Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 542764, 7 pages
http://dx.doi.org/10.1155/2014/542764

Research Article

Platform to Build the Knowledge Base by Combining Sensor Data and Context Data

Sungho Shin,[1,2] Jungho Um,[1] Dongmin Seo,[1] Sung-Pil Choi,[1] Seungwoo Lee,[1] Hanmin Jung,[1] and Mun Yong Yi[2]

[1] *Department of Computer Intelligence Research, Korea Institute of Science and Technology Information, 245 Daehak-ro, Yuseong-gu, Daejeon 305-806, Republic of Korea*
[2] *Department of Knowledge Service Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea*

Correspondence should be addressed to Seungwoo Lee; pinesnow.lee@gmail.com

Received 29 August 2013; Accepted 21 December 2013; Published 10 February 2014

Academic Editor: Hwa-Young Jeong

Sensor data is structured and generally lacks of meaning by itself, but life-logging data (time, location, etc.) out of sensor data can be utilized to create lots of meaningful information combined with social data from social networks like Facebook and Twitter. There have been many platforms to produce meaningful information and support human behavior and context-awareness through integrating diverse mobile, social, and sensing input streams. The problem is that these platforms do not guarantee the performance in terms of the processing time and even let the accuracy of output data be addressed by new studies in each area where the platform is applied. Thus, this study proposes an improved platform which builds a knowledge base for context awareness by applying distributed and parallel computing approach considering the characteristics of sensor data that is collected and processed in real-time, and compares the proposed platform with existing platforms in terms of performance. The experiment shows the proposed platform is an advanced platform in terms of processing time. We reduce the processing time by 40% compared with existing platform. The proposed platform also guarantees the accuracy compared with existing platform.

1. Introduction

Once information extracted from texts is well organized with each other, it evolves to and gives the knowledge used for supporting human's decision. For example, the entities such as names of person, location, organization, and technical terms are extracted from texts, and they can be related with each other. The entities and their relations from texts are one of the most useful knowledge recently. The technological knowledge itself is much valuable when used for practical services to support humans such as an intelligence service and a decision support system. What is significantly considered in terms of the performance is to reduce processing time to achieve goals of a platform. On the other hand, as smart phones are supplied to each individual and furthermore a variety of sensors used for convenience are deployed throughout the residential environment of the people, even invisible around us, there exist a huge amount of sensor networking and sensor data. Sensor data is getting utilized in a wide range of areas recently. Even though sensor data is a kind of structured data, it can be thought of as meaningless data because it is the signal itself which is simply generated by sensors. It also contains much overhead [1]. However, life-logging data (time, location, etc.) out of sensor data can be utilized to create the individual's life stories, when combined with other semantic information [2]. From this perspective, as its input data such as papers, patents, and web articles includes the sensor data, the proposed platform can be considered a class of platform which takes advantage of sensor data.

In this research, we propose an improved platform which builds a knowledge base to support human behavior and context-awareness utilizing context data and sensor data, and also compare its performance with existing platforms. The main idea is to apply a machine learning method based on the distributed and parallel environment to the new platform.

