech recognition*, Proceedings of the IEEE, vol. 77, pp. 275-286, 1989.

[9] R. Srikant and R. Agrawal, *Mining sequential patterns: Generalizations and performance improvements*, EDBT1996, pp. 3–17, 1996.

[10] M. J. Zaki, *SPADE: An efficient algorithm for mining frequent sequences*, Machine Learning, vol. 42, pp. 31–60, 2001.

Cellular Flow in Mobility Networks

Alfredo Milani, Eleonora Gentili and Valentina Poggioni

***Abstract*—Nearly all the members of adult population in major developed countries transport a GSM/UMTS mobile terminal which, besides its communication purpose, can be seen as a mobility sensor, i.e. an electronic individual tag. The temporal and spatial movements of these mobile tags being recorded allows their flows to be analyzed without placing costly ad hoc sensors and represents a great potential for road traffic analysis, forecasting, real time monitoring and, ultimately, for the analysis and the detection of events and processes besides the traffic domain as well. In this paper a model which integrates mobility constraints with cellular networks data flow is proposed in order to infer the flow of users in the underlying mobility infrastructure. An adaptive flow estimation technique is used to refine the flow analysis when the complexity of the mobility network increases. The inference process uses anonymized temporal series of cell handovers which meet privacy and scalability requirements. The integrated model has been successfully experimented in the domain of car accident detection.**

***Index Terms*—Mobile networks, spatial data mining, traffic flow analysis.**

Introduction and Related Work

THE basic laws governing human mobility are becoming an essential part in scientific works ranging from urban planning, road traffic forecasting to spread of biological viruses [1], contextual marketing and advertising. New opportunities arise for the study of human mobility with the advent of the massive diffusion of mobile networks for personal devices such as GSM, UMTS, IEEE 802.16 WiMAX, IEEE 802.11 WLAN. Nearly all the members of adult population in major developed countries transport a GSM/UMTS mobile terminal, i.e. an electronic individual tag, with themselves. Moreover, in order to provide the service, the data of GSM/UMTS networks are already logged by mobile phone companies. The analysis of temporal and spatial movements of these mobile tags allows accurate estimation of urban/extraurban traffic flow without placing costly ad hoc sensors. Mobile network data represent a powerful mean for road traffic analysis, forecasting and real time monitoring and, ultimately, for the analysis and the detection of events and processes besides the traffic domain(e.g. traffic jam, velocity, congestions, road work, accidents etc.), which can affect the motion behavior of the masses (e.g. sport and leisure events, concerts, attractive shopping areas, working/living area cyclic processes etc.).

Techniques and models for mobile device flow analysis [2] have mostly focused on predictive models aiming at optimizing some mobile network system parameters such as cell dimensioning, antenna distribution, and load balancing [3], [4].

Alfredo Milani, Eleonora Gentili, Valentina Poggioni are with the Department of Mathematics and Computer Science, University of Perugia, Italy.

On the other hand a number of projects [3], [5] try to use the cellular network traffic to estimate different road traffic and transportation related quantities [6], [7], [8], such as speed and travel times between destinations [9], [10], [11], origin/destination (O/D) matrices [12], [2], road traffic congestions [11], road traffic volume or density [13], [14], etc.

Many projects are also active in the relatively recent area of mobile device localization which focuses on the position of the single mobile terminal for the purpose of providing spatial contextual services.

The main limitation of the existing approaches to traffic estimation is the lack of a model taking explicitly into account of the mobility and transportation infrastructures. The estimates are often based on purely statistical correlation approaches which usually assume users movement directions following a uniform probability distribution. On the other hand, physical and normative constraints to user mobility inside a cell (e.g. as roads topology, mandatory directions etc.) are usually not taken into account in those models, with few exceptions [15], [4], while relationships with traffic domain external events, such as social events and social processes (e.g. work/home commuting, shopping periods etc.) are completely ignored.

Moreover some issues such as *privacy* and *scalability* are also problematic. For instance, techniques for inferring O/D matrices [2] uses information about the *Location Areas (LA)* over the time, where a LA is a set of cells where the mobile terminal is assumed to be located. In other words the algorithm needs to identify time, origin and destination LAs of the whole trip made by each single telephone, thus representing a remarkable privacy infringement. Mobile device localization detect the spatial position of the single user, by using techniques based on distance from the cell antenna (for example in [2]), or assuming the placement of special detector antennas for enhancing the accuracy of the localization. Although the remarkable precision is obtained, in both cases there are relevant problems of privacy and scalability. In fact, due to the huge amount of data generated by monitoring, each single terminal position in a cell would requires an enormous bandwidth, storage and computational cost.

In this work we propose a model which integrates spatial networks with mobile phone networks, in order to monitor, analyze and predict the user traffic on the mobility infrastructure and to make detection and inference about social events and processes in place, on the basis of anonymous aggregated data. The aim is that by integrating mobility constraints (e.g. available roads), it is possible to improve the accuracy of predictions the cellular network based on the mobility/transportation network and vice versa. Moreover social event/processes which take place can also be detected, and conversely the knowledge of those events/processes can

improve the predictive model in the mobility domain.

In particular consider temporal data series describing the "handover" of anonymous users, i.e. the number of users which traverses any of the six boundaries of an hexagonal cell in a mobile phone network. The choice of handover data is due to different reasons: (1) *Privacy issues*. Anonymized handovers can easily be made available and can be securely and effectively transmitted while tracking the positions of a single terminal would represent sensitive data about the individual user behavior. (2) *Performance* and *scalability issues*. The size of the information to process remains constant as the number of users increase.

The rest of the paper is organized as follows: Definitions and relationships between spatial networks and mobility networks are introduced in Section I, while a model for inferring spatial mobility flow from one word data is presented in Section II. An adaptive estimate model, used as a basis for event detection, is presented in Section III. Experiments for car accident detection are presented in Section IV, and discussion on possible directions for future works in Section V concludes the article.

I. Spatial network and Cellular Network

In this section we introduce a model for integrating the knowledge of a spatial network which constraints users movement and the knowledge of the cellular network covering the same physical area.

A *spatial network*, or *mobility network*, is a set of physical means and normative constraints, such as roads, railways, underground transportation, pedestrian area, one-way lanes and highways, which narrow the mobile user mobility. In general more than one cellular network with different sizes and topologies can insist on the same area. Here we assume that a single *cellular network* is operating in the given area and it is organized in the usual hexagonal grid of cells with each antenna centered in a cell. According to the usual notation, given a reference cell, say cell 0, we refer to its neighbour cells, by numbers from 1 to 6 clockwise as shown in Fig.1.

A *spatial network* S can be described as $S = (N, A, D, loc)$ where N are nodes, $A \subseteq N \times N$ are directed arcs, and loc is a *location* function $loc : N \to D$ mapping nodes onto positions in the bi-dimensional area of interest $D \subseteq \mathbb{R}^2$.

A *cellular network* $M = (C, D, g, m)$, organized in an hexagonal grid, is defined by a set of cells C, a function $g : C \times \{1 \dots 6\} \to C$, which describes the *grid topology*, (g returns the i-th neighbour of a given cell or returns the same cell if no i-th neighbour exists,e.g.it is on the border) and a boolean function $m : C \times D \to \{T, F\}$ checking whether a given position of $D \subseteq \mathbb{R}^2$ belongs to a cell.
Note that g and m should verify the hexagonal grid topology.

When a cellular network M shares the same domain area D with a spatial network S_0, we can consider the spatial network "projection" over the cells, or equivalently we can see C as "cutting" S_0 into a family of disjunct spatial subnetworks $\{S_i\}$. In order to identify the spatial subnetwork corresponding to each cell, it is useful to introduce additional nodes in correspondence of the cutting edges whenever an arc of the spatial network crosses the boundary between two cells.

Projected spatial network. The projection $S = \pi(M, S_0)$ of $S_0 = (N_0, A_0, D, loc_0)$ according to a cellular network $M = (C, D, g, m)$, is the spatial network $S = (N, A, D, loc)$ obtained by S_0, such that
1) $\forall n \in N_0 \Rightarrow n \in N$ and $loc_0(n) = loc(n)$,
2) $\forall (n', n'') \in A_0$ s. t. $\exists c \in C$ with $m(loc(n'), c) = m(loc(n''), c) = T$ then $n', n'' \in A$, i.e. all the arcs in S_0 which originates and ends in the same cell, also belong to S,
3) for each arc (n', n'') whose ends do not lie in the same cell, let $m(loc(n'), c') = m(loc(n''), c'') = T$ such that $c' \neq c''$ are two neighbor cells, then a new node n''' and two new arcs, respectively (n', n''') and (n''', n'') will be added to the set of network nodes N and arcs A; the position of the new node $loc(n''')$, will be assigned such that the node lies on the border between the two neighbors cells (note that n''' belongs to both cells, i.e. $m(loc(n'''), c')$ and $m(loc(n'''), c'')$ are both true),
4) finally, if an arc of S_0 traverses more than two cells, then the arc is cut in a series of subarcs according to the previous procedure.

An example of a spatial network and its projection on a cellular network is shown in Fig.1.

Cell spatial network. The projection operation $\pi(M, S_0)$ partitions S into subnetworks. In particular for each cell $c \in C$ there is an associated *cell spatial subnetwor* $S \mid_c$ defined by the restriction of S to all nodes and arcs lying inside c, i.e. in the domain area $D \mid_c = \{d \in D \mid m(c) \text{ is true}\}$. It is possible to identify in $S \mid_c$ two family of sets of nodes $I_{c,c_i} \subseteq N$ (respectively $O_{c,c_i} \subseteq$ N) for $i = 1 \dots 6$, which represent the set of nodes on the edge between the neighbors c_i of the cell c and connect inbound (outbound) arcs of c with outbound (inbound) arcs of c_i. The set of nodes $I_c = \bigcup_{i=1}^{6} I_{c,c_i}$ and $O_c = \bigcup_{i=1}^{6} O_{c,c_i}$ represent respectively the *source* and *sink* nodes for the spatial subnetwork limited by cell c. Since after the projection, by construction, it does not exists any arc of S crossing cell boundaries, I_c and O_c are the only sources and sinks for the flow in cell c.

The projection operation π is defined by successive incremental splits upon properties of connectivity of the spatial graph and the cell area domains. It is easy to see that projection process can be extended for more complex characterizations of the spatial network which consider features on arcs or nodes, such as costs, distances, speed and time between nodes, capacities and probabilities.

II. An Integrated Model for Spatial and Communication Network

Given a spatial network $S \mid_c$ delimited by a given cell c (cell 0 or c_0 in the following) the amount of user flow inside/outside the cell is completely described by the data available from the cell control unit. Assume that U_0^t denotes the amount of users in the current cell c at the time slot t(stationary users); $HO^t(i, j)$ represents the handovers, i.e. the amount of mobile terminals moving from the cell c_i towards the cell c_j at the time t, then $HO_{in}^t = \sum_{i=1}^{6} HO^t(i, 0)$ and $HO_{out}^t = \sum_{i=1}^{6} HO^t(0, i)$ represent respectively all the users coming in and going out the reference cell at time t. In order to relate these data to the traffic flow in the different parts the mobility network we need to introduce some definitions.

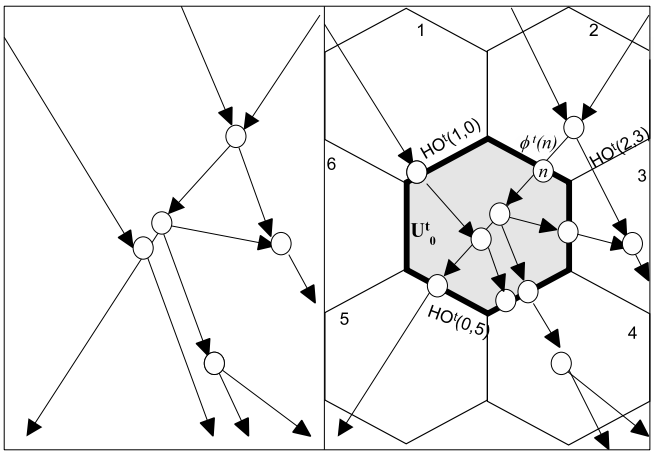


Fig. 1. Spatial mobility network and cellular network projection

Let P the set of connected components of $S\mid_c$, for each $p \in P$:

- $IN(p)$ is the set of the *source nodes* of the component p, i.e. all the nodes in $p \cap I_{c,c_j}$, $\forall j = 1..6$,
- $OUT(p)$ is the set of the *sink nodes* of the component p, i.e. all the nodes in $p \cap O_{c,c_j}$, $\forall j = 1..6$.

