Chapter 3
Agent-Based Distributed Data Mining: A Survey

Chayapol Moemeng, Vladimir Gorodetsky, Ziye Zuo, Yong Yang and Chengqi Zhang

Abstract Distributed data mining is originated from the need of mining over decentralised data sources. Data mining techniques involving in such complex environment must encounter great dynamics due to changes in the system can affect the overall performance of the system. Agent computing whose aim is to deal with complex systems has revealed opportunities to improve distributed data mining systems in a number of ways. This paper surveys the integration of multi-agent system and distributed data mining, also known as agent-based distributed data mining, in terms of significance, system overview, existing systems, and research trends.

3.1 Introduction

Originated from knowledge discovery from databases (KDD), also known as data mining (DM), distributed data mining (DDM) mines data sources regardless of their physical locations. The need for such characteristic arises from the fact that data produced locally at each site may not often be transferred across the network due to the excessive amount of data and privacy issues. Recently, DDM has become a critical components of knowledge-based systems because its decentralised architecture reaches every networked business.

This chapter discusses a symbiont synthesising the two widely accepted research fields: data mining and agents. Readers are recommended to review surveys regarding distributed data mining in [23], [34], and [46].

3.2 Why Agents

DDM is a complex system focusing on the distribution of resources over the network as well as data mining processes. The very core of DDM systems is the scalability

L. Cao (ed.), Data Mining and Multi-agent Integration, DOI: 10.1007/978-1-4419-0522-2_3,

as the system configuration may be altered time to time, therefore designing DDM systems deals with great details of software engineer issues, such re-usability, extensibility, and robustness. For these reasons, agents' characteristics are desirable for DDM systems.

Furthermore, the decentralisation property seems to fit best with the DDM requirement. At each data site, mining strategy is deployed specifically for the certain domain of data. However, there can be other existing or new strategies that data miner would like to test. A data site should seamlessly integrate with external methods and perform testing on multiple strategies for further analysis. Autonomous agent can be treated as a computing unit that performs multiple tasks based on a dynamic configuration. The agent interprets the configuration and generates an execution plan to complete multiple tasks.

[2], [21], [14], [20], [27], and [26] discuss the benefits of deploying agents in DDM systems. Nature of MAS is decentralisation and therefore each agent has only limited view to the system. The limitation somehow allows better security as agents do not need to observe other irrelevant surroundings. Agents, in this way, can be programmed as compact as possible, in which light-weight agents can be transmitted across the network rather than the data which can be more bulky. Being able to transmit agents from one to another host allows dynamic organisation of the system. For example, mining agent a_1, located at site s_1, posses algorithm alg_1. Data mining task t_1 at site s_2 is instructed to mine the data using alg_1. In this setting, transmitting a_1 to s_2 is a probable way rather than transfer all data from s_2 to s_1 where alg_1 is available.

In addition, security, a.k.a. trust-based agents[41][19], is a critical issue in ADDM. Rigid security models intending to ensure the security may degrade the system scalability. Agents offer alternative solutions as they can travel through the system network. As in [29], the authors present a framework in which mobile agents travel in the system network allowing the system to maintain data privacy. Thanks to the self-organisation characteristic which excuses the system from transferring data across the network therefore adds up security of the data.

A trade-off for the previous discussed issue, scalability is also a critical issue of a distributed system. In order to inform every unit in the system about the configuration update of the system, such as a new data site has joined the system, demands extra human interventions or high complex mechanism in which drops in performance may occur. To this concern, collaborative learning agents[40][6] are capable of sharing information, in this case, about changes of system configuration, and propagate from one agent to another allowing adaptation of the system to occur at individual agent level. Furthermore, mobile agents as discussed earlier can help reduce network and DM application server load as in state-of-art systems[3][24].