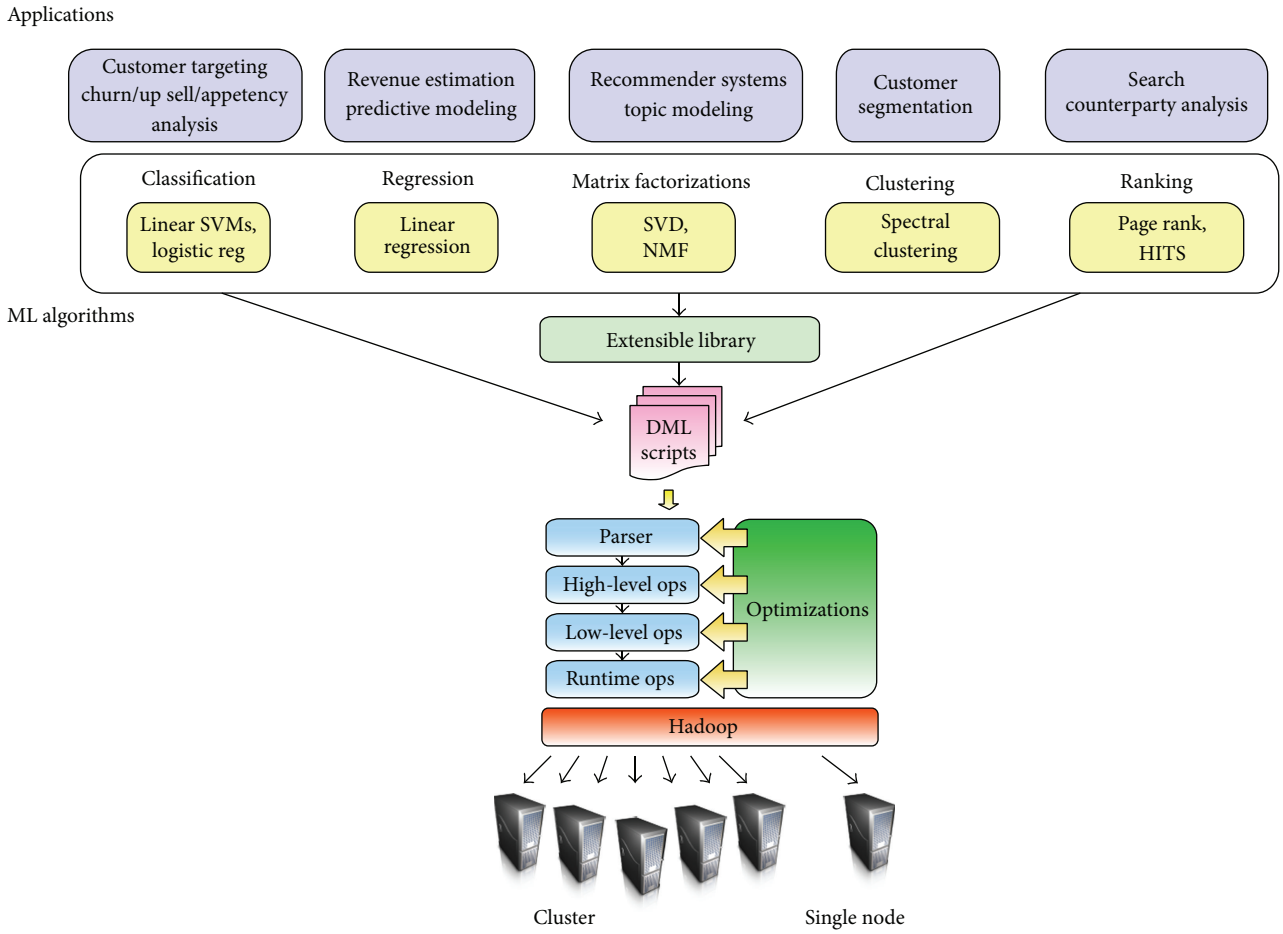

Figure 1: SystemML architecture.

Existing platforms lack speed in processing data. In addition, a certain platform lets the accuracy be addressed from new studies in each area where the platform is used [3]. The new platform we propose challenges the issues, and it achieves better result compared with existing platforms.

2. Related Work

Existing platforms have some limitations of the processing time to extract knowledge from data including real-time and streaming sensor data. We first investigates some big data processing methods, focusing on the distributed and parallel computing based machine learning methods, and then reference the architectures of them for designing the new platform.

It is generally known that machine learning methods are more preferred recently than handcrafted rules on which early studies were mostly based [4]. If the training data to learn platforms is large enough to guarantee the quality of extraction, machine learning methods are also more accurate than other methods.