Moreover $ON(k)$ is the set of all the nodes in $IN(p)$ and $OUT(p)$ lying on the edge between cell c and its neighbor c_k.

A. Mobility Network Flow Equations

Given the HO series between the current cell c and all its neighbors $c_1 \dots c_6$, and given the network topology $S \mid c$ projected on the cell 0, it is possible to define an inference model for deriving the flow of mobile users on the mobility network.

The model is based on the flow equations which relate the user flow in the cellular network with the flow in the spatial network that restricts user mobility. Let U_0^t be the amount of users in the current cell c and HO_{in}^t, HO_{out}^t the handover data at the time-slot t. The flow on the spatial network delimited by the cell c is *admissible* if

$$HO_{in}^t + U_0^t = HO_{out}^{t+1} + U_0^{t+1}. \tag{1}$$

Considering the set P of all connected components, we can assume that for any $p \in P$ there exists an admissible flow, and let $\phi^t : N \to \mathbb{N}$ be the function assigning to each node $n \in N$ the number of users $\phi^t(n)$ in the node n at the time t and $U_{0,p}^t$ the users stationary at the time t in the nodes of p inside the cell 0, then

$$\forall p \in P, \sum_{n \in IN(p)} \phi^t(n) + U_{0,p}^t = \sum_{n \in OUT(p)} \phi^{t+1}(n) + U_{0,p}^{t+1}. \tag{2}$$

Assuming that only the handover and stationary users data are available from the cellular network, it is not possible to know how the users are distributed over the paths in the cell. So for each connected component $p \in P$ we know the exact values of $\phi^t(n)$ and $U_{0,p}^t$ only when, for each edge k of cell 0, $IN(p) \cap ON(k) = \{n_1\}$ and $OUT(p) \cap ON(k) = \{n_2\}$. In this case we have $\phi^t(n_1) = HO^t(k,0)$ and $\phi^t(n_2) = HO^t(0,k)$.

Let consider the following equivalence relation $\sim$ between the elements of P: $\forall p_1, p_2 \in SP$ then $p_1 \sim p_2 \Leftrightarrow \exists o_1 \in OUT(p_1)$, $\exists o_2 \in OUT(p_2)$, $\exists k_1 \in \{1,\dots,6\}$: $o_1, o_2 \in ON(k_1)$, or $\exists i_1 \in IN(p_1)$, $\exists i_2 \in IN(p_2)$, $\exists k_2 \in \{1,\dots,6\} : i_1, i_2 \in ON(k_2)$.

The relation $\sim$ partitions P into equivalence classes having either sources or sinks on the same side of the cell. Therefore it can be more useful to provide (2) with respect to the $\sim$ equivalent classes. Since the connected components having paths on the same edge of the cell belong to the same equivalent class, and since

$$\sum_{n \in IN(p) \cap ON(k)} \phi^t(n) = HO^t(k,0),$$

$$\sum_{n \in OUT(p) \cap ON(k)} \phi^t(n) = HO^t(0,k),$$

both equations (1) and (2) can be rewritten for each equivalence class induced by the relation $\sim$.

In practice, the equivalent classes can be thought as *clusters of paths* originating from or sinking to the same set of cells.

Fig.2 represents some possible spatial networks related to the reference cell. In Fig.2.a only one connected component exists. Then, the general flow equation (1) coincides with the one of the connected component. In this case our model is exact to estimate the number of users in the paths and we say that we reach *component level* accuracy. In Fig.2.b we can see two connected components belonging to two different clusters. In this case we reach *component level* accuracy. The cases represented in Figs.2.c and 2.d are equivalent in terms of handover data, but they are different from the topological point of view. While Fig.2.d has a unique connected component, we have two connected components in Fig.2.c which belong to the same equivalent class. Even if an equation for each connected component can be written, the handover data are provided for each edge (and not for single path). So the accuracy level decreases to *cluster level*.

B. Inferring user flow

Assuming that the initial number of mobile user in component cluster c at the initial time slot 0, is known, it is possible to infer the number of stationary users in a given time slot in the cluster by iteratively applying the flow equations generated by the spatial network on a cell C.

In fact, from the general equation of admissible flow (1), for each cluster of components (i.e. for each equivalent class) we have:

$$U_0^{t+1} = HO_{in}^t + U_0^t - HO_{out}^{t+1}.$$

With consecutive substitutions, we obtain

$$U_0^{t+1} = HO_{in}^t + HO_{in}^{t-1} + U_0^{t-1} - HO_{out}^t - HO_{out}^{t+1};$$

By regroupings terms, we obtain

$$U_0^{t+1} = \sum_{j=0}^{t} HO_{in}^j + U_0^0 - \sum_{j=0}^{t} HO_{out}^{j+1}, \tag{3}$$

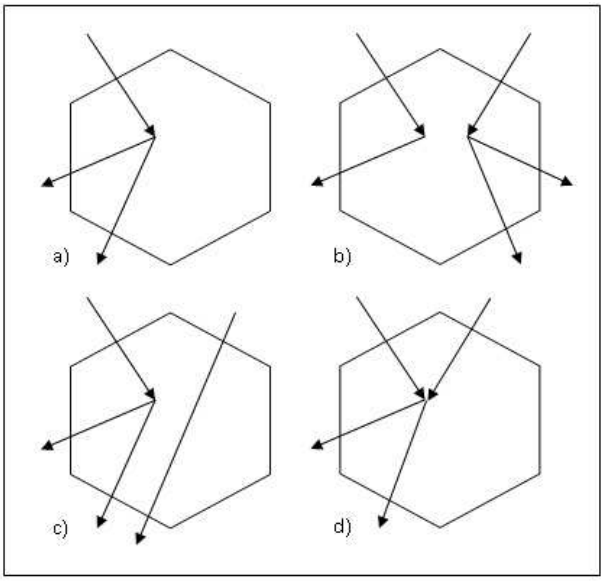


Fig. 2. Connected components and clusters in cell spatial network

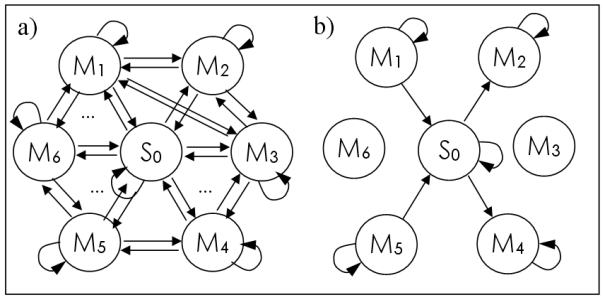


Fig. 3. State diagram: a)complete b)reduced by mobility constraints

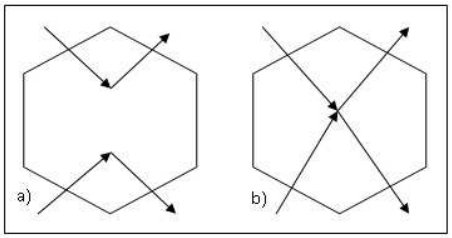


Fig. 4. An example of different spatial networks that are unrecognizable from their handover

.

The inference procedure assumes that the amount of users in stationary state inside every cluster at time 0 is known.

It is easy to note that the ability of distinguish the flow within a connected component of $S\ |_c$ is limited by the number of classes induced by $\sim$. In other words if two connected components are in the same class the amount of their individual inbound/outbound flow cannot be precisely determined considering only the cell handovers. In the ideal case if each connected component belongs to a distinct class, its flow is fully described by the handovers data.

The effectiveness and accuracy of the inference technique, based solely on handover data, greatly depends on the cell resolution/granularity, i.e. the relative size of the cell with respect to the spatial network, and on the spatial network connectivity. For instance the presence of high connectivity subnetworks or hubs, such a square or a park, where the mobile phone holders can move "freely" in any direction, can narrow down the accuracy. On the other hand, a cell covering an highway section in an area with no other road can provide high accuracy.

III. Spatial Networks Prediction and Estimation

In this section we present a prediction model based on Markov chain and an adaptive flow estimation model which exploits the underlying spatial network in order. These models can improve the performance of predictions and give a better estimate the flow within the single connected component when inference based on flow equations cannot determine a unique answer, i.e. the equations have not a unique solution due to large clusters of components. The two models represent the core modules of the event detection system presented in the Experiments Section.

A. Prediction Markov Model

The prediction model is based on the Markov model proposed in [4] for mobile network management. Mobile user movements towards/from the cell are represented by a state diagram associated to a transition matrix assigning probabilities assigned to each movement in the given time slot. A complete state diagram for 7 direction levels is represented in Fig.3. The parameters of the Markov model, i.e. the specific transition probabilities, can be effectively determined by a statistical analysis of handover series at the given time slot granularity.

It is worth noticing that spatial network constraints can reduce the number of states, the entries of the incidence matrix and thus the complexity of the Markov model. For instance the projected spatial network of Fig.4.a can reduce the predictive model to 4 states as shown in Fig.3.b. Nevertheless the Markov model is not adequate by itself for flow analysis since qualitatively different mobility networks, as the ones shown in Fig.4 can lead to the same Markov model structure. The technique shown in Section II-B can be applied in order to calculate the flow in clusters of connected components.

B. Flow estimation

In order to improve the accuracy of flow inference within a class of connected component, it is possible to use an estimate of flow distributions on sources/sinks, when deterministic inference is not possible.

Assume that for each set of source nodes $I(c, c_i)$ (sink nodes $I(c, c_i)$) lying on the same border i of cell c, the distribution $\rho^t(n)\ \forall n \in I(c, c_i)$ i.e. the expected percentage of handovers $H(i, 0)$ ($HO(0, i)$)which take place at time t because of users entering (leaving) cell c from node n is known. It is apparent

that the flow equations can be restated in the form of estimate for each single connected component $\forall p \in P$,

$$\sum_{\substack{n \in IN(p) \\ k \in 1...6}} \sigma^t(k,0) + V^t_{0,p} = \sum_{\substack{n \in OUT(p) \\ k \in 1...6}} \sigma^{t+1}(k,0) + V^{t+1}_{0,p} \quad (4)$$

where the term $\sigma^t(k,0) = HO^t(k,0) * \rho^t(n)$ represents the estimated flow through each source/sink node and V the current estimated stationary users calculated iteratively. The estimate can also be propagated along the mobility network and between cells by simple boundary equations, since incoming flow for a cells is the outgoing flow for its neighbor and vice versa.

It is worth noticing that the flow estimates are also required to be *admissible*, i.e. they should not contradict the general flow equations. On the other hand, contradictions can emerge over the time by iterating wrong estimates. For example a low estimate of flow distribution along a path can lead to observing many more users than expected exiting from that path in a neighbor cell. In this case, for example, the distribution parameters can be increased along the contradictory path to re-establish the consistency.

On this basis it is possible to design a scheme for adaptive flow estimation, where the estimation parameters are dynamically changed in order to maintain consistency between the estimate and the observed data (i.e. handover and total stationary users):

Adaptive Flow Estimation Scheme:

1) Estimate current flow along sources and network paths of cell c using the real data and current distribution parameters
2) Calculate flow constraints in neighbor cells of c using current estimate data and parameters
3) If current estimate conflicts with the previous constraints for cell c then (3.1) revise distribution parameters, $\sigma^{t+1} = r(\sigma^t)$ to establish consistency and (3.2) back-propagate revision to c neighbors.

A key point of the adaptive algorithm is the update function r which revises the estimate distribution parameters $(\sigma^t(k,0))$. The current implementation uses an iterative algorithm based on PSO [16]–[18] to find the increment/decrement size distribution for re-establishing the consistency.

