Neurocomputing 152 (2015) 77–84

Contents lists available at ScienceDirect

Neurocomputing

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

Ontology-based quantitative similarity metric for event matching in publish/subscribe system

Hongjae Kim [a], Sanggil Kang [b,*], Sangyoon Oh [a]

[a] Department of Computer Engineering, Ajou University, 206, Worldcup-ro, Yeongtong-gu, Suwon, Gyeonggi, 443–749, Republic of Korea
[b] Department of Computer and Information Engineering, Inha University, 100, Inha-ro, Nam-gu, Incheon, 402-751, Republic of Korea

ARTICLE INFO

Article history:
Received 18 October 2013
Received in revised form
12 October 2014
Accepted 4 November 2014
Available online 22 November 2014

Keywords:
Event matching
Ontology
Publish/subscribe system
Quantitative similarity metric
Semantic ambiguity

ABSTRACT

Naval ships can achieve information superiority using publish/subscribe systems that integrate heterogeneous applications. The performance of publish/subscribe communication systems depends on the effectiveness of the event matching between events and subscribers. Semantic Web technologies, including ontology, provide a platform and tools for event matching in publish/subscribe systems. Since semantic ambiguity exists in the ontologies, however, matching performance is not ideal. To improve event matching performance, we propose an ontology-based quantitative similarity metric where the frequency of keywords is used to measure the quantitative similarity. When compared to another ontology technique for semantic matching, the approximate semantic matching method, our evaluation results show a performance improvement in precision, recall, and F-measure.

1. Introduction

In the publish/subscribe communication paradigm, subscribers declare their interests to an event brokering system (i.e., a publish/subscribe event broker) to receive certain kinds of events. A distinct advantage of publish/subscribe systems over conventional client/server systems that requires a direct and tight connection between a sender and a receiver is the strong scalability that results from decoupling the information senders and receivers. An information sender (a publisher) is not tightly coupled with an information subscriber (a subscriber) since an event broker sits in-between two participants and plays a proxy for information delivery. We can add and remove participants dynamically because of this loosely coupled-ness in both time (asynchronous) and space (location transparency). Thus, it is easier to build a large-scale distributed system that comprises hundreds of participants.

Designing a distributed application with coupled participants is a challenging task. Further, since it is likely that the participants are disparately located and possess varying programming characteristics, the building process can also be burdensome and error-prone. In contrast, the publish/subscribe paradigm supports a flexible and dynamic communication model that can overcome the constraints of a point-to-points communication models (e.g., a client/server communication model). By introducing a message broker in the middle (1) participants are decoupled from each other (e.g., locations of participants can be transparent to others since they only communicate with the broker), and (2) programming interfaces are transparent to participants with heterogeneous platforms. Consequently, the use of a publish/subscribe paradigm for communication greatly facilitates construction of large-scale heterogeneous distributed systems.

Compared to the conventional point-to-point and synchronous communication paradigm, however, there is much room for improvement with respect to communication performance. Since there is an additional layer (i.e., message broker) between information senders and receivers, an efficient event matching algorithm is required in the message broker. Despite various matching algorithms such as Tree, Binary Decision Diagram (BDD), and Distributed Hash Table (DHT) [1–6], some issues remain for supporting events in different formats and semantics. Semantic Web technologies provide a platform and tools to match events in publish/subscribe systems [7,8]. Utilizing ontology, we can improve the precision of event matching without sacrificing efficiency. Predictive automatic matching [9,10] is a method for predicting future subscriptions or events. However, because semantic ambiguity exists in the ontologies, the requisite matching performance is seldom achieved.

To address this problem, we propose a similarity event matching approach that uses quantitative measures of the event ontology for publish/subscribe system. Instead of predicting subscriptions and events, as in previous approaches [9,10], we perform event matching using semantic similarity between keywords. In this article, among many large-scale software systems that use publish/subscribe system as their communication methods target domain is military software. The evaluation results show an apparent performance improvement

* Corresponding author. Tel.: +80 32 860 8377; fax: +80 32 865 0480.
E-mail address: sgkang@inha.ac.kr (S. Kang).

http://dx.doi.org/10.1016/j.neucom.2014.11.017
0925-2312/

over popular approaches such as approximate semantic matching method [8] in precision, recall, and F-measure.

The remainder of this article is organized as follows. We present related work in Section 2, and the publish/subscribe communication system architecture for naval ships is described in Section 3. We present a similarity metric for event matching in Section 4. Simulation experiment results are discussed in Section 5, and we conclude our article in Section 6.

2. Related works

In this section, we discuss work related to our study. First, we present background for the military system used here, then we discuss various event matching approaches used in publish/subscribe systems.

Information superiority is an important factor in modern warfare [11]. "Finding the enemy before they find us" is critical in naval battles. One of the most important factors in achieving information superiority is seamless communications between battlefield platforms. However, communication systems installed on naval ships are predefined, so it is difficult to achieve information superiority with the current communication models. The publish/subscribe system is well suited to this purpose since it reflects the dynamic nature of the problem and removes the negative effects of heterogeneity. Data distribution service (DDS) of Real-Time Innovations, Inc. (RTI) [12] is a popular choice for event brokering in military software systems.

