Building an integrated CBR-Big Data Oriented Architecture based on SEASALT

PhD Proposal

Kareem Amin

German Research Center for Artificial Intelligence, Knowledge Management, Trippstadter Straße 122, 67663 Kaiserslautern, kareem.amin@dfki.de

Abstract. The growth of intensive data-driven decision-making is now being recognized broadly. In this paper I propose a CBR – Big Data oriented architecture based on the SEASALT architecture. SEASALT will be enhanced to be compliant with Big Data frameworks. I will use the state-of-the-art/best practices approaches for managing Big Data and CBR. I will go through the process starting from gathering data stage till building a CBR system that is able to answer streams of questions and come up with accurate retrieved results in a reasonable time.

Keywords: Case-Based Reasoning, Big Data, Distributed Architecture, SEASALT, Multiple Agents

1 Introduction

Data is being generated extremely fast — a process that never stops like the data generated from social media networks. Facebook, the most active of social network, with over 1.4 billion active monthly users, generates the most amount of social data – users like over 4 million posts every minute – 4,166,667 to be exact, which adds up to 250 million posts per hour [3]. With the volume of data growing at unprecedented rate; Case-Based Reasoning (CBR) has a new challenge to deal with this large amount of data. Over the last years, CBR has proved its efficiency in different domains. CBR's rule of thumb is based on reusing the experiences and by this avoiding having to "re-invent the wheel". It is using the natural human reasoning of looking back to find the most similar cases that could help to solve the new issues. Since all companies now have a lot and different data sources, over days the size of data is getting bigger and the need to deal with this amount of data seamlessly is also increasing (e.g., electronic manuals, history of failure, electronic medical records) [10,12]. Therefore, the key factor for the next generation of CBR applications is the ability to deal with the complex and large amount of data that is generated every day and every moment. I need to use new distributed technologies to be able to scale CBR solutions up, to find new ways to

overcome the cases retrieval latency problems that might be crucial in critical systems (e.g., security systems, condition monitoring, sensors data, etc.).
This research proposal aims to enhance the SEASALT architecture [13, 14] by combining the Multiple Agent CBR System benefits with Big Data processing capabilities, in order to get a generic coherent architecture that can be applied with the future CBR systems.

2 Background

This section describes briefly the motivation for working with Big Data and Case-Based Reasoning and gives a short overview of the related work that has been carried out to achieve a similar goal.

2.1 Big Data, Big Challenges

Big Data management has very big challenges and opportunities, since almost everything today is generating data. Big Data management is not anymore driven by computer scientists or researchers, it is a must for companies today to benefit from their data, or they will not be able to survive in the current fast changing business environment [18]. Big Data management plays a major role in the competition between companies.
Big enterprises like SAP, IBM, Google, Microsoft, SAS and EMC are running now with maximum speed to build the most advanced Big Data platforms, not only from the software side, but also from the hardware one. They aim to attract new customers and help companies to gain the maximum benefit from their data.
The new amount of data requires new, innovative technologies to be able to give the results in a reasonable time [1].

The Big Data term refers to dynamic, large, structured and unstructured volumes of data generated from [2]:

- Traditional data sources – includes the transactional data created from ERP systems, CRMs, web store transactions, etc.
- Machine generated data – includes sensors data, smart meters, web logs, etc.
- Social data – includes data generated from social networks like Facebook, Twitter, LinkedIn, etc.

2.2 Distributed Case-Based Reasoning

CBR needs efficient techniques to manage its subtasks such as collecting and formatting data, case base maintenance, cases retrieval, cases adaptation and retaining new cases. From this point of view, the need to build distributed CBR systems for maximum efficiency increased. Multiple Agent CBR systems are widely used and very well known in Distributed CBR systems area, a lot of frameworks and related work have

been carried out elaborating different architectures and techniques to manage the CBR sub-tasks.

Most researches in Distributed CBR concentrated more on distributing resources within the CBR architecture but not on distributing the case base itself. One of the successful distributed CBR platforms is jCOLIBRI [4]. It supports the development of wide range of CBR software, it provides the required infrastructure to implement CBR systems [5, 6]. jCOLIRBI is depending on multiple agents to perform the subtasks associated with CBR. Multi Agent Systems (MAS) distribute the case base itself and/or some aspects of the reasoning among several agents [7]. I can categorize the research efforts in the area of distributed CBR using two criteria [8]:

1. How knowledge is organized/managed within the system (i.e. single vs. multiple case bases).
2. How knowledge is processed by the system (i.e. single vs. multiple processing agents).

