Future Generation Computer Systems 54 (2016) 68–78

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Distributed volunteer computing for solving ensemble learning problems

Eugenio Cesario *, Carlo Mastroianni, Domenico Talia

ICAR-CNR, via P. Bucci 41C, 87036 Rende (CS), Italy

HIGHLIGHTS

- We present Mining@Home, a framework for distributed data mining.
- Mining@Home combines the benefits of P2P protocols with those of the volunteer computer paradigm.
- The framework is used to discover classifiers by applying the "bagging" technique on real data.
- We present performance results showing the efficiency and scalability of our approach.
- Performance results are obtained through real experiments, simulation and analytical assessment.

ARTICLE INFO

Article history:
Received 30 July 2014
Received in revised form
15 May 2015
Accepted 22 July 2015
Available online 3 August 2015

Keywords:
Ensemble learning
Distributed data mining
Peer-to-peer
Volunteer computing

ABSTRACT

The volunteer computing paradigm, along with the tailored use of peer-to-peer communication, has recently proven capable of solving a wide area of data-intensive problems in a distributed scenario. The Mining@Home framework is based on these paradigms and it has been implemented to run a wide range of distributed data mining applications. The efficiency and scalability of the architecture can be fully exploited when the overall task can be partitioned into distinct jobs that may be executed in parallel, and input data can be reused, which naturally leads to the use of data cachers. This paper explores the opportunities offered by Mining@Home for coping with the discovery of classifiers through the use of the bagging approach: multiple learners are used to compute models from the same input data, so as to extract a final model with high statistical accuracy. Analysis focuses on the evaluation of experiments performed in a real distributed environment, enriched with simulation assessment – to evaluate very large environments – and with an analytical investigation based on the iso-efficiency methodology. An extensive set of experiments allowed to analyze a number of heterogeneous scenarios, with different problem sizes, which helps to improve the performance by appropriately tuning the number of workers and the number of interconnected domains.

1. Introduction

The global information society is a restless producer and exchanger of huge volumes of data and an increasing effort is needed for the extraction of valuable information, useful for business and scientific applications, from data. Fortunately, the notable advancements and the advent of new paradigms for distributed computing, such as Grids, P2P systems, and Cloud Computing, help us in many scenarios to cope with this data deluge. The efficiency of distributed approaches to data analysis has improved for several reasons: (i) the wide availability of Cloud infrastructures, which allow even a small company to offload applications to remote data centers or to integrate on-premises hosts with elastic resources utilized in a pay-per-use fashion; (ii) data links have larger bandwidths than before, enabling the assignment of tasks and the transmission of related input data in a distributed scenario; (iii) data caching techniques can help to reuse data needed by different tasks, (iv) Internet computing models such as the "public resource computing" or "volunteer computing" paradigm facilitate the use of spare CPU cycles of a large number of computers.

The algorithms, methodologies and architectural efforts aiming to extract knowledge in a distributed scenario are collectively known as "distributed data mining" [1,2]. Knowledge discovery is speeded up by concurrently executing a number of data mining tasks on different data subsets: specific attention must be given

* Corresponding author.
E-mail addresses: cesario@icar.cnr.it (E. Cesario), mastroianni@icar.cnr.it (C. Mastroianni), talia@dimes.unical.it (D. Talia).

http://dx.doi.org/10.1016/j.future.2015.07.010

to the efficient combination of distributed analysis of data and centralized collection of results.

This paper deeply investigates the performance of Mining@Home, a framework designed to solve distributed data mining problems through the distribution of data and parallelization of mining tasks. The approach used in the framework combines solutions developed in two different fields, volunteer computing and peer-to-peer data mining. Volunteer computing [3] has become a success story for many scientific applications, as a means for exploiting huge amount of low cost computational resources with a few manpower getting involved. So far this field has experienced little integration with the area of distributed and peer-to-peer data mining. The main reason for this is the centralized nature of popular volunteer computing platforms available today, such as *BOINC* [4] and *XtremWeb* [5,6], which requires all data to be served by a group of centrally maintained servers. However, the centralized approach can generate bottlenecks and single points of failure in the system. Moreover, a centralized solution is not naturally suited for applications in which input data files are initially stored in distributed locations.