3.3 Agent-Based Distributed Data Mining

Applications of distributed data mining include credit card fraud detection system, intrusion detection system,health insurance, security-related applications, distributed clustering, market segmentation,sensor networks, customer profiling, evaluation of retail promotions, credit risk analysis, etc. These DDM application can be further enhanced with agents. ADDM takes data mining as a basis foundation and is enhanced with agents; therefore, this novel data mining technique inherits all powerful properties of agents and, as a result, yields desirable characteristics.

In general, constructing an ADDM system concerns three key characteristics: interoperability, dynamic system configuration, and performance aspects, discussed as follows.

Interoperability concerns, not only collaboration of agents in the system, but also external interaction which allow new agents to enter the system seamlessly. The architecture of the system must be open and flexible so that it can support the interaction including communication protocol, integration policy, and service directory. Communication protocol covers message encoding, encryption, and transportation between agents, nevertheless, these are standardised by the Foundation of Intelligent Physical Agents (FIPA)[1] and are available for public access. Most agent platforms, such as JADE[2] and JACK[3], are FIPA compliant therefore interoperability among them are possible. Integration policy specifies how a system behaves when an external component, such as an agent or a data site, requests to enter or leave. The issue is further discussed in [47] and [29]

In relation with the interoperability characteristic, dynamic system configuration, that tends to handle a dynamic configuration of the system, is a challenge issue due to the complexity of the planning and mining algorithms. A mining task may involve several agents and data sources, in which agents are configured to equip with an algorithm and deal with given data sets. Change in data affects the mining task as an agent may be still executing the algorithm.

Lastly, performance can be either improved or impaired because the distribution of data is a major constraint. In distributed environment, tasks can be executed in parallel, in exchange, concurrency issues arise. Quality of service control in performance of data mining and system perspectives is desired, however it can be derived from both data mining and agents fields.

Next, we are now looking at the overview of our point of focus. An ADDM system can be generalised into a set of components and viewed as depicted in figure 3.1. We may generalise activities of the system into request and response, each of which involves a different set of components. Basic components of an ADDM system are as follows.

---

[1] FIPA, http://www.fipa.org/

[2] Java Agent DEvelopment Framework, http://jade.tilab.com/

[3] JACK, http://www.agent-software.com.au/products/jack/index.html

Data: Data is the foundation layer of our interest. In distributed environment, data can be hosted in various forms, such as online relational databases, data stream, web pages, etc., in which purpose of the data is varied.

Communication: The system chooses the related resources from the directory service, which maintains a list of data sources, mining algorithms, data schemas, data types, etc. The communication protocols may vary depending on implementation of the system, such as client-server, peer-to-peer, etc.

Presentation: The user interface (UI) interacts with the user as to receive and respond to the user. The interface simplifies complex distributed systems into user-friendly message such as network diagrams, visual reporting tools, etc.

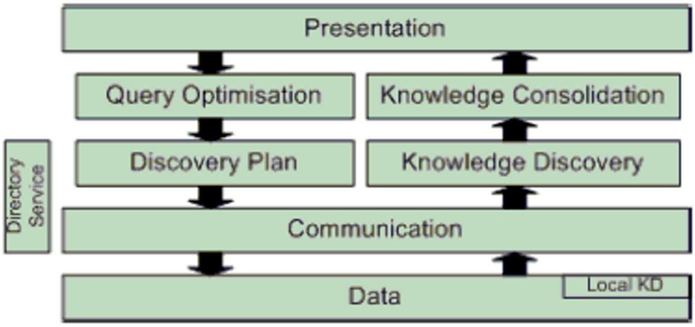

Fig. 3.1: Overview of ADDM systems

On the other hand, when a user requests for data mining through the UI, the following components are involved.

Query optimisation: A query optimiser analyses the request as to determine type of mining tasks and chooses proper resources for the request. It also determines whether it is possible to parallelise the tasks, since the data is distributed and can be mined in parallel.

Discovery Plan: A planner allocates sub-tasks with related resources. At this stage, mediating agents play important roles as to coordinate multiple computing units since mining sub-tasks performed asynchronously as well as results from those tasks.