IV. Experiments: Car accident Detection in Highways

The proposed model for the analysis and the estimation of traffic flow in mobility network has been experimented in the domain of car accident detection in the Great Ring Highway (GRH) A90 surrounding the city of Rome in Italy. Timed data series of handover logs from a major national GSM mobile phone network has been used. The provided data regard 32 months for a cluster of 24 GSM cells covering a section of the GRH with different cells dimensions and road density in the domain area (see Fig.5). In addition, reports from the national highway traffic control system have been used as a source of car accidents events in the GRH; the salient types of information include: *start/end time of event (i.e. return to normal traffic condition), place and direction of the event,*

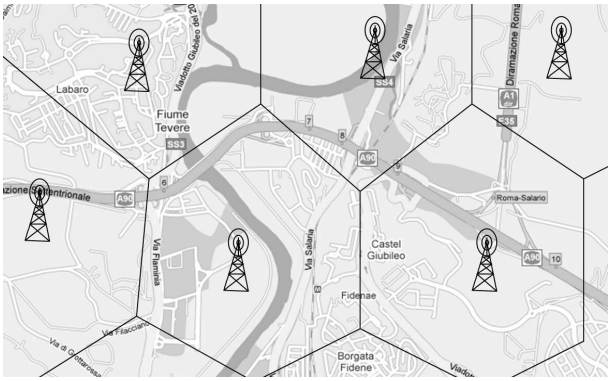


Fig. 5.

class of traffic impact (from 0=null to 6=complete block). The event features which have been considered are: *start/ending time*, *place* (i.e. mobility network connected component), *direction* (which GRH lane is concerned for the event), and *type of event* (car accident or generic anomalous events).

The data of the first 24 months have been used to determine the initial values for the adaptive estimation model and the weights of the Markov predictive model and alert thresholds, while data of the last six months have been used from the actual experiment of detection. The parameters have been computed for each 15 minutes time slot on a week day base, Monday to Saturday, while Sundays and public holidays have been included in a different class, since their traffic behaviors exhibit common similar patterns.

The general architecture of the detection systems is based on different classes of indicators and thresholds which trigger alerts in the algorithm. Indicators based on global handover traffic in the cell are compared with the predictive model in order to detect start/end and type of events, while indicators of deviation from the adaptive estimation model are used to detect the place of the event the direction of accident. The scheme for the event detection loop is depicted below:

```
if event(HO_in,HO_out) then
  eventStarted ← true
  if carAcc(HO_in,HO_out) then
    if carConn() then
      output estimatePlace()
      output estimateDirection()
    end if
  end if
else
  eventStarted ← false
end if
```

Any start/end event is firstly detected by a relevant change in global handover volume $HO_{in} + HO_{out}$ with respect to the expectation according to the Markov based model. The value of the corresponding threshold ϑ^t_g is based on the variance of handovers volume (g represents the event type). The beginning of an event of type car accident is related with a sudden increase of the number of HO_{in} with respect to HO_{out}, see Fig.6). A threshold ϑ^t_{car} is compared against the averaged difference $HO_{in} - HO_{out}$ over consecutive time

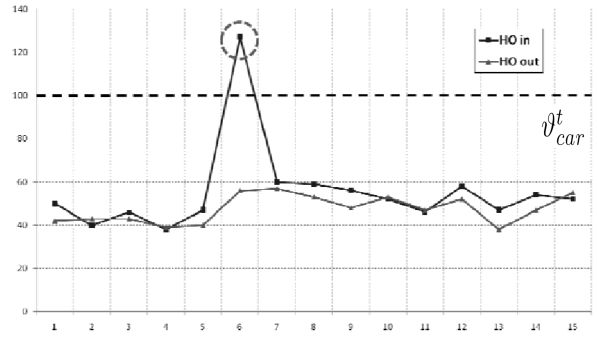


Fig. 6. A peak in incoming handovers over threshold ϑ^t_{car}.

TABLE I
DETECTION RESULTS

	Events	Precision	Start $\leq$	Direction
Alg_{est}	382	97 %	85 %	98 %
Alg_{det}	495	75 %	83 %	65 %

intervals to distinguish car accidents from other events such as anomalous increment of traffic. On the other hand the "end" of the event is recognized by a return to a normal traffic condition. When a potential car accident is detected, the estimated flow allows ones to determine the location of the event, which is decided to be the connected component/s with the greater estimated flow variation. The direction of the event is calculated by comparing the inbound/outbound estimated flow in the connected component corresponding to the highway lanes. The control *carConn()* filters out those flow variations which are due to car accidents already detected in the nearby cells and could be erroneously recognized as new events. The experimental results of the original algorithm Alg_{est} have been compared with a version, Alg_{det}, which does not take into account of mobility network estimates, but only uses the deterministic inference rules. Car accidents with null effect on the traffic have been excluded from the statistics.

As shown in Table I, the results are quite encouraging: Both Alg_{est} and Alg_{det} algorithms detected all the 371 accidents events in the traffic control report, while Alg_{det} has a considerable lower precision with a remarkable number of false positives (*Events* and *Precision*). It is interesting to note that the *start of event time* returned by both algorithms (*Start*$\leq$) is better, i.e. anticipated, with respect to the starting time given by the national traffic control system. This is because the mobile users data are acquired in real time, while the accident alert reach the national traffic system by different channels, e.g. drivers, police patrols etc., which are not always promptly activated.

The number of false positives (*Precision*) of Alg_{det} is mostly due to the inability of distinguishing the "noise" of events taking place in the urban area nearby the highway, while Alg_{est} uses the analysis of the traffic on the urban connect component to filter out events not taking place in GRH. The accuracy of direction detection (*Direction*) is found to be high. Failure of detection are sometimes inevitable due to a number of reasons. For example, car accidents in a lane sometimes can slow down the traffic in the other one for different reasons: rescuing cars blocking it, traffic police deviating the traffic on the other lane or the phenomenon of "accident curiosity" which draws the attention of drivers on the event slowing down the opposite lane traffic. If this happens within the first 15 minutes time slot, the algorithm is not able to detect a suitable direction since the two cannot be distinguished, while a finer time granularity in the data is expected to improve the direction detection ability. A further analysis has shown that most of the false positives detected by Alg_{est} are due to traffic variation induced by car accidents in nearby cells. This suggests that the management of connected events should be further refined.

V. FUTURE DEVELOPMENTS AND CONCLUSIONS

Cellular networks, besides their communication purpose, can be seen as mobility sensor networks already in place which offer a great potential for the analysis of users flow in an area. A model which integrates mobility constraints and cellular networks has been proposed in order to analyze, monitor, forecast and detect events and processes in the mobility infrastructure. The use of cell level handover meets data privacy and scalability requirements, while the knowledge of the mobility infrastructure allows ones to obtain reasonable estimates of the flow at the connected component level. The integrated model and the proposed technique of adaptive flow estimation have been successfully experimented in the domain of car accident detection.

Future works includes the investigation of techniques for the application of the model to high density urban area, where the high road density does not allow a fine grain analysis of the flows, although the increasing diffusion of the so called microcells and nanocells is soon expected to provide a suitable granularity.

More generally suitable models, which integrate "sensors already in place" (e.g. cellular networks, payment systems, bus/train ticketing systems, video surveillance etc.) and mobility infrastructures constraints, are of great interest for the analysis of social events (e.g. entertainment, sport events, festival, commercial/leisure area attractors etc.) and social processes (e.g. working day/vacation days cycle, work/school/home cycle etc.) which involve movement of people in the physical space and conversely, for analyzing the impact of events on the mobility infrastructures and their planning and management.

REFERENCES

[1] M. C. Gonzales, C. A. Hildago, and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature*, vol. 453 (5), pp. 779–782, 2008.

[2] N. Caceres, J. Wideberg, and F. Benitez, "Deriving traffic data from a cellular network," in *13th World Congress for ITS system and services*, vol. 3, 2006, pp. 1–10.

[3] ——, "Review of traffic data estimations extracted from cellular networks," *Intelligent Transport Systems, IET*, vol. 2, no. 3, pp. 179–192, Sept. 2008.

[4] P. Fülöp, K. Lendvai, T. Szálka, S. Imre, and S. Szabó, "Accurate mobility modeling and location prediction based on pattern analysis of handover series in mobile networks," in *MoMM '08: Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*. New York, NY, USA: ACM, 2008, pp. 219–226.

[5] D. Valerio, A. D'Alconzo, F. Ricciato, and W. Wiedermann, "Exploiting cellular networks for a road traffic estimation: a survey ad a research roadmap," in *IEEE 69th Vehicular Technology Conference (IEEE VTC '09-Spring)*, Barcelona, Spain, April 26-29 2009.
[6] N. P. 70-01, "Probe-based traffic monitoring: State of the practice report," University of Virgina - Center for Transportation Studies, Tech. Rep., November 2005.
[7] J. Virtanen, "Mobile phones as probes in travel time monitoring," Finnish Road Administration, Helsinki, Finland, Tech. Rep., 2002.
[8] M. Höpfner, K. Lemmer, and I. Ehrenpfordt, "Cellular data for traffic management," in *6th European Congress and Exibition on Intelligent Transport System and Services*, 2007.
[9] Y. Youngbin, "The state of cellular probes," California PATH Research Project, University of California, CA, USA, Tech. Rep., July 2003.
[10] J. Remy, "Computing travel time-estimates from gsm signalling messages: the strips project," *Intelligent Transport Systems, IET*, 2001.
[11] J. Ygnace, "Travel timespeed estimates on the french rhone comdor network using cellular phones as probe," INRETS, Lyon, France, Lyon, France, Tech. Rep., December 2001.
[12] J. White and I. Wells, "Extracting origin destination information from mobile phone data," in *11th Int. Conf. on Road Transport Information and Control, LONDON*, 2002, pp. 30–34.
[13] C. Ratti, R. Pulselli, S. Williams, and D. Frenchman, "Mobile landscapes: using location data from cell-phones for urban analysis," *Environ. Plann. B Plann. Des.*, vol. 33 (5), pp. 727–748, 2006.
[14] K. THIESSENHUSEN, R. SCHAEFER, and T. LANG, "Traffic data from cell phones: a comparison with loops and probe vehicle data," Institute of Transport Research German Aerospace Center, Germany, Tech. Rep., 2003.
[15] J. Kammann, M. Angermann, and B. Lami, "A new mobility model based on maps," *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, vol. 5, pp. 3045 – 3049, 2003.
[16] B. White and M. Shah, "Automatically tuning background subtraction parameters using particle swarm optimization," in *IEEE International Conference on Multimedia and Expo*, 2007, pp. 1826–1829.
[17] X. Li, F. Yu, and Y. Wang, "PSO algorithm based online self-tuning of pid controller," in *CIS 2007, International Conference on Computational Intelligence and Security.*, 2007, pp. 128–132.
[18] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*. Morgan Kaufmann, 2001.

An Embedded Two-Layer Feature Selection Approach for Microarray Data Analysis

Pengyi Yang and Zili Zhang

***Abstract*—Feature selection is an important technique in dealing with application problems with large number of variables and limited training samples, such as image processing, combinatorial chemistry, and microarray analysis. Commonly employed feature selection strategies can be divided into filter and wrapper. In this study, we propose an embedded two-layer feature selection approach to combining the advantages of filter and wrapper algorithms while avoiding their drawbacks. The hybrid algorithm, called GAEF (Genetic Algorithm with embedded filter), divides the feature selection process into two stages. In the first stage, Genetic Algorithm (GA) is employed to pre-select features while in the second stage a filter selector is used to further identify a small feature subset for accurate sample classification. Three benchmark microarray datasets are used to evaluate the proposed algorithm. The experimental results suggest that this embedded two-layer feature selection strategy is able to improve the stability of the selection results as well as the sample classification accuracy.**

***Index Terms*—Feature selection, Filter, Wrapper, Hybrid, Microarrays.**

I. Introduction

CURSE-OF-DIMENSIONALITY is a major problem associated with many classification and pattern recognition problems. When addressing the classification problems with a large number of features, the classifier created will often be very complex with poor generalization property. This is especially true in analyzing microarray datasets which inherently have several thousand of features (genes) with only a few dozen of samples [1]. One effective way to deal with such problems is to apply feature selection technologies [2]. The benefits of feature selection are as follows:

- Reducing the number of features to a sufficient minimum will cut the computational expenses.
- Feature selection can reduce the noise introduced in the classification process, which then will improve sample classification accuracy.
- From the biological perspective, minimizing feature size can help the researchers to concentrate on the selected genes for biological validation etc.
- The higher the ratio of the number of training sample to the number of features used by classifier, the better the generalization ability of the resulting classifier [3]. In other words, minimizing the size of the features can improve the generalization property of the resulting classification model.