SystemML is a platform developed by IBM to enable a variety of machine learning algorithms to be executed in a MapReduce based distributed processing environment [5]. In Figure 1, machine learning tasks controlled by a Declarative Machine leaning Language (DML) script are compiled through the High-Level Operator (HOP), and the Low-Level Operator (LOP), and executed in the MapReduce environment for parallel processing. Methods of machine learning, for example, linear regression, descriptive statistics, and linear Support Vector Machines (SVMs), are provided. SystemML is important in that the existing knowledge extraction algorithms are implemented to be driven in the distributed processing environment in order to process big data.

Mahout is also intended to provide a variety of ML algorithms through Mahout as a library. The idea is that the library is to extend the library effectively in a cloud environment by using the Apache Hadoop to solve the issue of processing time taken to learn a large data set which is one of disadvantages of existing machine learning algorithms (http://mahout.apache.org/). Exemplary open sources include Lucene in charge of preprocessing of machine learning, Hadoop which enables the machine learning algorithm to be executed in the distributed processing environment, and Hama which enables MapReduce to be effectively used. From the above related works, the implication is that machine learning methods are executed in a distributed and parallel

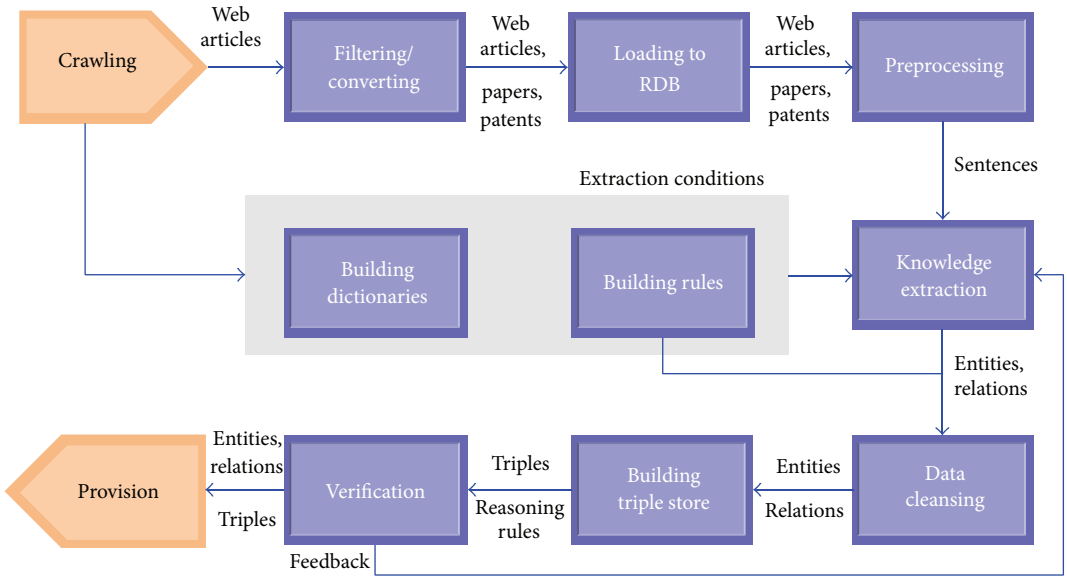

Figure 2: Process of the existing platform.

processing environment based on MapReduce so that the performance is better than that of single machine.

The sensor network recently has expanded to the meaning of wireless network because of many wireless sensors. The wireless sensor network is composed of distributed autonomous sensors embedded in objects to check physical or environmental situations, for example, brightness, temperature, sound, and pressure, and to cooperatively hand over their data through the network to a main location. As a type of wireless network devices, mobile devices are everywhere and are almost available at anytime. They also can be used to deliver location and notify a user's identity/presence in a room or place [6, 7]. However, mobile devices alone are not enough to provide a full picture of context, and in particular go through a difficult time inferring an individual's preferences. Social networks can provide detailed contextual information mentioning individual's interests and preferences. Accordingly, data from these three networks, sensor network, mobile network, and social network are combined together and enhance the understanding of context surrounding humans. This is one of research trends and applications of sensor network.