Another example of successful CBR tools is myCBR. myCBR is a joint effort of the Competence Centre CBR at DFKI, Germany, and the School of Computing and Technology at UWL, UK (see http://mycbr-project.net/). myCBR Workbench [9] provides user-friendly graphical user interfaces for modelling various kinds of attribute-specific similarity measures and for evaluating the resulting retrieval quality. In order to reduce also the effort of the preceding step of defining an appropriate case representation, myCBR Workbench includes tools for generating the case representation automatically from existing raw data. The accompanying Software Development Kit (SDK) allows for integration into other applications and extension to specific requirements such as additional similarity calculations. Agent-based systems technology has generated lots of excitement in recent years because of its promise as a new paradigm for conceptualizing, designing, and implementing software systems [17]. In [13, 14] the SEASALT architecture is an application- independent architecture to work with heterogeneous data repositories and modularizing knowledge to be structured. It was proposed based on the CoMES approach to develop collaborative multi-expert systems. SEASALT aims to provide a coherent multi-agent CBR architecture that can define the outlines and interactions to develop multi-agent CBR systems. The SEASALT team has applied it in [14] to travel medicine as part of the docQuery project. It was as a textual CBR application domain to showcase how SEASALT could be used. In [15] Albert Pla at al. have provided a user friendly tool for medical prognosis (eXiT*CBRv2). They proposed an innovative multi-agent system architecture, in which they have a coordinator agent that is responsible for receiving new cases, then pass it to *n* agents. Each agent is connected with case base to retrieve cases based on different retrieval calculations. Afterwards, they all pass the results again to the coordinator agent to assess and compare results and at the end it gives the final results. They illustrated the use of the tool through several experiments carried out with a breast cancer database and they show how easy it is to compare distributed approaches that maintain naturally distributed clinical organization, compared to centralized systems.

Generally, CBR and MAS have proved efficiency with different successful distributed CBR systems. Currently systems are getting more complex and agents need to be

smarter to be able to deal with its environments. MAS bring a lot of advantages and benefits to CBR but also have a lot of challenges and issues that should be taken into consideration while building systems:

1. How do we design our algorithms to decompose tasks to agents and allocate problems to them?
2. If systems are widely distributed, how are agents going to communicate and what communication protocols will they use?
3. What if we lost the communication between agents?
4. How do we ensure that agents are working properly and every single agent is doing its task in the perfect manner?
5. How do we troubleshoot issues across all agents?

2.3 Case-Based Reasoning & Big Data

CBR and Big Data collaboration is an emerging topic, some researches and efforts have been carried out in this area. Since the growth of digital data is widely heralded. A 2014 article estimates that "Almost 90% of the world's data was generated during the past two years, with 2.5 quintillion bytes of data added each day" [10]. In [11] Yu-Hui X & Xiao-Yun Tian provided a CBR model *NT-CBR* based on the data mining technology *NT-SMOTE*. They have tried to solve the problems associated with enterprise risk management and compared their results with different methodologies. The NT-CBR model used big internet data to do the forecasting of risks and give smarter and faster solutions to the risk. In [12] Vahid Jalali & David Leake have initially developed *ensembles of adaptation for regression* (EAR), a family of methods for generating and applying ensembles of adaptation rules for case-based regression. That model suffered from high computational complexity and therefore they decided to go to Big Data techniques (Map Reduce) to improve their model performance and they called it BEAR. BEAR uses MapReduce and Locality Sensitive Hashing (LSH) for finding nearest neighbors of the input query. It consists of two main modules: LSH for retrieving similar cases and EAR for rule generation and value estimation. As a conclusion they got very promising results that encourage them to perform bigger experiments to ensure that the model is reacting in the perfect manner.

3 Research Focus

In my research I will focus on integrating several methodologies, CBR, Big Data, Data Streaming, Multi-Agent Systems and SEASALT as a background architecture orchestrating the interaction between different entities. I mainly aim to address the handling and processing of big case bases to avoid cases retrieval latency. I will also extend the SEASALT [13, 15] architecture to be compliant with Big Data systems architectures. It is also important to mention that at **ICCBR 2016** a dedicated workshop will take place to discuss all time problems related to CBR systems like real time data processing and big data (see http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=55489&copyownerid=89330).

3.1 Motivation

My work is motivated by the study done on the prior work related to CBR, MAS, and Big Data. I aim to build a big data oriented multi-agent CBR architecture. The main challenge is how I will develop a framework for integrating all these methodologies into one robust architecture and orchestrating the interaction between them all, which will allow CBR systems to deal with big case bases.

3.2 Research Problem

As the case base grows, the swamping utility problem could affect the case retrieval times, degrading system performance and credibility [12]. For example, calculating metrics such as number of visitors or page views for a social media or e-commerce web site with hundreds or million users is a common practice in industry, but in current CBR research, experiments with tens or thousands of cases, or even much fewer, are common. The proposed approach is trying to focus on building scalable CBR systems that able to work with the big case bases and being able to deal with online data streams. I will work to integrate the new Big Data frameworks like Spark or Flink with a Multi-Agent CBR Framework (JADE) in order to build an innovative solution based on the proposed architecture.