Pengyi Yang is with School of Information Technologies (J12), The University of Sydney, NSW 2006, Australia.
E-mail: yangpy@it.usyd.edu.au

Zili Zhang is with Faculty of Computer and Information Science, Southwest University Chongqing 400715, China; School of Information Technology, Deakin University, Geelong, Victoria, Australia, 3217.
E-mail: zili.zhang@deakin.edu.au

Based on the selection manners, feature selection methods can be broadly divided into filter, wrapper and embedded approaches [4]. Among them, filter and wrapper approaches are the most popular ones in biological data analysis. Genetic Algorithm (GA), as an advanced type of wrapper selector, has been applied as the search scheme for microarray data analysis recently [5], [6], [7]. Unlike forward selection and backward elimination wrappers which select features linearly, GA selects features nonlinearly by creating feature combinations randomly. This character of GA accommodates the identification of the nonlinear relationship among features. Moreover, GA is efficient in exploring large feature space [8], [9], which makes it a promising solution for gene selection of microarray. However, as many wrapper selection strategies encountered, GA often suffers from overfitting [10] because an inductive algorithm is usually used as the sole criterion in feature subset evaluation. Another problem is that GA is unstable in feature selection because of its stochastic nature. Furthermore, GA is a near optimal search algorithm. This means when applying GA, we are facing the risk of trapping into local optimal solutions. This risk rises exponentially with the increase of the feature size.

Different from wrapper strategies, filter approaches do not optimize the classification accuracy of a given inductive algorithm directly. Instead, they try to select a feature set with a predefined evaluation criterion. Examples include t-test [11], χ^2-test [12], Information Gain [13] etc. Although filtering algorithms are superior in selecting of better generalization features which often extended well on unseen data, there are manifold disadvantages they suffered from. Firstly, filtering approaches totally ignore the effects of the selected feature subset on the performance of the inductive algorithm. However, the performance of the inductive algorithm may be crucial for accurate phenotype classification [14]. Secondly, filtering approaches are often deterministic and greedy based. This leads to only one feature profile being selected, which is often suboptimal, whereas a different feature profile may produce better classification results. Moreover, Jaeger et al. demonstrated that in microarray data analysis genes obtained by aggressive reduction with filter based methods are often highly correlated with each other, thus, redundant [15]. In classifier construction and sample classification, such a redundant feature set often increases the model complexity while decreases the generality [3].

In order to combine the strengths of filter and wrapper approaches while avoiding their drawbacks, we recently in-

troduced several hybrid feature selection strategies [16], [17]. In those studies, however, the filtering algorithm is used either as prior evaluator [16] or an intermediate scoring criteria [17]. In this study, we gives the filtering algorithm more control over the feature selection results and propose an embedded two-layer feature selection framework. The aim is to testify whether such formulation could improve sample classification accuracy and feature selection stability. This approach justifies its name because a filter algorithm is embedded in the GA algorithm. The embedded filter is used to evaluate and reduce the feature subsets randomly generated by GA and then feed the reduced subsets to the inductive algorithm for pattern recognition. Hence, the feature selection process is broken into two stages. We named it GAEF (Genetic Algorithm with embedded filter) for convenience. Different from many hybrid methods relying on manipulating learning datasets [18], this embedded two-layer feature selection model has following advantages:

- With the random selection of GA and the pattern recognition of the classifier, stochastic nature is integrated into the hybrid system as well as the performance information of the inductive algorithm.
- The unstable issue of GA is minimized because GA is designated to pre-select a very large feature subset while the final feature set is actually determined by the filter algorithm embedded in it.
- Since GA only "loosely" selects a large feature subset, the possibility of trapping into a suboptimal solution is minimized while generalization property is enhanced.
- The integration of the performance information of a given classifier in sample classification is used to minimize the correlation of the filter selected features implicitly, resulting in a redundancy reduced and information enriched feature subset.

Therefore, this GAEF algorithm is expected to possess more stable and generalization quality in feature selection, which contribute to a higher sample classification accuracy comparing with those obtained by applying its components alone. We apply the proposed method to three benchmark microarray datasets, including binary-class as well as multi-class classification problems. The empirical results obtained by using the proposed model are compared with those obtained by using GA wrapper and filter algorithms individually. Moreover, the classification results of a popular GA/KNN algorithm developed by Li et al. [5] for microarray data analysis are provided as the third yardstick. It's worth noting that the proposed algorithm can also be applied to other feature selection domains such as image processing and combinatorial chemistry with minor modification.

The paper is organized as follows: In Section II, we present the overview of the proposed method. In Section III, the implementation and evaluation issues are detailed. Section IV provides the experimental results while Section V and Section VI discuss and conclude of the paper.

II. Embedded Two-Layer Feature Selection Approach

A. System Overview

From the data mining perspective, each sample in dataset is commonly described as a vector of the form $\mathbf{s}_i$=$[f_1, f_2, ..., f_n],\ (i = 1, ..., m)$, where m is the number of samples and n is the number of the features. The dataset is described as a $m \times n$ matrix D_{mn}=$\{(\mathbf{s}_1, y_1), (\mathbf{s}_2, y_2), (\mathbf{s}_m, y_m)\}$, where y_i is the class value of the ith sample. Feature selection is essentially to generate a reduced feature vector $\mathbf{s}'_i$=$[f_1, f_2, ..., f_d],\ (\mathbf{s}'_i \subset \mathbf{s}_i)$ which confines the dataset matrix into D_{md}=$\{(\mathbf{s}'_1, y_1), (\mathbf{s}'_2, y_2), (\mathbf{s}'_m, y_m)\}$ with the expectation to reduce the noisy and redundancy. The proposed GAEF approach utilizes a standard GA as the first layer of feature selection to generate and select large, pre-selected feature subsets $\mathbf{s}'_i$=$[f_1, f_2, ..., f_{d_1}],\ (\mathbf{s}'_i \subset \mathbf{s}_i)$. The embedded filter algorithm which serves as the second layer of feature selection is used to further determine a compact feature subset $\mathbf{s}''_i$=$[f_1, f_2, ..., f_{d_2}],\ (\mathbf{s}''_i \subset \mathbf{s}'_i)$ from each pre-selected feature subset of GA. Those further selected feature subsets are then fed into the classification algorithm for pattern recognition. For convenience, and without loss of generality, we simplify the notation of $\mathbf{s}'_i$ to s in the rest of the paper. Figure 1 illustrates the work flow of the GAEF model.

The algorithm performs following steps:

S1: Initially, GA randomly creates a set of chromosomes which representing various pre-selected feature subsets.

S2: Filter algorithm is invoked to select a further reduced feature subset from each pre-selected feature subset provided in GA chromosome.

S3: Feature sets selected by filter are then fed into classifier for sample classification and pattern recognition. After a classifier evaluates a given feature subset, it returns the classification strength of this feature subset to its corresponding pre-selected feature subset.

S4: After the whole population are evaluated, GA selects favorite chromosomes that can produce good feature subsets with a given filter in sample classification.

S5: The crossover and mutation operations are then conducted on the selected chromosomes with a predefined P_C (probability of crossover) and P_M (probability of mutation), respectively, and the next generation begins.

S6: Repeat steps 2-5 until terminating generation is reached and the final filter selected feature subsets are collected as the optimal feature profiles for sample classification and pattern recognition.

B. Subset Evaluation and Selection

In GA, the goodness of a candidate solution is evaluated by calculating a given fitness function using the bits configuration of this solution. In feature selection, such fitness function is often defined as the simple classification accuracy. However, the problem of using simple classification accuracy is that when the numbers of samples in different classes are imbalanced, the fitness score provided by such a measure could be misleading [19]. This can be shown with following examples. Suppose a binary-class dataset contains 5 samples from class A and 45

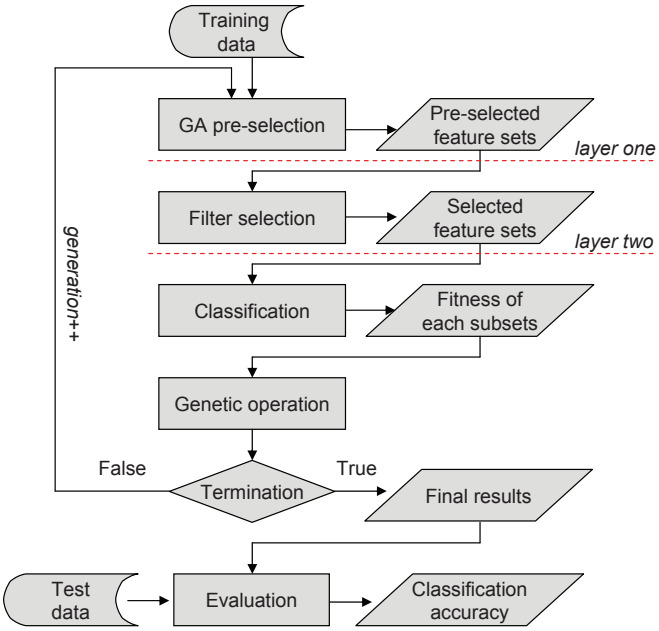


Fig. 1. GAEF work flow. GA is used to produce large, pre-selected candidate feature sets and a filter algorithm is invoked to select a compact feature set from those pre-selected sets for sample classification.

samples from class B. If a classifier misclassifies all samples in class A but correctly classifies other 45 samples in class B, the fitness score produced by simple classification accuracy measure is $45/50 \times 100 = 90\%$. However, no differential pattern is actually identified by the classifier, and the resulting feature subset is in fact useless for sample separation of unseen data. The problem worsen if the dataset at hand is multi-class. To overcome such problems, we utilized a balanced classification accuracy for feature subset evaluation and fitness calculation. Fitness function derived from such a balanced classification accuracy is defined as:

$$fitness(s) = \frac{\sum_{i=1}^{c} Se_i}{c} \quad (1)$$

where c denotes the number of classes in the dataset, and s denotes the subset under evaluation. Se_i denotes the classification sensitivity of the samples in class i, which is calculated as follows:

$$Se_i = \frac{N_i^{TP}}{N_i} \times 100, \quad (2)$$

where N_i^{TP} denotes the number of true positive classification of samples in class i, and N_i denotes the total number of samples in class i. For previous example, the fitness score given by this balanced accuracy measure is $(0/5+45/45)/2 = 50\%$. This result is significantly lower than that of simple classification accuracy measure which helps to correct the fitness score.

Followed by subset evaluation, tournament selection strategy is used for the selection of favorite chromosomes. In tournament selection, larger tournament size gives faster convergence speed of GA, and we found three member tournament selection is a good trade-off. Formally, the winner is determined as follows:

$$Winner = \arg\max_{s \in S} fitness_i(R(s)) \quad (i = 1, 2, 3) \quad (3)$$

where $R(.)$ is the random function which randomly selects feature subset from the population S of GA, while $fitness(.)$ determines the fitness of the randomly selected feature subsets.

C. Filters

χ^2-test and Information Gain are popular filtering algorithms and are commonly used in gene selection of microarrays [12], [13]. We used this two types of filtering algorithms for forming the proposed hybrid algorithm, respectively. When used for feature selection purpose, χ^2-test can be considered as to evaluating occurrence of certain value of a feature and occurrence of the class. The feature is then ranked with respect to the following quantity:

$$\chi^2(f) = \sum_{v \in V} \sum_{i=1}^{m} \frac{(N(f = v, c_i) - E(f = v, c_i))^2}{E(f = v, c_i)} \quad (4)$$

where $c_i,\ (i = 1, ..., m)$ denotes the possible classes of the dataset, while f is the feature that has a set of possible values denoted as V. $N(f = v, c_i)$ and $E(f = v, c_i)$ are the observed and the expected co-occurrence of $f = v$ with the class c_i, respectively.