3. Analysis of the Existing Platform

Figure 2 shows the process of technological knowledge extraction used for the InSciTe Adaptive service. InSciTe Adaptive service is a user adaptive intelligent service to support making decisions on things related to technologies or products. In order to supply the knowledge base for the service, the existing platform applies a rule-based information extraction method, and the process contains some postprocessing tasks to guarantee the quality of data. After extraction, triple store is made with semantic triples to extend the outputs of extraction. The platform addresses 5.3 million web articles, 9.8 million scientific papers, and 7.6 million patents, and it makes about 500 million semantic triples, which takes approximately 5 days.

The detail explanation on each activity is as follows.

Crawling. The platform collects web articles (HTML) from multiple websites. They are metadata and abstracts for papers and patents.

Filtering/Converting. Filtering error data and converting data format from HTML to XML follow the crawling. Web articles collected in HTML form can be converted into XML form that has an advantage of easy parsing.

Loading to RDB. This is a beginning job of knowledge extraction loading raw data in XML form to RDB (MySQL).

Preprocessing. The platform makes parsing individual documents and separating them into sentence and assigning a sentence-specific unique identifier number. Also, it performs morphological analysis for each sentence as well as POS (Part of Speech) tagging.

Knowledge Extracting. With the entity dictionary, patterns, and rules, the platform extracts entities and relations between entities in the sentences, and adds URI to the entities and the relations.

Data Cleansing. This is to refine incorrectly extracted entities and relations by applying stop word lists or refinement rules.

Building Triple Store. This generates triples and loads in the retrievable stores by applying ontology for extracted relations.

Verification. The platform verifies final entities and triples. When errors are found, it analyzes the causes of the error and then reflects them in the extraction task.

Provision. The last step is to provide data to the service that uses the processed data.

The existing platform has mainly two drawbacks. One is that it takes more time to extract technological knowledge

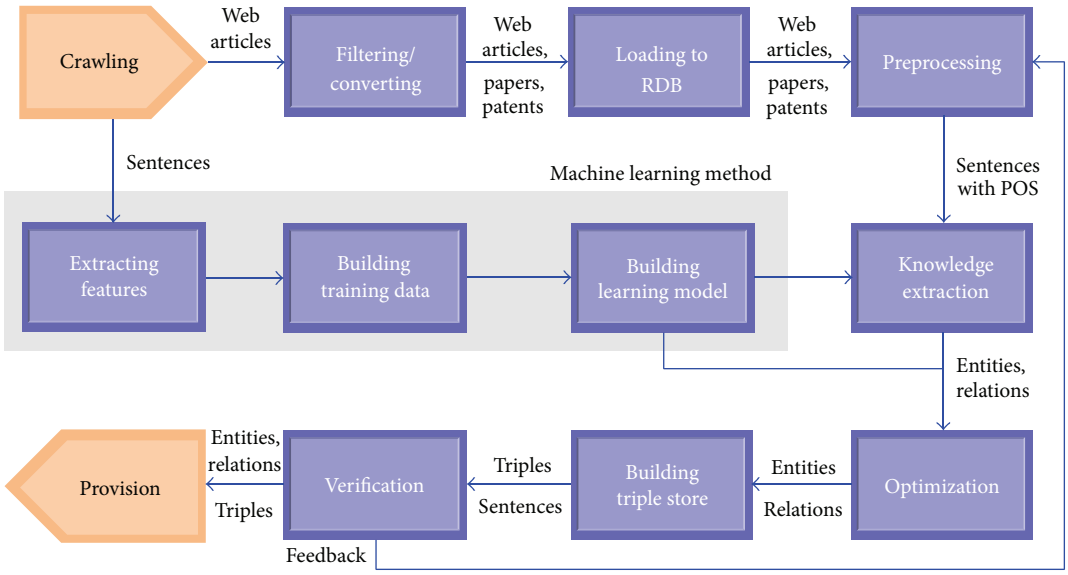

Figure 3: Process of the proposed platform.

and build the knowledge base than it is expected because the platform is implemented to run on single machine. The platforms which operate on single server are not able to deal with large amounts of data due to physical resource limitations. It influences the accuracy of the information extraction result. The other is that it adopts a rule-based knowledge extraction method which is typically domain dependent and requires a high cost with significant amount of manual efforts [8].