On the other hand, when a mining task is done, the following components are taken place,

Local Knowledge Discovery (KD): In order to transform data into patterns which adequately represent the data and reasonable to be transferred over the network, at each data site, mining process may take place locally depending on the individual implementation.

Knowledge Discovery: Also known as mining, it execute the algorithm as required by the task to obtain knowledge from the specified data source.

Knowledge Consolidation: In order to present to the user with a compact and meaningful mining result, it is necessary to normalise the knowledge obtained from various sources. The component involves a complex methodologies to combine knowledge/patterns from distributed sites. Consolidating homogeneous knowledge/patterns is promising and yet difficult for heterogeneous case.

3.4 Interaction and Integration

Let us briefly outline the basic works on ADDM. A survey on agent technology for data mining can be found in [33] . [12] is one of the first works attracting attention to ADDM arguing its advantages in mining vast amount of data stored in network and using collaborative capabilities of DM agents. It studies agent-based approach to distributed knowledge discovery using Inductive Logic Programming (ILP) approach and provides for some experiment results of application of the agent-based approach to DDM.

The paper [22] proposes PADMA (Parallel Data Mining Agents) system addressing use of agent architecture to cope large scale and distributed nature of data sources as applied to hierarchical clustering. It is intended to handle numeric and textual data with the focus on the latter. Agency of DDM system consists of DM agents that are responsible for local data access and extraction of high level useful information, agent-facilitator coordinating the DM agents operation while handling their SQL queries and presenting them to user interface. Agent-facilitator gets “conceptual graphs”from DM agents, combines them and passes the results to user. The focus of the paper is to show the benefit of agent-based parallel data mining.

JAM (Java Agents for Meta-learning) system proposed in [37] is an agent based system supporting the launching of learning, classifier and meta-learning agents over distributed database sites. It uses parallelism and distributed nature of meta-learning and its possibility to share meta-information without direct access to distributed data sets. JAM [36] in constituted of distributed learning and classification programs operating in parallel on JAM sites that are linked to a network. In turn, JAM site contains one or more local data base; one or more learning agents (machine learning programs); one or more meta-learning agents, intended for combing decisions produced by local classifier agents; a repository of decisions computed locally and imported by local and meta classifier agents; a local user configuration file and graphical user interface. Once the local and meta-classifiers are generated the user manages the execution of the above modules to classify new (unlabelled) data. A peculiarity of JAM system operation is that each local site may import decisions of remote classifiers from peer JAM sites and combine these decisions with own local classifier decisions using local meta-learning agent. JAM sites are operating simultaneously and independently. Administration of JAM site local activity is performed by the user via local user configuration file. Details of the JAM system architecture and implementation can be found in [36].

The paper [42] is motivated by the desire to attack increasingly difficult problems and application domains which often require to process very large amounts of data or data collected at different geographical locations that cannot be processed by sequential and centralised systems. It is also motivated by capabilities of MAS to provide for computing processes with robustness, fault tolerance, scalability, and speed-up as well as by maturity of the computer and network technology supporting implementation of parallel and distributed information processing. Based on multi-agent learning, target system can be built as self improving their performance. In particular, the paper considers job assignment problem, the core of many scheduling tasks, aiming to find reasonable solution in a reasonable time. The nodes of partially ordered set of jobs are considered as active entities or agents and the jobs are considered as passive entities or resources to be used by the agents. The agents interact in order to find a solution meeting some predefined criteria. The solution search procedure implemented by agents is based on distributed reinforcement learning starting from an initial solution that is performed using low-level communication and coordination among agents. The paper experimentally proves some advantages of the developed multi-agent model for implementation of parallel and distributed machine learning in the problem in question.