These considerations inspired the design of the Mining@Home architecture, which exploits the methodologies of volunteer computing and tailors them to properly match the characteristics and benefits of peer-to-peer protocols and algorithms. The Mining@Home architecture is data-oriented, and it exploits distributed cache servers for the efficient dissemination and reutilization of data files. This kind of solution can improve the performance of public computing systems, in terms of efficiency, flexibility and robustness, and it can also enlarge the use of the public computing paradigm, since any user is allowed to define its own data mining application and specify the jobs that will be executed by remote volunteers. The approach differs from the centralized architectures, such as that one used in BOINC, in that Mining@Home integrates P2P networking directly into the system, as job descriptions and input data are provided to a P2P network instead of being directly delivered to the hosts that execute the tasks.

In [7], Mining@Home was assessed through an early simulator, and it proved able to extract *closed frequent itemsets* from a transactional database. After those early simulations, we worked to provide a full implementation in Java of the framework, and now it can efficiently support the execution of different data mining applications in distributed scenarios. To the best of our knowledge, Mining@Home is the first implemented volunteer computing framework that run data mining tasks in a distributed environment. In [8] the basic implementation of the framework was sketched together with a preliminary evaluation on ensemble learning tasks. The ensemble learning approach combines multiple mining models together instead of using a single model in isolation [9]. In particular, the "bagging" strategy consists of sampling an input dataset multiple times, to introduce variability between the different models, and then extracting the combined model with a voting technique or a statistical analysis.

This paper extends the work presented in [8] in many ways: (i) the experimental testbed was extended from two to four computing domains, and the size of input data was extended from 2 million instances to 5 million instances of a reference dataset; (ii) experimental evaluations were complemented with a simulation tool that incorporates data coming from real experiments (for example, job durations) enabling the assessment of wider scenarios, with up to 128 workers distributed among 32 domains; (iii) mathematical analysis, based on iso-efficiency methodologies, was used to investigate how the performances, in particular in terms of execution time and speedup, are related to the number of workers, the number of domains and the dataset size. The results confirm the feasibility of the approach, the scalability and efficiency of the framework, and also show that it may be possible to optimize the performance by choosing the appropriate system and network configuration, for example, by tuning the number of workers and the number of domains on which the workers are distributed.

The reminder of the paper is organized as follows: Section 2 presents the architecture of Mining@Home and the peer-to-peer protocol used for the assignment of tasks to workers. Section 3 discusses the ensemble learning strategy. Section 4 illustrates the scenario of the experiments, presents the results obtained in a real testbed and via simulation, and uses the iso-efficiency model to separately evaluate the contributions of useful and overhead computation. Finally, Section 5 discusses related work, specifically in the fields of distributed data mining and public resource computing, and Section 6 concludes the paper.

2. Architecture and implementation of Mining@home

As already mentioned, a simulator of the Mining@Home framework was introduced in [7] for solving the problem of finding closed frequent itemsets in a transactional database, and simulation results were reported. After that, the Mining@Home system was fully implemented and used for coping with a number of different data analysis scenarios involving the execution of different data mining tasks in a distributed environment. The architecture of the Mining@Home framework distinguishes between nodes accomplishing the mining task and nodes supporting data dissemination. In the first group:

- the **data source** is the node that stores the dataset to be read and mined.
- the **job manager** is the node in charge of decomposing the overall data mining application in a set of independent tasks. This node produces a *job advert* document for each task, which describes its characteristics and specifies the portion of the data needed to complete the task. The job manager is also responsible for the collection of results.
- the **miners** are the nodes available for job execution. Assignment of jobs follows the "pull" approach, as required by the volunteer computing paradigm.

Data exchange and dissemination is done by exploiting the presence of a network of super-peers for the assignment and execution of jobs, and adopting caching strategies to improve the efficiency of data delivery. Specifically:

- **super peer** nodes constitute the backbone of the network. Miners connect directly to a super-peer, and super-peers are connected with one another through a high level P2P network.
- **data cachers** nodes operate as data agents for miners. In fact, data cachers retrieve input data from the data source or other data cachers, forward data to miners and store data locally to serve miners directly in the future.

The super-peer network allows the queries issued by miners to rapidly explore the network. The super-peer approach is chosen to let the system support several public computing applications concurrently, without requiring that miners know in advance the location of the job manager and/or of the data cachers and the data source. Super-peers are used as rendezvous points that match job queries issued by miners with job adverts generated by the job manager.

The algorithm is illustrated with reference to Fig. 1. Firstly, the job manager partitions the data mining application in a set of tasks that can be executed in parallel. For each task, a "job advert" specifies the characteristics of the task to be executed and the related input data. An available miner issues a "job query" message to retrieve one of these job adverts. If the miner knows the location of the job manager, it delivers the query directly to it. If the location