3.3 The Proposed SEASALT – Big Data Oriented Architecture

One key innovation of the SEASALT – Big Data Oriented architecture is to improve the problem-solving technology of CBR at the level of big data with the help of the modern frameworks that distribute processes for better performance.

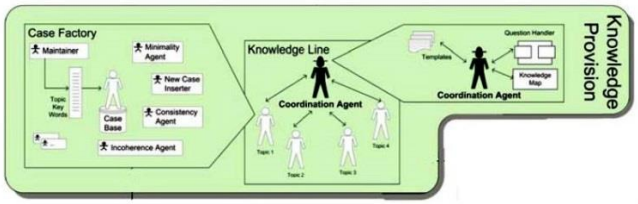

Fig. 1. SEASALT Knowledge Provision Layer

Figure 1 depicts the Knowledge Provision Layer in SEASALT, in which the coordination agent is connected with a number of topic agents. Every topic agent is considered as a standalone CBR system with its own case factory.

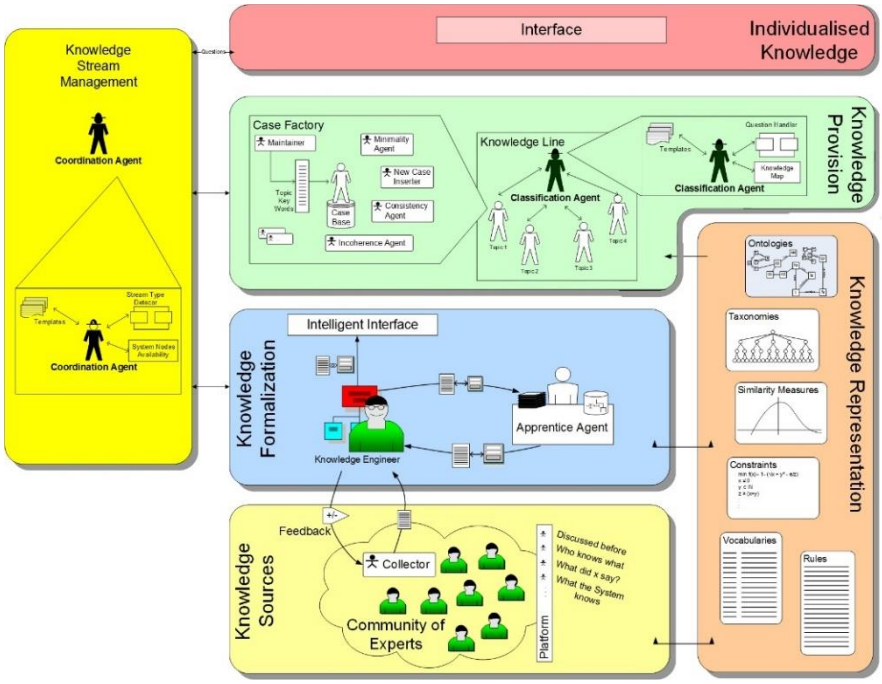

Fig. 2. SEASALT-Big Data Oriented Architecture

Figure 2. Illustrates the changes made to the Knowledge Provision layer and the new Knowledge Stream Management layer. The Knowledge Stream Management layer is getting two types of input, one from Knowledge Formalization layer that is related to new data insertion, and the other input is coming as stream of questions from the Individualized Knowledge layer. According to our architecture, system nodes would be the available processing power – *Node = Single Processing Power*. The Knowledge Provision layer will be distributed across several nodes, and hence each node contains Knowledge Provision agents. The Coordination Agent will act as the system manager who is aware of all the system nodes and responsible for the whole system control. He will be the *data tap* that uses the underlying framework to distribute the incoming requests across the system nodes. Normally, there are two kinds of nodes, one for *Queries* processing to retrieve results and the second for *New Cases* processing. It is possible to have up to N nodes in the system according to the hardware availability. The more nodes, the better performance we get. In every node, there is a Classification Agent to classify the received data and assign it to the intended Topic Agent. Each Classification Agent is aware of the knowledge map gathered from knowledge sources and classify

the incoming requests according to predefined classes. Then, the Classification Agent assigns the request to the intended Topic Agent(s). The Topic Agent is performing queries to retrieve the most similar cases. Since I use distributed nodes in the hardware cluster, the Case Base will be replicated to avoid data integrity using the replication channels to replicate data between the Case Base instances (see Figure 2). Since our Case Base will be distributed among several nodes, the Case Factory agents will be centralized. Therefore the Case Factory will have only one instance that performs case maintenance on a single Case Base. Afterwards, the results will be distributed to the whole system nodes using the replication channels.
This architecture is addressing the domains/tasks that require CBR and are in need for Big Data processing in the same time (e.g., Anomaly Detection, Condition Monitoring, Medical Diagnosis, etc.). A cooperative study with the AGATA (Analyse großer Datenmengen in Verarbeitungsprozessen, engl.: analysis of large amount of data in manufacturing processes) project [16] at DFKI is going to be established to showcase the performance and evaluate the proposed approach.