[3] describes the developed Papyrus system that is Java-based and intended for DDM handling with clusters and meta-clusters distributed over heterogeneous data sites. Mobile DM agents of the system are capable to move data, intermediate results and models between clusters for local processing thus reducing network load. Papyrus supports several techniques for exchanging and combining locally mined decisions (predictive models) and meta-data that are necessary to describe the above models specified in terms of a special mark-up language.

[25] studies advantages and added value of using ADDM, reviews and classifies existing agent-based approaches to DDM and proposes agent-oriented implementation of a distributed clustering system. The paper explicitly formulates why agent-based approach is very perspective one for DDM (autonomy, interactivity, capability to dynamically select data sources in changeable environment, etc.). The proposed KDEC scheme addressing computing statistical density estimation and information theoretic sampling to minimise communication between sites is implemented on the basis of agent technology as distributed data clustering system. In addition, one of its distinctive features is that it preserves the local data privacy.

[10] emphasises the possible synergy between MAS and DDM technologies. It particularly focuses on distributed clustering, having every increasing application domains, e.g. in sensor networks deployed in hostile and difficult to access locations like battle fields where sensors are measuring vibration, reflectance, temperature, and audio signals; in sensor networks for monitoring a terrain, smart home, and many other domains. In these domains analysing, e.g. DDM task, are non-trivial problems due to many constraints such as limited bandwidth of wireless communication channels, peer-to-peer mode of communication and the necessity of interaction in asynchronous network, data privacy, coordination of distributed computing, non-trivial decomposability, formidable number of data network nodes, limited computing resources, e.g. due to limited power supply, etc. The authors state that the traditional framework for centralised data analysis and DM algorithms does not really scale very well in such distributed applications. In contrast, this distributed problem solving can be very well coped with the multi-agent framework supporting

semi-autonomous behaviour, collaboration and reasoning, among other perspective MAS properties. From DDM domain perspective, the paper focuses on clustering in sensor networks that offers many aforementioned challenges. This paper suggests that traditional centralised data mining techniques may not work well in the above domains and underscores, among others, and that DDM algorithms integrated with MAS architecture may offer novel very perspective synergetic information technology for these domains.

Recently some efforts were paid to development of technological issues of multi-agent data mining that can be evaluated as a sign of increasing maturity. [20] states that the core problem of distributed data mining and machine learning design does not concern particular data mining techniques. Instead of this, its core problem is development of an infrastructure and protocols supporting coherent collaborative operations of distributed software components (agents) performing distributed learning. The paper proposes a multi-agent architecture of an information fusion system possessing of DDM and machine learning capabilities. It also proposes a design technology, which core is constituted by a number of specialised agent interaction protocols supporting distributed agent operations in various use cases (scenarios) including, in particular, DDM protocol. Further development of DDM system design technology is given in [16] and [17].

[39] proposes a framework (an abstract architecture) for agent-based distributed machine learning and data mining. The proposed framework , as it is motivated by the authors, is based of the observation that "despite the autonomy and self-directedness of learning agents, many of such systems exhibit a sufficient overlap in terms of individual learning goals so that beneficial operation might be possible if a model for flexible interaction between autonomous learners was available that allowed agents to (i) exchange information about different aspects of their own learning mechanism at different levels of detail without being forced to reveal private information that should not be disclosed, (ii) decide to what extent they want to share information about their own learning processes and utilise information provided by other learners, and (iii) reason about how this information can best be used to improve their own learning performance."

The idea underlying the proposed framework is that each agent is capable to maintain meta-description of own learning processes in a form that makes it admissible, due to privacy issue, to exchange meta-information with other agents and reason about it rationally, i.e. to reason in a way providing for improving of their own learning results. The authors state that this possibility, for learning agents, is a hypothesis they intend to justify experimentally within a proposed formal framework. Actually this paper presents a preliminary research results and there are a lot of efforts to be done to reach reliable evaluation of the hypothesis used.