4. Proposed Platform

We propose a new technological knowledge extraction platform including data collection whose process is shown in Figure 3. Considering the implications of related works, unlike the existing platform, the platform is equipped with machine learning method as well as rules and entity dictionary and is executed in a distributed and parallel environment; MapReduce framework and Hadoop file platform are applied to the new platform. Similar to the process of the existing platform, the proposed platform goes through crawling, filtering/converting, and loading to RDB. In addition, syntax analysis is further performed in the preprocessing step. Machine learning method is utilized for knowledge extraction. In this study, the structural SVM is applied. Unlike existing SVMs supporting binary classification and multiclass classification, the structural SVM supports more general structural problems (i.e., a morphological tagging, chunking, named entity recognition, parsing, etc.), and it shows better accuracy. As a preparation for using the structural SVM, first, define entity-specific features and provide the learning model consisting of combination of entity-specific quality values by using prebuilt training data. In knowledge extraction step, extract entities and relations in the sentences using the learning model. In general, as the accuracy of the knowledge extraction tool made at the beginning is not high enough, it is subjected to the performance optimization process. The process is performed through the heuristic-based simulation by adjusting the used qualities and their quality values. The rest of the process is the same as that of existing platforms.

The proposed platform is expected to spend most of the process time on extraction work. The reasons can be considered in two different ways. One is that building learning set during extraction work requires extramanual works. The other one is that actual extraction work using extractor can take longer time due to lots of target documents to be extracted. As you can see on the process, because building a learning set can be proceeding in parallel between other works rather than in sequence, it does not affect much on the overall processing time with the earlier preparation. And the problem caused by lots of target documents to be extracted is expected to be solved using distributed parallel-based modules.

The proposed architecture is based on distributed and parallel environment. Figure 4 shows each part of the proposed platform. It is composed of four parts: data collection (left side), knowledge extraction based on MapReduce and Hadoop (right side), and job management (top side). On each slaver server, the modules for the tasks such as preprocessing, information extraction, triple store construction, and reasoning are installed and executed. The master server has a knowledge extraction management module for the task management of each slave server, an input document management module for management of the first entered data, and an output document management module for the management of the final output data. The MapReduce framework is quite fast and attractive since a cluster consisting of a large number of low-cost servers processes data in a distributed and parallel method [9]. In addition, for the purpose of storing sensor data collected in the form of real time stream after converting a triple format, we use a large