Information Gain is another type of statistic measure for feature selection. It measures the number of bits of information provided in class prediction by knowing the value of feature. Again, let c_i belong to a set of discrete classes (1, ...,m).

V be the set of possible values for candidate feature f. The information gain of a feature f is then defined as follows:

$$Gain(f) = -\sum_{i=1}^{m} P(c_i)\log P(c_i) + \sum_{v\in V}\sum_{i=1}^{m} P(f=v)P(c_i|f=v)\log P(c_i|f=v) \quad (5)$$

D. Classification

kNN is a relatively computational efficient classifier which has been applied by several studies in evaluating gene selection [5], [6]. It calculates the similarity, called the distance, of a given instance with others and assign the given sample into the class to which the k most similar samples belong. Such a similarity can be defined as Euclidean distance, Manhattan distance or Pearson's correlation etc. We utilized k-Nearest Neighbor (kNN) classifier for sample classification and evaluation of the "merits" of feature subsets. In our GAEF algorithm, Euclidean distance is used for sample similarity comparison. Formally:

$$ED(\mathbf{x_1},\mathbf{x_2}) = \sqrt{\sum_{i=i}^{d}(x_1(f_i)-x_2(f_i))^2}, \quad (f_i \in s) \quad (6)$$

where $\mathbf{x_1}$ and $\mathbf{x_2}$ are two samples described by the subset s which is a feature vector $[f_1, f_2, ...f_d]$.

III. Experimental Settings

A. Datasets

Microarray technologies make parallel evaluation of several thousand of genes possible. On the contrary, the samples collected for such evaluation are often with limited size– a few dozen. Therefore, most microarray datasets are with large number of gene features and limited number of samples, which make them ideal for the evaluation of the proposed algorithm. In the initial experiment, we evaluated the proposed method with three benchmark microarray datasets. The first two, namely "Colon" and "Breast", are binary-class datasets, which are generated from microarray studies of colon cancer [20] and breast cancer [21], respectively. The third microarray dataset called "MLL" is a multi-class dataset generated from a leukemia study [22]. Table I summarizes each dataset.

TABLE I
MICROARRAY DATASETS USED IN EVALUATION

Name	Colon	Breast	MLL
No. of Gene	2000	24481	15154
No. of Sample	62	97	72
No. of Class	2	2	3
C1:	Normal (22)	Relapse (46)	ALL (24)
C2:	Cancer (40)	Non-relapse (51)	MLL (20)
C3:			AML (28)

Expression values of each gene in each dataset are normalized into [0,1] with the mean of 0 and the variance of 1 before feeding for pattern recognition. As to the Breast and the Prostate datasets, for the purpose of computational efficiency, we conducted a Symmetrical Uncertainty analysis [23] to reduce the feature dimension from 24481 to 2000 and from 15154 to 2000, respectively.

B. GAEF Implementation

A standard GA is used in GAEF implementation as the first layer of feature selection. The population size of GA is set to 100. We adopt single point crossover and mutation, with the probability of 0.6 and 0.02, respectively as they produced good classification results. Three members tournament selection strategy is utilized for favorite chromosome selection. We implemented three termination conditions. The first condition is that the algorithm reaches the 50th generation. The second one requires that the chromosomes in a GA generation converge to 90%. The last condition is that no fitness improvement is generated in the last 5 sequential GA generations.

As to the GA pre-selection size, after some preliminary test we decide to fix it to 400 genes as it produces good experimental results. In regard to the second gene selection layer, we examined χ^2-test and Information Gain algorithms. By exploring combining different filters, we are able to evaluate the generality of the proposed embedded two-layer feature selection model. Based on the previous study [24], in most cases only a few dozen (or a few) genes are needed for sample classification. Therefore, we vary the embedded filter selection of the gene sizes from 5 to 25 with a step of 5. Lastly, each gene subset is evaluated by kNN classifier. Previous studies, demonstrated that small values of k such as odd number of 3 and 5 often produce good classification results [5]. In our experiments, $k = 3$ is arbitrarily chosen.

Table II summarizes the parameter setting of the GAEF model.

TABLE II
GAEF PARAMETER SETTINGS

Parameter	**Value**
Genetic Algorithm	Single Objective
Population Size	100
Chromosome Size	400
Selector	Tournament Selection
Crossover	Single Point (0.6)
Mutation	Single Point (0.02)
Termination Condition	Multiple Condition
Candidate Filter	χ^2-test; InfoGain
Filtering Size	5 to 25 (step of 5)
Inductive Algorithm	kNN

C. Correlation Evaluation

As pointed out by Jaeger et al. [15], in microarray study genes obtained by aggressive reduction with filter based methods are often highly correlated which inevitably introduce noisy and redundancy. Therefore, several studies attempted to minimize the correlation of selected genes to the minimum [25], [26]. However, those measures try to get rid of correlation in the selected gene subset all together, while such correlation information may not be totally uninformative. For example, in study [27], Xu and Zhang suggested that such correlation itself may be used as predictor of sample class.

In our algorithm, the correlation of selected genes is minimized in a more moderate manner. That is, through the use of an inductive algorithm the correlation of the selected genes is minimized implicitly. Our objective is to minimize

the redundancy while keeping the usefulness. After all, the reason of minimizing gene correlation is to obtain higher classification accuracy. In our experiment, we compare the correlation of the most frequently selected genes using the proposed method to those obtained by using filter algorithms directly. The calculation of the average correlation is as follows:

$$P(x_i, x_j) = \frac{\sum x_i x_j - \frac{(\sum x_i)(\sum x_j)}{m}}{\sqrt{(\sum x_i^2 - \frac{(\sum x_i)^2}{m})(\sum x_j^2 - \frac{(\sum x_j)^2}{m})}} \quad (7)$$

$$Average\ Correlation = \frac{2 \times \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sqrt{P(x_i, x_j)^2}}{n(n-1)} \quad (8)$$

where x_i and x_j denote the expression level of two different genes in the selection result. $P(.)$ is the function of Pearson Product-Moment correlation coefficient. m denotes the total samples, while n denotes the total number of genes considered.

D. Cross Validation and Stability

Cross validation is one of the most popular evaluation strategies. When employing cross validation, the dataset is commonly divided into several folds. Taking n-fold cross validation as an example, while $n-1$ folds are used to train the classifier the remaining fold is used to evaluate the classification power of the classifier on unseen data. After each fold is used to evaluate the classification accuracy in an orderly fashion, the classification accuracy of the classifier is then calculated by averaging the classification accuracy of each fold. Cross validation is a robust evaluation method. It is particularly useful when the sample size of the dataset is small because the dataset is efficiently reused for measuring the error rate, resulting a more objective evaluation results [28]. In this work, 5-fold stratified cross validation is utilized.

With the consideration of the stochastic nature of GA, each GA based method is conducted with 5 independent runs, producing 5 independent cross validation results. The final results are given in the form of "mean $\pm$ standard deviation" ($\mu \pm \sigma$). The stability of each GA based method can then be assessed by comparing the value of the standard deviation σ.

IV. Results

A. Sample Classification Accuracy

For comparison purpose, we experimented using GA wrapper and filters (χ^2-test and Information Gain) separately for feature selection. The classification results of each individual selection method is compared with those obtained with GAEF.

Tables III–V give classification accuracy details of each method, using Colon, Breast and MLL microarray datasets, respectively [20], [21], [22]. Specifically, the second and the third columns of each table detail the sample classification using χ^2-test and Information Gain selected gene sets (from size of 5 to 25 with a step of 5) with kNN classifier. The fourth column shows the classification of GA wrapper selected gene sets with kNN classifier. And the last two columns provide the classification results obtained by using GAEF selected gene sets with embedded filters of χ^2-test and Information Gain, respectively. Each GA based selection method is averaged with 5 independent runs.

As can be readily observed, in most cases GAEF identified gene subsets produced better sample classification results. With Colon dataset, using GAEF selected gene sets we obtained the average sample classification accuracy of 83.36 and 82.46 using embedded filters of χ^2-test and Information Gain, respectively. Compared with using these two filter algorithms directly, which produced the average classification accuracy of 73.67 and 75.98, the improvement is significant. Similar results can be observed in the analysis results of both Breast dataset and MLL dataset. The classification accuracy of GAEF identified gene subsets for Breast dataset are 68.03 and 67.01, while for MLL dataset the figures are 86.28 and 87.89, using χ^2-test and Information Gain, respectively. In comparison, the classification accuracy produced with the two filter algorithms directly are 63.69 and 63.54 for Breast dataset, and 82.69 and 82.15 for MLL dataset. Although not so phenomenal compared with that of Colon dataset, the improvement is still obvious. Essentially, χ^2-test and Information Gain produced similar classification results regardless been used solely or embedded in GA. By applying GA wrapper directly for gene subsets selection, the average classification accuracy are 71.79 for Colon data, 64.46 for Breast data and 82.06 for MLL data. The results are similar to those achieved by applying filter based gene selection and sample classification.

With regard to the stability of the classification results, when applying GA wrapper directly, the variance σ is usually quite large, which is consistent with our assumption that GA is unstable and prone to local optimal with high feature-to-sample ratio data. This phenomenon is evident from the analysis results of all of the three microarray datasets. For Colon, Breast and MLL datasets, the average variance of the classification results are 5.29, 5.02 and 3.52, respectively (column 4 of Tables III–V).

In contrast, results yielded by using GAEF model are with smaller variance (column 5 and 6 of Tables III–V). With Colon dataset, the average variance of the classification result is 2.47 for the χ^2-test embedded model and 2.45 for the Information Gain embedded model. With Breast dataset, the average variance of the classification result is 2.77 for the χ^2-test embedded model and 2.82 for the Information Gain embedded model. As to the MLL dataset, the figures are 2.05 and 2.50 for the χ^2-test embedded model and the Information Gain embedded model, respectively. These results suggest that by adding an embedded filter, we are able to improve the stability of GA based feature selection algorithms.

B. Comparison of GA/KNN

Table VI provides the 5-fold stratified cross validation results utilizing kNN with the gene sets identified by GA/KNN algorithm [5], using identical divisions of training and test sets as that of GAEF. When applying GA/KNN, the chromosome length of 10 is used, and the number of near-optimal combinations selected is 1000. Majority voting and the $k = 3$ of

TABLE III
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF COLON DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	71.22	72.11	70.55 ± 5.79	81.22 ± 2.71	80.53 ± 2.17
10-gene	72.55	76.11	73.37 ± 5.40	84.62 ± 2.61	81.82 ± 2.36
15-gene	73.26	73.89	71.07 ± 5.96	83.02 ± 1.55	83.55 ± 3.44
20-gene	76.22	78.89	69.38 ± 4.47	83.58 ± 2.61	83.78 ± 0.67
25-gene	75.11	78.89	74.60 ± 4.82	84.34 ± 2.89	82.60 ± 3.61

TABLE IV
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF BREAST DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	57.14	55.34	64.24 ± 4.70	66.07 ± 4.49	62.51 ± 2.66
10-gene	66.11	62.54	62.58 ± 4.39	70.17 ± 3.08	66.75 ± 3.89
15-gene	68.29	66.30	64.44 ± 5.67	66.56 ± 1.92	68.99 ± 3.69
20-gene	61.99	64.28	65.61 ± 5.86	67.51 ± 1.80	69.61 ± 1.15
25-gene	64.96	69.24	65.43 ± 4.46	69.85 ± 2.58	67.20 ± 2.72

TABLE V
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF MLL DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	80.00	79.33	74.44 ± 4.53	84.47 ± 1.58	88.31 ± 2.93
10-gene	84.00	81.11	83.74 ± 4.27	86.84 ± 1.59	86.29 ± 1.07
15-gene	83.11	81.11	85.64 ± 2.16	85.49 ± 2.39	87.64 ± 3.45
20-gene	82.11	85.78	80.87 ± 5.39	87.69 ± 2.56	87.00 ± 1.32
25-gene	84.22	83.44	85.60 ± 1.24	86.89 ± 2.13	90.20 ± 3.74

the k-nearest neighbor are adopted. It should be noted that the cut off of the selection threshold for the chromosomes of GA/KNN depends on the characteristics of the datasets. Different thresholds are used according to its classification power on different datasets. Specifically, the threshold for the Colon dataset is that 4 samples are incorrectly classified at most. For Breast dataset and MLL dataset the thresholds are 5 and 2 samples are incorrectly classified at most.