The last system to review in this survey is F-TRADE (Financial Trading Rules Automated Development and Evaluation) [9][32]. The system uses a dynamic approach that allows integration of external data sources and mining algorithm. The system presents, to financial traders and brokers, a testbed on actual data sets, which can help them evaluate their favourite trading strategies iteratively with confidence before investing money into the real markets. The motivation of F-TRADE is to provide financial traders and researchers, and financial data miners with a practically flexible and automatic infrastructure.

With this infrastructure, they can plug their algorithms onto the system and concentrate on improving the performance of their algorithms with iterative evaluation on a large amount of real stock data from international markets. The system services include (i) trading services support, (ii) mining services support, (iii) data services support, (iv) algorithm services support, and (v) system services support. In order to support all these services, soft plug-and-play is essential in the F-TRADE. Each system components interact through XML schema which specifies details of the components and allows agents to examine and use.

Besides those role-model systems presented earlier, there are more ADDM related works which the reader is encouraged to review: [4], [6], [7], [28], [39], [30], [31], [35], and [45].

With regards to the symbiosis, interaction and integration of the two researches have attracted the communities' attention. Many research bodies have listed agent and data mining interaction and integration (AMII) as a special interest. [8] discuss challenges of interaction and integration of agents with data mining. There is a community website dedicated to this special interest research group at *AgentMining* (http://www.agentmining.org). which provides related resources, such as list of research topics, activities, workshops and conferences, links, related publications, research groups, etc.

3.5 Open Issues and Trends

The interaction and integration between the two technologies have explore the new challenges. Considering various ingredients for the integration could be a key to rapidly enhance the development process and usability of the system, let us examine them from different perspectives.

Research Perspective: Data distribution in real-life applications are either homogeneous or heterogeneous. Data can be partitioned both vertically and horizontally, and furthermore data splitting may not be available across the sites. For examples, two related customer databases may not reflect each others in which a customer may never provide contact details but somehow appear to buy some products. The applications will require a data mining technology to pay careful attention to the distributed computing, communication, and storage of the system.

Another approach to develop ADDM is an inspiration from the nature which has proven to be promising. Swarm intelligence is closely related to intelligent agents. Recently, researchers pay attention to the possibility to implement DDM systems with swarm intelligence. Sample applications of swarm intelligence in data mining are rule-based classifiers using ants, feature selection with ant colony optimisation, data and text mining with hierarchical clustering ants, etc. Further readings can be found in [1] and [38].

Software Engineering Perspective: Expectedly, ADDM frequently requires exchange of data mining models among the data sites. Therefore, seamless and transparent realisation of DDM technology will require standardised schemes to represent and exchange models. Therefore, software engineering tools that support the design of data mining and distributed database are desired. So far, PMML, the Cross-Industry Standard Process Model for Data mining (CRISP-DM), and other related efforts are likely to be very useful.

The very basic foundation of our focus is the database. Not only full-scale database, like relational database, is taken into consideration during system integration. Desktop and light-weight database running on limited devices, such as mobile phones, can be integrated into ADDM. Mobile agents can be migrated (downloaded) and perform task on the devices and take back only a representative model for further analysis.

The second ingredient is the emergence of service oriented architecture (SOA) that enables agent-based application to integrate better than ever. SOA is a promising architecture as it is widely adapted in several applications. We cannot deny the fact that web-based applications are becoming more and more popular. Internet has become a necessary element of a computer system.

System Perspective: A novel very perspective but poorly researched application area of agents and data mining synergy is mobile, ubiquitous and peer-to-peer (P2P) computing. A specific feature of such computing systems is that the latter operate with dynamic set of information sources. E.g., the mobile devices may move and freely enter to and exit from the network thus changing the set of network nodes and communication topology, changing the set of available services as well. Examples of such application areas are, e.g., smart space and ambient intelligence. In these environments, decisions are made on the basis of fusion of information received from distributed sensors and mobile devices populating the environment. One of the objectives of such application is adaptation to multiple human habits that can be achieved through learning of multiple human profiles. On the other hand, for class of applications in question, multi-agent approach supplies for most natural framework, appropriate architecture, as well as design technology. Thus, integrating agent and data mining in ubiquitous environments like smart space, ambient intelligence, etc., could be very perspective and promising to reach high quality performance of corresponding applied systems.