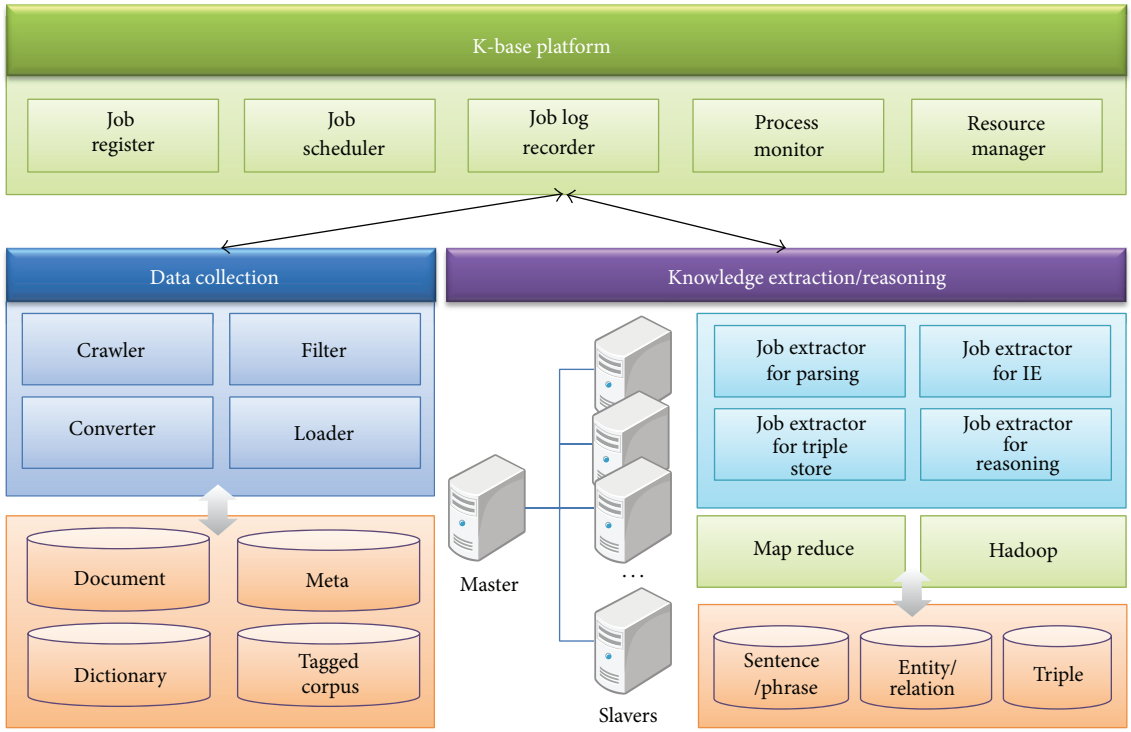

Figure 4: Architecture of the proposed platform.

triple store. It enables fast bulk loading through parallel data processing using MapReduce framework and provides the range and keyword search, and so forth by using Hbase for easy search.

One of the expected issues for the new system is related to the job management, especially to the job scheduler. The system should be automatically executed according to the sequential steps to reduce the idle time of the system. For this, the scheduler coordinates all tasks and makes them processed in order without the system waiting. This means that the scheduler controls all processes and monitors each thread generated by each module. Once one of modules fails to process the task assigned to the module, the scheduler cannot realize the failure so that it does not go forward to the next step. Even though the system fails to run a function, the scheduler should know that and should address the situation. For addressing this issue, two ways are considered: the method to use message queues for the process communication and the method to take advantage of the log table in which the status value of each job is recorded.

5. Comparison of Platforms

The process of existing platforms is mostly similar to that of the proposed platform, but still the following differences exist. The first is in the knowledge extraction method applied. Existing platforms are based on the rules and entity dictionary to extract knowledge [10]. On the contrary, the proposed platform extracts knowledge using the structural SVM as well as rules and entity dictionary. In particular, the structural SVM is a much more excellent tool for knowledge. It supports more general structural problems (i.e., parsing, morphological tagging, chunking, named entity recognition, etc.), and also it shows better accuracy than other machine learning methods [11, 12]. The platform uses 1-slack formulation of structural SVM proposed by Joachims for learning and the learning can be completed by the iteration of $O(1/e)$ while the cutting-plane algorithm takes $O(1/e^2)$ iterations [13]. Thus, the proposed platform is more helpful to rapidly extract knowledge than the existing platform. The second is in the activities after knowledge extraction. Knowledge extraction in the existing platforms is followed by data cleansing. Due to the nature of rule-based extraction method, it generates lots of error data; therefore, data cleansing should be an essential process to ensure the data accuracy. However, as data cleansing is mostly done by hand, it is labor-intensive and costly. In contrast, the proposed platform provides not only relatively lower probability of error data generation, but also higher data accuracy using automated method, even based on heuristic method. Therefore, when assuming the same level of accuracy, while the existing platform takes time for data cleansing, the proposed platform can reduce overall processing time. Furthermore, from the view of architecture, the proposed platform can shorten the extraction time significantly by applying distributed and parallel computing technique based on Hadoop and MapReduce.

The experiment is done to compare the performance of each platform and the result shows the new platform is faster than the existing one. The new platform has less processing time by 2 days than the existing platform (Table 1). We used

Table 1: Comparison between existing platform and new platform.