Comparing the results produced by our GAEF method with those obtained from GA/KNN algorithm, we can conclude that GAEF method is comparable or even superior in several cases to GA/KNN algorithm in terms of gene selection for microarray data classification.

TABLE VI
CLASSIFICATION ACCURACY OF GA/KNN ALGORITHM

GA/KNN			
Feature size	Colon	Breast	MLL
5-gene	74.78	66.63	86.22
10-gene	76.55	68.63	87.89
15-gene	83.11	69.81	85.45
20-gene	83.11	69.40	88.11
25-gene	83.11	69.36	87.00

C. Correlation of Frequently Identified Genes

Tables 5-7 give the top-5 most frequently selected genes of Colon dataset, Breast dataset and MLL dataset, respectively. Specifically, Hsa.37937 and Hsa.692 in Colon dataset, Contig7258_RC in Breast dataset, and 40763_at, 32847_at and 35614_at in MLL dataset are the most frequently selected genes using different methods. Each table is subdivided into four sub-tables corresponding to the gene selection methods of using χ^2-test and Information Gain directly, and using they as GA embeds. Selected genes in each sub-table are pairwised with each other for Pearson Product-Moment correlation coefficient calculation. It is evident that the average Pearson correlation coefficients of GAEF selected genes are generally lower than those identified directly by filter algorithms. Nevertheless, GAEF algorithm did not attempt to reduce the correlation between each pair of genes to the minimum. This is because as demonstrated in empirical study [27] correlation among genes does not necessarily be totally useless. On the contrary, it may facilitate the sample classification in some degree.

V. DISCUSSION

One major problem of applying GA based wrapper for feature selection of high dimensional dataset is that the algorithm is prone to overfitting and often quickly converge to a local optimal solution. Therefore, the selected feature subsets often perform poor on unseen data classification. This phenomenon is evident in our experimental results that using GA with kNN classifier for gene selection and data classification of microarrays. By embedding an filtering algorithm into the GA wrapper, we are able to minimize the overfitting of the resulting hybrid algorithm in feature selection and sample classification processes. The explanation of this improvement is straightforward. By adding a filter algorithm, candidate

TABLE VII
TOP-5 MOST FREQUENTLY SELECTED GENES OF COLON DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN					
Gene id	Hsa.627	Hsa.8147	Hsa.37937	Hsa.692(f765)	Hsa.1832
Hsa.627	-				
Hsa.8147	-0.277	-			
Hsa.37937	-0.315	0.815	-		
Hsa.692(f765)	-0.298	0.794	0.761	-	
Hsa.1832	-0.283	0.815	0.886	0.725	-
Average Pearson correlation coefficient: 0.597					

Info+kNN					
Gene id	Hsa.627	Hsa.8147	Hsa.37937	Hsa.692(f765)	Hsa.692(f267)
Hsa.627	-				
Hsa.8147	-0.277	-			
Hsa.37937	-0.315	0.815	-		
Hsa.692(f765)	-0.298	0.794	0.761	-	
Hsa.692(f267)	-0.285	0.886	0.739	0.851	-
Average Pearson correlation coefficient: 0.602					

GA+χ^2+kNN					
Gene id	Hsa.692(f267)	Hsa.37937	Hsa.601	Hsa.692(f765)	Hsa.3306
Hsa.692(f267)	-				
Hsa.37937	0.739	-			
Hsa.601	-0.243	-0.237	-		
Hsa.692(f765)	0.851	0.761	-0.279	-	
Hsa.3306	-0.223	-0.147	0.665	-0.189	-
Average Pearson correlation coefficient: 0.433					

GA+Info+kNN					
Gene id	Hsa.5971	Hsa.41323	Hsa.692(f245)	Hsa.2451	Hsa.2291
Hsa.5971	-				
Hsa.41323	0.705	-			
Hsa.692(f245)	-0.192	-0.120	-		
Hsa.2451	0.567	0.582	-0.155	-	
Hsa.2291	-0.178	-0.012	0.571	0.145	-
Average Pearson correlation coefficient: 0.323					

features are no longer evaluated by the sole criterion of the classification accuracy of a given inductive algorithm, but regulated by the filter algorithm implicitly. Hence, the hybrid algorithm itself does not seek for high classification accuracy of training dataset blindly and greedily but take into consideration of other characteristics of the data as well. In this way, different selection criteria are balanced, and the "importance" of a given feature to the dataset is evaluated from multiple aspects.

VI. CONCLUSIONS

Filter and wrapper algorithms are commonly treated as competitors in feature selection of datasets with high dimension. Several studies have been conducted to compare the strengths and the weaknesses of each method in microarray data analysis context [12], [29], [30], but few of them attempted to integrate individual methods. In this study, instead of treating each method as competitor, we take the effort to integrate them as components of a higher system. The proposed hybrid model called GAEF utilizes GA to pre-select large feature subsets and invokes a filter selector to further identify highly differential feature subsets for accurate sample classification. This model is tested on both binary-class dataset and multi-class dataset. The experimental results suggest that such an embedded two-stage feature selection model be able to improve sample classification accuracy as well as the stability of the selection results.

ACKNOWLEDGMENT

We would like to thank Chuan Cao from Thinkit Speech Lab., Institute of Acoustics, Chinese Academy of Sciences for valuable discussion and suggestion.

REFERENCES

[1] R.L. Somorjai, B. Dolenko and R. Baumgartner, "Class prediction and discovery using gene microarray and protenomics mass spectroscopy data: curses, caveats, cautions," *Bioinformatics*, vol. 19, pp. 1484-1491, 2003.

[2] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine learning Research*, vol. 3, pp. 1157-1182, 2003.

[3] S. Theodoridis and K. Koutroumbas, *Pattern Recognition (Third Edition)*. Elsevier Press, 2006.

[4] A.L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, pp. 245-271, 1997.

[5] L. Li, C.R. Weinberg, T.A. Darden and L.G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method," *Bioinformatics*, vol. 17, pp. 1131-1142, 2001.

[6] T. Jirapech-Umpai and S. Aitken, "Feature selection and classification for microarray data analysis: evolutionary methods for identifying predictive genes," *BMC Bioinformatics*, vol. 6, pp. 148, 2005.

[7] P. Yang and Z. Zhang, "Hybrid Methods to Select Informative Gene Sets in Microarray Data Classification," In: *Proceedings of the 20th Australian Joint Conference on Artificial Intelligence, LNAI 4830*, pp. 811-815, 2007.

[8] M. Kudo and J. Sklansky, "Comparison of algorithms that select features for pattern classifiers," *Pattern Recognition*, vol. 33, pp. 25-41, 2000.

[9] L. Kuncheva and L. Jain, "Designing classifier fusion system by genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, pp. 327-336, 2000.

TABLE VIII
TOP-5 MOST FREQUENTLY SELECTED GENES OF BREAST DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN					
Gene id	Contig31033_RC	Contig24311_RC	Contig15031_RC	Contig7258_RC	Contig30098_RC
Contig31033_RC	-				
Contig24311_RC	-0.069	-			
Contig15031_RC	0.679	0.175	-		
Contig7258_RC	-0.071	0.999	0.175	-	
Contig30098_RC	-0.305	0.277	-0.233	0.476	-
Average Pearson correlation coefficient: 0.346					

Info+kNN					
Gene id	Contig31033_RC	Contig7258_RC	Contig24311_RC	Contig15031_RC	NM_003344
Contig31033_RC	-				
Contig7258_RC	-0.071	-			
Contig24311_RC	-0.069	0.999	-		
Contig15031_RC	0.679	0.175	0.175	-	
NM_003344	0.522	0.154	0.156	0.531	-
Average Pearson correlation coefficient: 0.353					

GA+χ^2+kNN					
Gene id	AL080059	NM_020974	Contig7258_RC	AF073519	NM_014554
AL080059	-				
NM_020974	-0.462	-			
Contig7258_RC	0.458	-0.602	-		
AF073519	0.247	-0.414	0.391	-	
NM_014554	-0.130	-0.008	-0.204	0.007	-
Average Pearson correlation coefficient: 0.292					

GA+Info+kNN					
Gene id	NM_006115	NM_005744	NM_003258	Contig52554_RC	NM_016185
NM_006115	-				
NM_005744	0.363	-			
NM_003258	0.518	0.307	-		
Contig52554_RC	0.056	-0.248	-0.206	-	
NM_016185	0.360	0.373	0.692	-0.174	-
Average Pearson correlation coefficient: 0.329					

[10] Y. Saeys, I. Inza and P. Larranage, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, pp. 2507-2517, 2007.

[11] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield and E. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, vol. 286, pp. 531-537, 1999.

[12] H. Liu, J. Li and L. Wang, "A Comparative Study on Feature Selection and Classification Methods Using Gene Expression Profiles and Proteomic Patterns," *Genome Informatics*, vol. 13, pp. 51-60, 2002.

[13] Y. Su, T. Murali, V. Pavlovic, M. Schaffer and S. Kasif, "RankGene: Identification of Diagnostic Genes Based on Expression Data," *Bioinformatics*, vol. 19, pp. 1578-1579, 2003.

[14] R. Kohavi and G. John, "Wrapper for feature subset selection", *Artificial Intelligence*, vol. 97, pp. 273-324, 1997.

[15] J. Jaeger, R. Sengupta, W. Ruzzo, "Improved Gene Selection for Clssification of Microarrays," *Pacific Symposium on Biocomputing*, vol. 8, pp. 53-64, 2003.

[16] P. Yang, B. Zhou, Z. Zhang and A. Zomaya, " A multi-filter enhanced genetic ensemble system for gene selection in microarray data," to appear in APBC 2010.

[17] Z. Zhang, P. Yang, X. Wu and C. Zhang, "An agent-based hybrid system for microarray data analysis," *IEEE Intelligent Systems*, vol. 24, no. 5, pp. 53-63, 2009.

[18] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," In: *Proceedings of 18th International Conference on Machine Learning*, pp. 74-81, 2001.

[19] T. Khoshgoftaar, C. Seiffert and J. Hulse, "Hybrid Sampling for Imbalanced Data," In: *Proceedings of the IEEE International Conference on Information Reuse and Integration*, pp. 202-207, 2008.

[20] U. Alon, N. Barkai, D. Notterman, K. Gish, S. Ybarra, D. Mack and A. Levine, "Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays," *PNAS*, vol. 96, pp. 6745-6750, 1999.

[21] L. van't Veer, H. Dai, M. van de Vijver, Y. He, A. Hart, M. Mao, H. Peterse, K. van der Kooy, M. Marton, A. Witteveen, G. Schreiber, R. Kerkhoven, C. Roberts, P. Linsley, R. Bernards and S. Friend, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, pp. 530-536, 2002.

[22] S. Armstrong, J. Staunton, L. Silverman, R. Pieters, M. den Boer, M. Minden, S. Sallan, E. Lander, T. Golub and S. Korsmeyer, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genetics*, vol. 30, pp. 41-47, 2001.

[23] I. Witten and M. Frank, *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*. Elsevier, 2005.

[24] J. Hua, Z. Xiong, J. Lowey, E. Suh and E. Dougherty, "Optimal number of features as a function of sample size for variuos classification rules," *Bioinformatics*, vol. 21, pp. 1509-1515, 2005.

[25] Z. Cai, R. Goebel, M. Salavatipour and G. Lin, "Selecting dissimilar genes for multi-class classification, an application in cancer subtyping," *BMC Bioinformatics*, vol. 8, pp. 206, 2007.