In fact, ubiquitous and mobile computing form a novel and very perspective, although poorly researched, application area of agents and data mining synergy. A specific feature of such computing systems is that the latter often has to handle with dynamic set of information sources. E.g., the mobile devices may move and freely enter to and exit from the network thus changing the set of network nodes and communication topology, changing the set of available services as well. Examples of such application areas are, e.g., smart space and ambient intelligence. In these environments, decisions are made on the basis of fusion of information received from distributed sensors and mobile devices populating the environment. One of the objectives of such application is adaptation to multiple human habits that can be achieved through learning of multiple human profiles. On the other hand, for class of applications in question, multi-agent approach supplies for most natural framework, appropriate architecture, as well as sound design technology. Thus, integrating agent and data mining in ubiquitous environments like smart space, ambient intelligence, etc., could be very perspective and promising to reach high quality performance of corresponding applied systems. [15] presents a summary of challenges integrating ubicomp with MAS for data mining task.

Recently, peer-to-peer (P2P) computing has proven its excellence through its product, such as peer download software, file sharing software, which they gather users to join the service quickly. P2P is respected as one of the best scalable system, and thus it increases availability of the system as millions of peers can be attached to the network. P2P algorithm does not rely on a central server, each unit performs its own task and requests for data from others if available in order to save the redundant time. However, security is a critical issue in P2P due to exchanging information with other peers that can add a vulnerability to the network, such as denial of service or selfish behaviour. Some peers may only consume others' resources while they do not provide to others. Nevertheless, each peer must agree of terms and conditions of use before joining the network. P2P has caught researchers attention due to the compliance with multi-agent systems as appear in [18] and [11].

User Perspective: Finally human-computer interaction issues in DDM offers some unique challenges. It requires system-level support for group interaction, collaborative problem solving, development of alternate interfaces (particularly for mobile devices), and dealing with security issues.

References

1. Ajith Abraham, Crina Grosan, and Vitorino Ramos, editors. *Swarm Intelligence in Data Mining*, volume 34 of *Studies in Computational Intelligence*. Springer, 2006.
2. Sung W. Baik, Jerzy W. Bala, and Ju S. Cho. Agent based distributed data mining. *Lecture Notes in Computer Science*, 3320:42–45, 2004.
3. S. Bailey, R. Grossman, H. Sivakumar, and A. Turinsky. Papyrus: a system for data mining over local and wide area clusters and super-clusters. In *Supercomputing '99: Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM)*, page 63, New York, NY, USA, 1999. ACM.
4. R. J. Bayardo, W. Bohrer, R. Brice, A. Cichocki, J. Fowler, A. Helal, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, and Others. InfoSleuth: agent-based semantic integration of information in open and dynamic environments. *ACM SIGMOD Record*, 26(2):195–206, 1997.
5. F. Bergenti, M. P. Gleizes, and F. Zambonelli. *Methodologies And Software Engineering For Agent Systems: The Agent-oriented Software Engineering Handbook*. Kluwer Academic Publishers, 2004.
6. A. Bordetsky. Agent-based Support for Collaborative Data Mining in Systems Management. In *Proceedings Of The Annual Hawaii International Conference On System Sciences*, page 68, 2001.
7. R. Bose and V. Sugumaran. IDM: an intelligent software agent based data mining environment. *1998 IEEE International Conference on Systems, Man, and Cybernetics*, 3, 1998.
8. L. Cao, C. Luo, and C. Zhang. Agent-Mining Interaction: An Emerging Area. *Lecture Notes in Computer Science*, 4476:60, 2007.