Criteria	Existing platform	New platform
Extraction method	Rules and dictionary	Rules and dictionary Machine learning and (80,000 sentences for training)
Processing environment	Single machine	Cluster (Hadoop and MapReduce)
Execution	Semiautomatic execution	Automatic execution by scheduler
Volume of input data (including sensor data)	5.3 million web articles, 9.8 million papers, 7.6 million patents	6 million web articles, 12 million papers, 8.5 million patents
Volume of output data	500 million triples	600 million triples
Processing time	5 days	3 days (reduced by 40%)

Table 2: Comparison between context-aware computing platform and proposed platform.

Criteria	Context-aware computing platform	Proposed platform
Input data	Sensor data (time, location, etc.)	Sensor data (time, location, etc.)
	Mobile data	Metadata and abstract of papers and patents
	Social data	Full-text of web articles
Major function	Collecting data (regular interval)	Collecting data (regular interval)
	Extracting context	Extracting context
	Mining data	Mining data
Components	Data collecting components	Data collecting components
	Devices controlling components	Process scheduler
	Context classification (location classifier, physical condition classifier, friendship classifier, group-based classifier)	Knowledge extractor based on Machine learning method (the structural SVM)
	—	Distributed and parallel component (Hadoop and MapReduce)

20 machines for the cluster environment for the experiment: one is the master server and others are slaves. Each machine has 8 cores and each CPU clerk is 3.5 GHz.

We also compare the proposed platform with a context-aware computing platform addressing sensor data and context data to enable appropriate context-aware output action. Beach et al. proposed a context-aware computing Platform [3]. It can take appropriate actions by being aware of the circumstances around the individuals or groups with utilizing sensor data, mobile data, and social data (Figure 5). For example, it identifies the tastes of the individuals or groups in the vicinity and chooses the movies suitable to their tastes, and allows them to watch the movies through the media.

Similarly, the proposed platform for building knowledge base in this paper also supports the organization's key decision makers or researchers to be aware of the future research or the technology trend using sensor data, papers, patents, and web articles. Two platforms can be compared in terms of data, functions, and components utilized (Table 2). The main differences between two platforms depend on the component.

The context-aware platform has each of the context classifiers for user's location, physical condition, and group activity which become key input data. These classifiers identify and are aware of the context of data collected by the devices. One limitation of their study is that they separate the areas between data collection and data classification. In addition, they leave the accuracy of data classification to another area of study. However, the accuracy of data classification is very important even within the platform. For the usage of the platform, the accuracy of data classification should be guaranteed. Only some studies did performance evaluation of data aggregation for wireless sensor network [14] or performance evaluation of a movie recommendation [15], but they are not about the platform itself. On the other hand, the proposed platform in this paper pays attention to the accuracy of the data. It utilizes machine learning method to ensure the accuracy of input data. In particular, as it uses a structural SVM which is recently getting many choices from researchers, the platform is verified in terms of accuracy. Another difference is the distributed and parallel component. As the context-aware platform is just a prototype level, it may not have considered the distributed and parallel component which is required in the real environment. However, as sensor data is collected in real time in the form of large amounts of streaming data from various devices, the distributed and parallel component is required to process these data effectively. The proposed platform in this paper, which is based on Hadoop and MapReduce, supports the distributed and parallel computing. In the above two aspects, the proposed platform is expected to show a little better performance even in the context-aware applications.

6. Conclusion

We have reviewed the existing platforms to generate meaningful information and help human behaviors and context-awareness through data from mobile devices, social networks, and sensors around us. They lack in performance in terms of data processing time and accuracy of output data. That is, these platforms do not guarantee the performance in terms of the processing time and even let the accuracy of output data be addressed by new studies in each area that the platform is being applied in. Compared with the existing platform, the new platform is equipped with the distributed and parallel computing technology using the MapReduce and the Hadoop. The platform is also based on a machine learning method with external resources. The two main changes are applied to the process and architecture of the new platform.