There are two kinds of publish/subscribe schemes: topic-based and content-based schemes. In topic-based publish/subscribe systems, all events are published and received through individual topic channels [13], making the topic-based publish/subscribe system similar to group communication. In the content-based publish/subscribe system; all events are published and received based on the actual event content. In other words, events are not classified according to a predefined topic, but rather according to their properties. Thus, event matching is a key design issue in content-based publish/subscribe systems.

Various event matching mechanisms for publish/subscribe systems have been studied, such as Tree, BDD, and DHT [1–6]. Aguilera et al. [1] proposed a tree matching algorithm for content-based publish/subscribe systems. In their approach, the broker processes the set of subscriptions into a matching tree. Then, a parallel search tree is traversed to find leaf nodes that represent matched subscriptions. Under a large number of workloads, however, performance is limited. In response, Sadoghi et al. [2] present a BE-Tree to improve performance of the tree matching algorithm. They introduce a two-stage space-cutting technique (i.e., space partitioning and space clustering) that is used to construct a BE-Tree. A BE-Tree is dynamically constructed and has a self-adjusting mechanism that adapts as workloads change.

In event matching, time is an important factor and must be minimized. Campailla et al. [3] proposed a matching algorithm based on BDD for large-scale publish/subscribe systems. They claim that the bottleneck in very large-scale publish/subscribe systems is the event matching process. In their approach, BDD is a compact data structure for representing Boolean functions and subscriptions. Li et al. [4] use Modified Binary Decision Diagrams (MBDDs) to represent multiple subscriptions instead of having one BDD for each subscription. Consequently, subscriptions are merged and event matching time is reduced.

Centralized publish/subscribe systems have a global image of the system, which simplifies event matching. However, this approach suffers from scalability issues. Decentralized publish/subscribe systems have been studied to address this problem. Jafarpour et al. [5] present a distributed publish/subscribe framework based on a DHT. The framework consists of a set of stable brokers that are connected with an overlay network. In their approach, each client (i.e., publisher and subscriber) connects to one of the brokers, and contents are partitioned among the set of brokers. Each broker maintains subscriptions and is responsible for event matching. Although DHT can support exact-matching of equality predicates, it has difficulty supporting the range predicate. Triantafillou et al. [6] proposed a Chord-based publish/subscribe system that supports range predicates. Their approach leverages the advantages of DHT for scalability, self-organization, and performance; however, some problems remain in expressiveness of the publish/subscribe system. For example, existing publish/subscribe systems cannot support events in different formats and semantics [7].

Several researchers have adopted Semantic Web technologies for solving the problem of expressiveness. Wang et al. [7] proposed an ontology-based publish/subscribe system that makes use of event semantics and supports complex data structures. In their approach, published events are converted into an RDF graph to support complex formats. Then, they use a graph matching algorithm to match events, and subscribers receive events in the RDF format. Hasan et al. [8] proposed an approximate semantic matching technique for heterogeneous events. They claim that current publish/subscribe systems add explicit dependencies between interacting parties by assuming mutual agreement on event semantics. This mutual agreement can limit the scalability of publish/subscribe systems. To address semantic coupling and support heterogeneous events, they proposed vocabulary-free subscriptions with approximate semantic event matching.

In publish/subscribe systems, subscribers cannot receive events without sending a subscription to a broker. For example, if a subscriber forgets to send the subscription or a subscription is not delivered to the broker because of a network problem, then subscribers cannot receive the events. Thus, despite the fact that subscribers want to subscribe to certain events, they are unable to receive the information. Predictive automatic matching has been studied to address this issue. Muthusamy et al. [9] proposed predictive matching for publish/subscribe systems. Given assumptions about previous subscriptions, they predict the likelihood that a subscription will match in the future. Muthusamy et al. use a Markov model to perform the prediction and predict a future subscription using past subscriptions and the event stream. Hassan et al. [10] present a dynamic and fast matching algorithm for publish/subscribe systems. In their approach, they store published events using the Open Grid Services Architecture-Data Access and Integration (OGSA-DAI) [14], which is an extensible framework for data access and integration. Stored events are used to find trends in past events and to forecast and predict future events.

The above studies use predictive automatic matching, which requires future subscriptions or events to be predicted. Event matching performance is therefore affected by the precision of subscription or event predictions. To address the above problems, we propose an ontology-based quantitative similarity metric for event matching in publish/subscribe systems. Instead of predicting subscriptions and events, event matching is performed using semantic similarity between keywords extracted from the event and keywords found in historical events collected by subscribers. The detailed similarity calculation method is presented in Section 4.

3. System architecture

Many systems, applications, and communications are used to achieve information superiority with naval ships. However, since the systems are predefined, it is difficult to integrate them into a collaborative environment to achieve information superiority. The publish/subscribe system is well suited for efficient communication between naval ships. In this section, we present the overall