9. L. Cao, J. Ni, J. Wang, and C. Zhang. Agent Services-Driven Plug and Play in the F-TRADE. In *17th Australian Joint Conference on Artificial Intelligence*, volume 3339, pages 917–922. Springer, 2004.
10. J. Dasilva, C. Giannella, R. Bhargava, H. Kargupta, and M. Klusch. Distributed data mining and agents. *Engineering Applications of Artificial Intelligence*, 18(7):791–807, October 2005.
11. S. Datta, K. Bhaduri, C. Giannella, R. Wolff, and H. Kargupta. Distributed data mining in peer-to-peer networks. *Internet Computing, IEEE*, 10(4):18–26, 2006.
12. W. Davies and P. Edwards. Distributed Learning: An Agent-Based Approach to Data-Mining. In *Proceedings of Machine Learning 95 Workshop on Agents that Learn from Other Agents*, 1995.
13. U. Fayyad, R. Uthurusamy, and Others. Data mining and knowledge discovery in databases. *Communications of the ACM*, 39(11):24–26, 1996.
14. C. Giannella, R. Bhargava, and H. Kargupta. Multi-agent Systems and Distributed Data Mining. *Lecture Notes in Computer Science*, pages 1–15, 2004.
15. V. Gorodetskiy. Interaction of agents and data mining in ubiquitous environment. In *Proceedings of the 2008 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'08)*, 2008.
16. V. Gorodetsky, O. Karsaev, and V. Samoilov. Multi-Agent Data and Information Fusion. *Nato Science Series Sub Series Iii Computer And Systems Sciences*, 198:308, 2005.
17. V. Gorodetsky, O. Karsaev, and V. Samoilov. Infrastructural Issues for Agent-Based Distributed Learning. In *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pages 3–6. IEEE Computer Society Washington, DC, USA, 2006.
18. V. Gorodetsky, O. Karsaev, V. Samoylov, and S. Serebryakov. P2P Agent Platform: Implementation and Testing. In *Proceedings International Workshop "Agent and Peer-to-Peer Computing"(AP2PC-2007) associated with AAMAS-07. Honolulu, Hawaii*, pages 21–32, 2007.
19. V. Gorodetsky and I. Kotenko. The Multi-agent Systems for Computer Network Security Assurance: frameworks and case studies. In *IEEE International Conference on Artificial Intelligence Systems, 2002*, pages 297–302, 2002.
20. Vladimir Gorodetsky, Oleg Karsaev, and Vladimir Samoilov. Multi-agent technology for distributed data mining and classification. In *IAT*, pages 438–441. IEEE Computer Society, 2003.
21. Sven A. Brueckner H. Van Dyke Parunak. Engineering swarming systems. *Methodologies and Software Engineering for Agent Systems*, pages 341–376, 2004.
22. H. Kargupta, I. Hamzaoglu, and B. Stafford. Scalable, distributed data mining using an agent based architecture. In *Proceedings the Third International Conference on the Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, California*, pages 211–214, 1997.
23. H. Kargupta and K. Sivakumar. Existential pleasures of distributed data mining. *Data Mining: Next Generation Challenges and Future Directions*, pages 1–25, 2004.
24. Hillol Kargupta, Byung-Hoon Park, Daryl Hershberger, and Erik Johnson. Collective data mining: A new perspective toward distributed data mining. In *Advances in Distributed and Parallel Knowledge Discovery*, pages 133–184, 1999.
25. M. Klusch, S. Lodi, and G. Moro. Agent-based distributed data mining: The KDEC scheme. *AgentLink, Springer Lecture Notes in Computer Science*, 2586, 2003.
26. M. Klusch, S. Lodi, and G. Moro. The role of agents in distributed data mining: Issues and benefits. In *Proceedings of the 2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT 2003)*, pages 211–217, 2003.
27. Matthias Klusch, Stefano Lodi, and Gianluca Moro. Issues of agent-based distributed data mining. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, pages 1034–1035. ACM, 2003.
28. Raymond S. T. Lee and James N. K. Liu. ijade web-miner: An intelligent agent framework for internet shopping. *IEEE Transactions on Knowledge and Data Engineering*, 16(4):461–473, 2004.
29. Xining Li and Jingbo Ni. Deploying mobile agents in distributed data mining. *Lecture Notes in Computer Science*, 4819:322–331, 2007.