4 Conclusion and Future Directions

In this PhD proposal I have proposed and presented the SEASALT-Big Data oriented architecture. It is an integrated Big Data-oriented version of the SEASALT architecture that aims to tackle Big Data problems with CBR**.** In summary,

1. I want to deal with Big Data where processing requires manual knowledge modeling in addition as context/background knowledge.
2. I want to improve CBR to be able to deal with Big Data.
3. I want to develop a methodology for developing CBR-Big Data oriented applications.
4. I want to evaluate the developed methodology and tools based on AGATA and other applications.

As future directions, I would explore more literature and related work for joint researches between CBR and Big Data. I will also be looking for data sets in order to test the implementation and compare it with others.

References

1. A community white paper developed by leading researchers across the United States, "Challenges and Opportunities with Big Data," Purdue University, USA, 2012.
2. J. Singh, "Big Data Analytic and Mining with Machine Learning Algorithms," *International Journal of Information and Computation Technology,* 2014.
3. George Simos, [Online]. Available: http://wersm.com/how-much-data-is-generated-every-minute-on-social-media/, [Accessed 19 08 2015]

4. Juan A. Recio-García, Belén Díaz-Agudo and Pedro A. González-Calero, "The COLIBRI Platform: Tools, Features and Working Examples," *Springer-Verlag Berlin Heidelberg,* 2014.
5. Belén. Díaz-Agudo, Juan A. Recio-García and Antonio A. Sánchez-Ruiz-Granados, "Building CBR systems with JCOLIBRI," Special Issue on Experimental Software and Toolkits of the Journal Science of Computer Programming, pp. 68-75, 2007.
6. Juan A. Recio-García, Belén. Díaz-Agudo, Sergio González-Sanz and Lara Q. Sanchez, "Distributed deliberative recommender systems," *T. Computational Collective Intelligence,* pp. 121-142, 2010.
7. Enric Plaza and Lorraine Mcginty, "Distributed case-based reasoning," *Knowledge Engineering Review 20,* p. 261–265, 2006.
8. Aitor Mata, "A Survey of Distributed and Data Intensive CBR Systems," *Springer-Verlag Berlin Heidelberg,* pp. 582-586, 2009.
9. T. Roth-Berghofer, J. Antonio R. García, Christian S. Sauer, Kerstin Bach, KD. Althoff, B. D. Agudo and P. A González Calero, "Building Case-based Reasoning Applications with myCBR and COLIBRI Studio," in *ICCBR*, Lyon, 2012.
10. Gang-Hoon Kim, Silvana Trimi and Ji-Hyong Chung, "Big-data applications in the government sector," *Coomunications of the ACM,* pp. 78-85, 2014.
11. Yuhui Xu and Xiaoyun Tian, "Internet Big Data Information Analysis and Power Intelligent Automation Risk Prediction Based on Case Based Reasoning," in *3rd International Conference on Machinery, Materials and Information Technology Applications*, Qingdao, 2015.
12. Vahid Jalali and David Leake, "CBR Meets Big Data: A Case Study of Large-Scale Adaptation Rule Generation," *Case-Based Reasoning Research and Development,* pp. 181-196, 2015.
13. K. Bach, M. Reichle and Klaus-Dieter. Althoff, "A Domain Independent System Architecture for Sharing Experience," in *Workshop Wissens- und Erfahrungsmanagement*, 2007.
14. Meike Reichle, Kerstin Bach and Klaus-Dieter. Althoff, "Knowledge engineering within the application-independent architecture SEASALT',," *International Journal Knowledge Engineering and Data Mining,* vol. 1.1, no. 3, pp. 202-215, 2011.
15. Albert Pla, B. Lopez, P. Gay and C. Pous, "eXiT*CBR.v2: Distributed case-based reasoning tool for medical prognosis," *Decision Support Systems,* vol. 54, no. 3, p. 1499–1510, February 2013.
16. S. Windmann, A. Maier, O. Niggemann, C. Frey, A. Bernardi, Ying Gu, H. Pfrommer, T. Steckel, M. Kruger and R. Kraus, "Big Data Analysis of Manufacturing Processes," in *12th European Workshop on Advanced Control and Diagnosis*, 2015.
17. Katia P. Sycara, "Multiagent Systems," *AI Magazine,* 1998.
18. "Better business outcomes with IBM Big Data Anlytics," 2016. Available http://www935.ibm.com/services/multimedia/59898_Better_Business_Outcomes_White_Paper_Final_NIW03048-USEN-00_Final_Jan21_14.pdf, [Accessed 01 07 2016].