[26] B. Hanczar, M. Courtine, A. Benis, C. Hennegar, K. Clement and J. Zucker, "Improving classification of microarray data using prototype-based feature selection," *SIGKDD Explorations*, vol. 5, pp. 23-30, 2003.

[27] X. Xu and A. Zhang, "Virtual gene: Using correlations between genes to select informative genes on microarray datasets," *Transaction on Computational System Biology II, LNBI 3680*, pp. 138-152, 2005.

[28] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, vol. 2, pp. 1137-1143, 1995.

[29] I. Inza, P. Larranage, R. Blanco and A. Cerrolaza, "Filter versus wrapper gene selection approaches in DNA microarray domains," *Artificial Intelligence in Medicine*, vol. 31, pp. 91-103, 2004.

[30] J. Lee, J. Lee, M. Park and S. Song, "An extensive comparison of recent classification tools applied to microarray data," *Computational Statistics & Data Analysis*, vol. 48, pp. 869-885, 2005.

TABLE IX
TOP-5 MOST FREQUENTLY SELECTED GENES OF MLL DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN					
Gene id	36239_at	35164_at	32847_at	40763_at	39318_at
36239_at	-				
35164_at	0.580	-			
32847_at	0.712	0.745	-		
40763_at	-0.159	-0.152	-0.193	-	
39318_at	0.626	0.578	0.629	-0.142	-
Average Pearson correlation coefficient: 0.452					

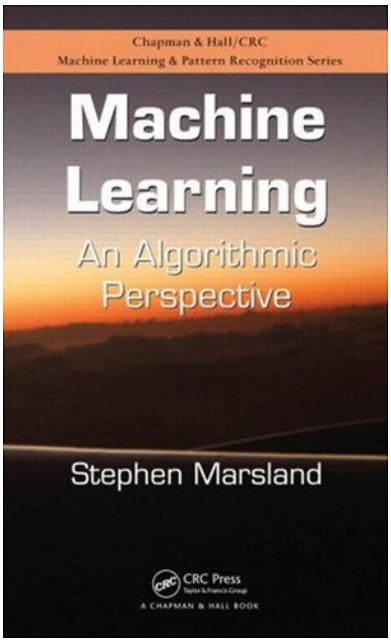
Info+kNN					
Gene id	36239_at	35164_at	32847_at	40763_at	37539_at
36239_at	-				
35164_at	0.580	-			
32847_at	0.712	0.745	-		
40763_at	-0.159	-0.152	-0.193	-	
37539_at	0.688	0.625	0.669	-0.152	-
Average Pearson correlation coefficient: 0.468					

GA+χ^2+kNN					
Gene id	31886_at	41747_s_at	35164_at	266_s_at	40763_at
31886_at	-				
41747_s_at	0.304	-			
35164_at	0.356	0.457	-		
266_s_at	0.602	0.528	0.650	-	
40763_at	-0.109	0.136	-0.152	-0.163	-
Average Pearson correlation coefficient: 0.346					

GA+Info+kNN					
Gene id	36122_at	32847_at	35260_at	35614_at	1914_at
36122_at	-				
32847_at	0.413	-			
35260_at	0.494	0.732	-		
35614_at	0.334	0.661	0.738	-	
1914_at	-0.169	-0.279	-0.170	-0.213	-
Average Pearson correlation coefficient: 0.420					

Machine Learning: An Algorithmic Perspective

STEPHEN MARSLAND



REVIEWED BY J.P. LEWIS

When several good books on a subject are available the pedagogical style of a book becomes more than a secondary consideration. This is particularly true in the case of mathematical and algorithmic subjects such as machine learning, where the level of formal rigor is a consideration. Peter Naur spent a portion of his career considering this issue. As the 'N' in the Backus-Naur formalism (BNF), one would expect Naur to champion the role of correct formal derivations in learning mathematical topics. On the contrary, in several of the chapters in his books [1], [2] Naur shows that formal approaches are at best incidental, and more often detrimental, to the learning and understanding of subjects that involve formal systems. Said differently, humans are much better at learning by example and experimentation than by attempting to follow proofs. While Naur demonstrates this in studies and observation of beginning programmers, he also illustrates the problem in a professional setting: mathematical proofs in peer-reviewed papers as short as several pages have been found to have errors after publication.

Marsland's new book *Machine Learning: An Algorithmic Perspective* takes a decisive approach to this issue, based on algorithmic experimentation. Each topic is motivated by creative examples (such as learning to dance at a nightclub) and then presented both mathematically and algorithmically. Many of the exercises require exploring and revising the code fragments in the book. There are mathematical illustrations, but no explicit proofs.

The book's example-based approach evidently effects the ordering of topics, which is occasionally odd from the perspective of someone who already has the big picture of the field. For example, the curse of dimensionality is a consideration for all machine learning approaches and thus might logically be introduced in an abstract overview chapter along with Maximum Likelihood, MAP, and so on. Instead, Marsland's book introduces it as it arises in a discussion of spline and radial basis interpolation.

The algorithmic examples in the book use the Numpy and Scipy environments in the Python language. For those not familiar, Python+Numpy is rapidly taking a place along side Matlab for the rapid prototyping of mathematical algorithms. Appealing aspects of Python are that it is a well designed and structured language with broad adoption, and the fact that it is free and open source. Experienced Matlab programmers will note many operations with similar names and behavior [3], and Numpy shares Matlab's expressiveness in representing linear algebra computations. For example, a linear discriminant example in the book is 15 lines of code, and the kernel PCA algorithm is 14 lines. On the negative side, Numpy and Scipy are still rapidly evolving and somewhat immature.

Subjects covered by *Machine Learning: An Algorithmic Perspective* include linear discriminants, neural networks, radial basis functions and splines, support vector machines, regression trees, basic probability theory and the bias-variance tradeoff, classification by neighbor neighbors, mixture models and EM, ensemble techniques, k-means, vector quantization and self organising maps, dimensionality reduction, MDS, and manifold learning, genetic algorithms, reinforcement learning, hidden Markov models, and MCMC. The book has a chapter introducing Python for Matlab and R users. A chapter on optimization initially seems out of place, though it sets the context for several other chapters.

Although several excellent and tested books on machine learning exist (e.g. [4], [5], [6]), Marsland's text stands out as the only book suited to undergraduate or Masters level teaching or equivalent self-instruction, and I expect that it will shine in this role. The book could be improved with the addition of a concluding summary chapter wherein fundamental concepts (ML, MAP, etc.) are revisited in their broadest context. After reading Naur's work [1], [2] I now believe that *all* books in the algorithmic and mathematical areas should contain strong reader advisories to watch out for typos and errors – particularly in first editions such as this.

REFERENCES

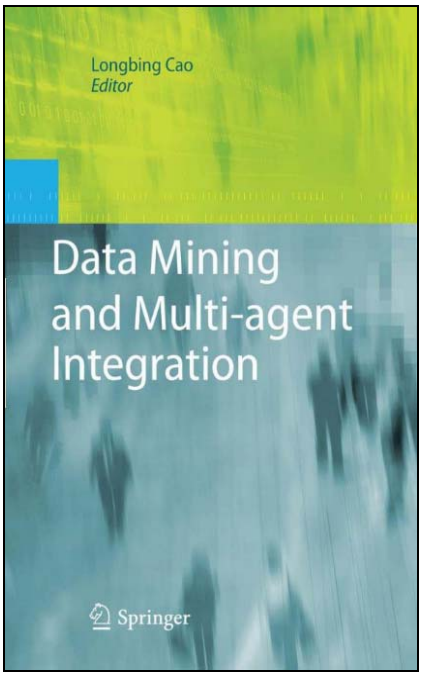
[1] P. Naur, *Computing: A Human Activity*, ACM Press 1992
[2] P. Naur, *Knowing and the Mystique of Logic and Rules*, Kluwer Academic, 1995
[3] http://www.scipy.org/NumPy_for_Matlab_Users
[4] C. Bishop, *Pattern Recogntion and Machine Learning*, Springer, 2007
[5] R. Duda, P. Hart, D. Stork, *Pattern Classification*, Wiley, New York, 2001
[6] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer Verlag, New York, 2009

ABOUT THE REVIEWER:

JOHN LEWIS: Weta Digital and Massey University. Contact him at zilla@computer.org, and see http://scribblethink.org for more information.

Data Mining and Multi-agent Integration

BY LONGBING CAO (EDITOR) – ISBN: 978-1-44190-521-5



REVIEWED BY ANDREAS SYMEONIDIS

Nowadays, both Agent Technology (in the form of individual agents, multi-agent systems, and societies) and Data Mining technologies have reached an acceptable level of maturity, and, each one alone, has its own scope and applicability. Naturally enough, a fruitful synergy of the two technologies has already been proposed, that would combine the benefits of both worlds and would offer computer scientists with new tools in their effort to build more sophisticated software systems. This integration is at least bidirectional: on the one hand, the ***Agent-driven Data Mining*** approach identifies that, since Data Mining comprises a number of discrete, nevertheless dependent tasks, agents can be employed in order to regulate, control, and organize the, potentially distributed activities involved in the knowledge discovery process. On the other hand, the ***Data Mining-driven Agents*** approach argues that knowledge hidden in voluminous data repositories, social networks and web transactions, can be extracted by data mining and provide the inference mechanisms or simply the behavior of agents and multi-agent systems. In other words, the discovered knowledge nuggets may constitute the building blocks of agent intelligence.

However, while the pieces are already there, the puzzle is far from complete. Coupling the two technologies does not come seamlessly, since the inductive nature of data mining imposes logic limitations and hinders the application of the extracted knowledge on deductive systems, such as multi-agent systems. One should take all the relevant limitations and considerations into account, in order to provide a pathway for employing data mining techniques in order to augment agent intelligence.

"***Data Mining and MultiAgent Integration***", edited by Longbing Cao, one of the experts in the field and founder of the AMII Special Interest Group, focuses on exactly this synergy between Agent Systems and Data Mining. ***Agent Mining***, as defined in the latest bibliography, is expected to create innovative interaction and integration tools and services, and unify results under one new technology.

"*Data Mining and MultiAgent Integration*" attempts to present the latest attempts and trends in agent mining, rather than to cover the field in a dogmatic manner. To this, it has been divided into three parts. Part I provides an overview on the integration of agents and data mining, giving an inside view on the expected benefits and practical problems addressed upon integration. Part II presents a number of representative data mining-driven agents, carefully selected in order to cover a wide scope of applications and domains. Finally, Part III focuses on Agent-driven data mining, depicting the state-of-the-art and challenges through a number of research cases.

Part I is organized in three chapters. Chapter 1, written by the editor, serves as a synopsis of the content to follow. It pinpoints the main driving forces of the new technology, and summarizes the disciplinary framework, case studies, trends and directions currently in the field. Based on studying issues related to agent-driven data mining, data mining-driven agents and their interdependence, Chapter 1 acknowledges the potential of the new technology and depicts the theoretical and practical issues that the integration has. Chapter 2 follows a bottom-up approach. Through two pilot case demonstrators, a MAS for supporting manual annotation for DNA function prediction and a MAS to assist in digital forensics, an effort is made to clarify the benefits of Data mining-driven agents. Finally, Chapter 3, following again a bottom–up approach, performs a thorough survey and provides evidence on the exploitation of agents in distributed data mining, in terms of significance, system architectures, and research trends.

Having sketched the bigger picture in Part I, Part II provides the reader with design and implementation details in a variety of problems solved through the use of data mining-driven agents. Chapter 4 presents an agent system for web session clustering, based on swarm intelligence. Chapter 5 focuses on improving agent intelligence through discovering temporal agent behavior patterns, while Chapter 6 employs Web usage and Web structure mining in order to analyze user interaction habits and predict user behavior. In the same context. Chapter 7 presents a distributed recommender system and the methodology employed for sharing and

generating improved recommendations. Chapter 8 integrates a multi-class supervised classification algorithm with agent technology in the domain of network intrusion detection, where a multi-agent design methodology is coupled with a highly accurate, fast, and lightweight *PCC* Classifier and *CRSPM* schemes. Chapter 9 proposes genetic algorithms for data extraction based on the evolution and grammatical composition of regular expressions, while Chapter 10 employs a weight-driven network module in order to increase projection of knowledge nodes in a system, enrich their repositories and stimulate the corresponding user communities. Chapter 11 defines the notion of Goal mining and utilizes it in order to extract knowledge on user goals, residing in common query logs. Finally, Chapter 12 concludes the collection of manuscripts on data mining driven agents, discussing an agent-based diagnostic workbench equipped with classification capabilities, in order to support real medical diagnosis.