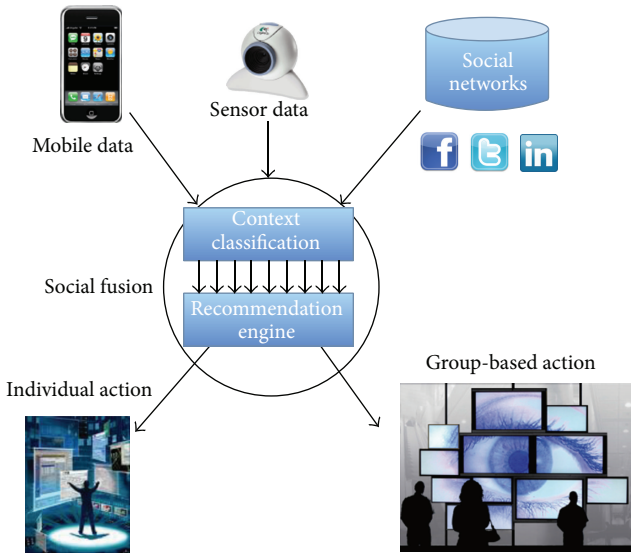

Figure 5: How to fuse mobile, sensor, and social data to generate context awareness.

One of the limitations in this study is that we need to compare our new platform with a context-awareness computing platform which pays attention to the accuracy of output data. This experiment will be expected to give a more interesting result.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] M. Yoon, Y. K. Kim, and J. W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, pp. 15–22, 2013.

[2] J. Reddington and N. Tintarev, "Automatically generating stories from sensor data," in *Proceedings of the 15th ACM International Conference on Intelligent User Interfaces (IUI '11)*, pp. 407–410, ACM, February 2011.

[3] A. Beach, M. Gartrell, and X. Xing, "Fusing mobile, sensor, and social data to fully enable context-aware computing," *IEEE Network*, vol. 22, no. 4, pp. 50–55, 2008.

[4] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Linguisticae Investigations*, vol. 30, pp. 3–26, 2007.

[5] A. Ghoting, R. Krishnamurthy, E. Pednault et al., "SystemML: declarative machine learning on MapReduce," in *Proceedings of the IEEE 27th International Conference on Data Engineering*, pp. 231–242, 2011.

[6] A. Beach, M. Gartrell, S. Akkala et al., "WhozThat? Evolving an ecosystem for context-aware mobile social networks," *IEEE Network*, vol. 22, no. 4, pp. 50–55, 2008.

[7] N. Eagle and A. Pentland, "Social serendipity: mobilizing social software," *IEEE Pervasive Computing*, vol. 4, no. 2, pp. 28–34, 2005.

[8] L. Chiticariu, R. Krishnamurthy, Y. Li, F. Reiss, and S. Vaithyanathan, "Domain adaptation of rule-based annotators for named-entity recognition tasks," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '10)*, pp. 1002–1012, October 2010.

[9] J. H. Um, S. Shin, Y. S. Choi et al., "A knowledge extraction system using the MapReduce framework for massive amounts of technical data," in *Proceedings of the 2nd Joint International Semantic Technology Conference*, 2012.

[10] H. W. Chun, C. H. Jeong, S. Shin et al., "Information extraction for technology trend analysis," *Advances in Information Sciences and Service*, vol. 5, no. 7, pp. 336–344, 2013.

[11] A. Ekbal and S. Bandyopadhyay, "Named entity recognition using support vector machine: a language independent approach," *International Journal of Computer Systems Science and Engineering*, vol. 2, no. 4, pp. 155–170, 2008.

[12] J. Kazama, T. Makino, Y. Ohta, and J. Tsujii, "Tuning support vector machines for biomedical named entity recognition," in *Proceedings of the ACL*, 2002.

[13] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural SVMs," *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009.

[14] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, no. 13, pp. 1–17, 2013.

[15] W. H. Jeong and S. Kim, "Performance improvement of a movie recommendation system based on personal propensity and secure collaborative filtering," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 157–172, 2013.