30. F. Menczer, R. K. Belew, and W. Willuhn. Artificial life applied to adaptive information agents. In *AAAI Spring Symposium on Information Gathering*, pages 128–132, 1995.
31. S. Merugu and J. Ghosh. A distributed learning framework for heterogeneous data sources. In *Conference on Knowledge Discovery in Data*, pages 208–217. ACM Press New York, NY, USA, 2005.
32. Peerapol Moemeng, Longbing Cao, and Chengqi Zhang. F-TRADE 3.0: An Agent-Based Integrated Framework for Data Mining Experiments. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 3, pages 612–615, Los Alamitos, CA, USA, 2008. IEEE Computer Society.
33. L. Panait and S. Luke. Cooperative Multi-Agent Learning: The State of the Art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, 2005.
34. Byung H. Park and Hillol Kargupta. Distributed data mining: Algorithms, systems, and applications. In Nong Ye, editor, *Data Mining Handbook*, pages 341–358. Lawrence Earlbaum Associates, 2002.
35. M. Penang. Distributed Data Mining From Heterogeneous Healthcare Data Repositories: Towards an Intelligent Agent-Based Framework. In *Proceedings of the 15th IEEE Symposium on Computer-Based Medical Systems:(CBMS 2002): 4-7 June, 2002, Maribor, Slovenia*. IEEE Computer Society, 2002.
36. Andreas Leonidas Prodromidis. *Management of intelligent learning agents in distributed data mining systems*. PhD thesis, Columbia University, New York, NY, USA, 1999.
37. S. Stolfo, A. L. Prodromidis, S. Tselepis, W. Lee, D. W. Fan, and P. K. Chan. JAM: Java agents for meta-learning over distributed databases. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, pages 74–81, 1997.
38. H. Van Dyke Parunak Sven A. Brueckner. Swarming distributed pattern detection and classification. In *Environments for Multi-Agent Systems*, pages 232–245. Springer, 2005.
39. J. Tozicka, M. Rovatsos, and M. Pechoucek. A framework for agent-based distributed machine learning and data mining. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. ACM New York, NY, USA, 2007.
40. Jan Tozicka, Michal Jakob, and Michal Pechoucek. Market-inspired approach to collaborative learning. *Lecture Notes in Computer Science*, 4149:213–227, 2006.
41. Frank E. Walter, Stefano Battiston, and Frank Schweitzer. A model of a trust-based recommendation system on a social network, Nov 2006.
42. Gerhard Weiss. A multiagent perspective of parallel and distributed machine learning. In *Agents*, pages 226–230, 1998.
43. M. Wooldridge and N. R. Jennings. Agent Theories, Architectures, and Languages: A Survey. *Intelligent Agents*, 22, 1995.
44. M. Wooldridge, N. R. Jennings, and D. Kinny. The Gaia Methodology for Agent-Oriented Analysis and Design. *Autonomous Agents and Multi-Agent Systems*, 3(3):285–312, 2000.
45. S. Z. H. Zaidi, S. S. R. Abidi, S. Manikam, and Cheah Yu-N. Admi: a multi-agent architecture to autonomously generate data mining services. In *Intelligent Systems, 2004. Proceedings. 2004 2nd International IEEE Conference*, volume 1, pages 273–279 Vol.1, 2004.
46. Mohammed J. Zaki. Parallel and distributed association mining: A survey. *IEEE Concurrency*, 7(4):14–25, 1999.
47. Gorodetskiy, V.; Karsaev, O.; Samoilov, V.; Serebryakov, S., Agent-based Service-Oriented Intelligent P2P Networks for Distributed Classification *Hybrid Information Technology*, 2006.