Part III addresses the complementary to Part II approach. First, in Chapter 13, EMADS is presented a framework that extends current work in data mining frameworks and employs different types of cooperating agents in order to perform complex classification and association rule extraction tasks. Next, Chapter 14 proposes a multi-agent system for dealing with online hierarchical clustering of streaming data, while Chapter 15 proposes two models for Agents-driven clustering of large datasets: a divide-and-conquer method and a data-dependent method. Chapter 16 introduces MACC, a multi-ant colony and multi-objective clustering algorithm that handles distributed data, through the assignment of specific objectives to different colonies and the synthesis of all results. Chapter 17 proposes an interactive environment for psychometrics diagnostics, where agents monitor user actions and perform data mining on them, in order to discover potentially interesting information. Chapter 18, following a completely different approach, implements a two-level agent system that performs association rule mining and frequency based mining on log files, in order to discover firewall policy rules, subsequently employed to detect intra- and inter- firewall anomalies. Chapter 19 employs simple data mining and statistical analysis on a heterogeneous data grid and proposes a game theory-based multi-agent model for competitive knowledge extraction, hierarchical knowledge mining, and Dempster-Shafer result combination. Chapter 20 discusses a normative multi-agent enriched data mining architecture and ontology framework to support citizens in accessing services provided by public authorities. Chapter 21 works on the combination of static and dynamic agent societies assigned with the task of identifying (though classification) groups of users with common interests. Finally, Chapter 22, the last Chapter of Part III and the book, describes an agent -based video contents identification scheme using a watermark based filtering technique, aiming to prevent a user from uploading illegal video content into a user defined web storage.

Overall, reading this book is a pleasant surprise. The editor, having satisfied good quality contributions from the authors, has succeeded in producing a book that may serve as the basis for further probing. The objectives and expected outcome of reading the book become clear from the very beginning, the structure is concise and the pilot cases provided with respect to the two established lines of work in agent mining are representative. Having read "*Data Mining and MultiAgent Integration*", the user is triggered to explore the issues related to the coupling of the two technologies, deciding to follow any agent mining path, either one already established, or a completely new one. It is with interest that we expect the L. Cao et al monograph, *"Agents and Data Mining: Interaction and Integration"*.

ABOUT THE REVIEWER:

ANDREAS SYMEONIDIS: Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Greece. Contact him at: asymeon@eng.auth.gr and see http://users.auth.gr/symeonid for more information.

RELATED CONFERENCES, CALL FOR PAPERS/PARTICIPANTS

TCII Sponsored Conferences

.

WI 2010
The 2010 IEEE/WIC/ACM International Conference on Web Intelligence
Toronto, Canada
August 31- September 3, 2010
http://www.yorku.ca/wiiat10/

Web Intelligence (WI) has been recognized as a new direction for scientific research and development to explore the fundamental roles as well as practical impacts of Artificial Intelligence (AI) (E.g., knowledge representation, planning, knowledge discovery and data mining, intelligent agents, and social network intelligence) and advanced Information Technology (IT) (E.g., wireless networks, ubiquitous devices, social networks, semantic Web, wisdom Web, and data/knowledge grids) on the next generation of Web-empowered products, systems, services, and activities. It is one of the most important as well as promising IT research fields in the era of Web and agent intelligence. The 2010 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2010) will be jointly held with the 2010 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2010). The IEEE/WIC/ACM 2010 joint conferences are organized by York University, Toronto Canada, and sponsored by IEEE Computer Society Technical Committee on Intelligent Informatics (TCII), Web Intelligence Consortium (WIC), and ACM-SIGART.

WI 2010 is planned to provide a leading international forum for researchers and practitioners (1) to present the state-of-the-art of WI technologies; (2) to examine performance characteristics of various approaches in Web-based intelligent information technology; and (3) to cross-fertilize ideas on the development of Web-based intelligent information systems among different domains. By idea-sharing and discussions on the underlying foundations and the enabling technologies of Web intelligence, WI 2010 will capture current important developments of new models, new methodologies and new tools for building a variety of embodiments of Web-based intelligent information systems. A doctoral mentoring program will be also organized.

IAT 2010
The 2010 IEEE/WIC/ACM International Conference on Intelligent Agent Technology
Toronto, Canada
August 31- September 3, 2010
http://www.yorku.ca/wiiat10/

The 2010 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2010) will be jointly held with the 2010 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2010). The IEEE/WIC/ACM 2010 joint conferences are organized by York University, Toronto Canada, and sponsored by IEEE Computer Society Technical Committee on Intelligent Informatics (TCII), Web Intelligence Consortium (WIC), and ACM-SIGART.

IAT 2010 will provide a leading international forum to bring together researchers and practitioners from diverse fields, such as computer science, information technology, business, education, human factors, systems engineering, and robotics, to (1) examine the design principles and performance characteristics of various approaches in intelligent agent technology, and (2) increase the cross fertilization of ideas on the development of autonomous agents and multi-agent systems among different domains. By encouraging idea-sharing and discussions on the underlying logical, cognitive, physical, and sociological foundations as well as the enabling technologies of intelligent agents, IAT 2010 will foster the development of novel paradigms and advanced solutions in agent based computing. The joint organization of IAT 2010 and WI 2010 will provide an opportunity for technical collaboration beyond the two distinct research communities. A doctoral mentoring program will be also organized.

ICDM 2010
The Tenth IEEE International Conference on Data Mining
Sydney, Australia
December 13-17, 2010
http://datamining.it.uts.edu.au/icdm10/

The IEEE International Conference on Data Mining series (ICDM) has established itself as the world's premier research conference in data mining. It provides an international forum for presentation of original research results, as well as exchange and dissemination of innovative, practical development experiences. The conference covers all aspects of data mining, including algorithms, software and systems, and applications. In addition, ICDM draws researchers and application developers from a wide range of data mining related areas such as statistics, machine learning, pattern recognition, databases and data warehousing, data visualization, knowledge-based systems, and high performance computing. By promoting novel, high quality research findings, and innovative solutions to challenging data mining problems, the conference seeks to continuously advance the state-of-the-art in data mining. Besides the technical program, the conference features workshops, tutorials, panels and, since 2007, the ICDM data mining contest.

Topics related to the design, analysis and implementation of data mining theory, systems and applications are of interest. These include, but are not limited to the following areas: data mining foundations, mining in emerging domains, methodological aspects and the KDD process, and integrated KDD applications, systems, and experiences. A detailed listing of specific topics can be found at the conference website.

ICDM proceedings are published by the IEEE Computer Society Press. A selected number of ICDM accepted papers will be expanded and revised for possible inclusion in the KAIS journal (Knowledge and Information Systems, by Springer-Verlag) each year. This will be mentioned in all calls for papers of the ICDM conference. KAIS will publish the calls for papers of the ICDM conferences once a year without any charges, by the conference

organizers' request, to publicize the mutual support for the success of ICDM and KAIS.

Related Conferences

AAMAS 2010
The Ninth International Conference on Autonomous Agents and Multi-Agent Systems
Toronto, Canada
May 10-14, 2010
http://www.cse.yorku.ca/AAMAS2010/

AAMAS 2010, the 9th International Conference on Autonomous Agents and Multiagent Systems will take place at the Sheraton Centre Toronto Hotel in downtown Toronto Canada, on May 10-14 2010. AAMAS is the premier scientific conference for research on autonomous agents and multiagent systems.

AAMAS is the leading scientific conference for research in autonomous agents and multiagent systems. The AAMAS conference series was initiated in 2002 by merging three highly-respected meetings: International Conference on Multi-Agent Systems (ICMAS); International Workshop on Agent Theories, Architectures, and Languages (ATAL); and International Conference on Autonomous Agents (AA). The aim of the joint conference is to provide a single, high-profile, internationally-respected archival forum for scientific research in the theory and practice of autonomous agents and multiagent systems. AAMAS 2010 is the Ninth conference in the AAMAS series, following enormously successful previous conferences, and will be held at the Sheraton Centre Toronto Hotel in downtown Toronto. See the IFAAMAS wb site for more information on the AAMAS conference series.

AAMAS 2010 seeks high-quality submissions of full papers, limited to 8 pages in length. Submissions will be rigorously peer reviewed and evaluated on the basis of originality, soundness, significance, presentation, understanding of the state of the art, and overall quality of their technical contribution. Reviews will be double blind; authors must avoid including anything that can be used to identify them. Where submission is for full (8 page) papers only, in some cases they may be accepted as 2 page extended abstracts. Please see the formatting instructions.

In addition to submissions in the main track, AAMAS is soliciting papers in two special tracks on robotics, and on virtual agents (see below). The review process for the special tracks will be the same as for the main track, but with specially-selected program committee members. Special Track on Robotics (Chair: Michael Beetz): Papers on theory and applications concerning single and multiple robots will be welcome, namely those focusing on real robots interacting with their surrounding environments. The goal is to foster interaction between researchers on agent and robotics systems, so as to provide a cradle for cross-fertilization of concepts from both fields. Special Track on Virtual Agents (Chair: Stacy Marsella): Virtual agents are embodied agents in interactive virtual or physical environments that emulate human-like behavior. We encourage papers on the design, implementation, and evaluation of virtual agents as well as challenging applications featuring them. The goal is to provide an opportunity for interaction and cross-fertilization between the AAMAS community and researchers working on virtual agents and to strengthen links between the two communities.

SDM 2010
2010 SIAM International Conference on Data Mining
Columbus, Ohio, USA
April 29-May 1, 2010
http://www.siam.org/meetings/sdm10/

Data mining is an important tool in science, engineering, industrial processes, healthcare, business, and medicine. The datasets in these fields are large, complex, and often noisy. Extracting knowledge requires the use of sophisticated, high-performance and principled analysis techniques and algorithms, based on sound theoretical and statistical foundations. These techniques in turn require powerful visualization technologies; implementations that must be carefully tuned for performance; software systems that are usable by scientists, engineers, and physicians as well as researchers; and infrastructures that support them.

This conference provides a venue for researchers who are addressing these problems to present their work in a peer-reviewed forum. It also provides an ideal setting for graduate students and others new to the field to learn about cutting-edge research by hearing outstanding invited speakers and attending tutorials (included with conference registration). A set of focused workshops are also held on the last day of the conference. The proceedings of the conference are published in archival form, and are also made available on the SIAM web site.

AAAI 2010
The Twenty-Fourth AAAI Conference on Artificial Intelligence
Atlanta, Georgia, USA
July 11-15, 2010
http://www.aaai.org/Conferences/AAAI/aaai10.php/

AAAI 2010 is the Twenty-Fourth AAAI Conference on Artificial Intelligence (AI). The purpose of this conference is to promote research in AI and scientific exchange among AI researchers, practitioners, scientists, and engineers in related disciplines. AAAI 2010 will have multiple technical tracks, student abstracts, poster sessions, invited speakers, and exhibit programs, all selected according to the highest reviewing standards.

AAAI 2010 welcomes submissions on mainstream AI topics as well as novel cross-cutting work in related areas. Topics include but are not limited to the following: Agents; Cognitive modeling and human interaction; Commonsense reasoning; Constraint satisfaction and optimization; Evolutionary computation; Game playing and interactive entertainment; Information integration and extraction; Knowledge acquisition and ontologies; Knowledge representation and reasoning; Machine learning and data mining; Model-based systems; Multiagent systems; Natural language processing; Planning and scheduling; Probabilistic reasoning; Robotics; Search Papers that extend the state of the art, and explore parts of the design space of AI that are not well explored are particularly encouraged. A full list of keywords is available.

IEEE Computer Society
1730 Massachusetts Ave, NW
Washington, D.C. 20036-1903

Non-profit Org.
U.S. Postage
PAID
Silver Spring, MD
Permit 1398