Ersch. in: Foundations of intelligent Systems : 20th international symposium ; proceedings, ISMIS 2012, Macau, China, December 4 - 7, 2012 / Li Chen ... (ed.). - Berlin : Springer, 2012. - S. 425-434. - (Lecture notes in computer science ; 7661 : Lecture notes in artificial intelligence). - ISBN 978-3-642-34623-1
http://dx.doi.org/10.1007/978-3-642-34624-8_48

Discovering Dynamic Classification Hierarchies in OLAP Dimensions*

Nafees Ur Rehman, Svetlana Mansmann, Andreas Weiler, and Marc H. Scholl

Department of Computer & Information Science
University of Konstanz, Germany
{nafees.rehman,svetlana.mansmann,andreas.weiler,
marc.scholl}@uni-konstanz.de
http://dbis.uni-konstanz.de/

Abstract. The standard approach to OLAP requires measures and dimensions of a cube to be known at the design stage. Besides, dimensions are required to be non-volatile, balanced and normalized. These constraints appear too rigid for many data sets, especially semi-structured ones, such as user-generated content in social networks and other web applications. We enrich the multidimensional analysis of such data via content-driven discovery of dimensions and classification hierarchies. Discovered elements are dynamic by nature and evolve along with the underlying data set.

We demonstrate the benefits of our approach by building a data warehouse for the public stream of the popular social network and microblogging service Twitter. Our approach allows to classify users by their activity, popularity, behavior as well as to organize messages by topic, impact, origin, method of generation, etc. Such capturing of the dynamic characteristic of the data adds more intelligence to the analysis and extends the limits of OLAP.

Keywords: Data Warehousing, OLAP, Data Mining, multidimensional data model, OLAP cube, OLAP dimensions.

1 Introduction and Motivation

The necessity to integrate OLAP and data mining was postulated in the late 90-es [5]. Today, a powerful data mining toolkit is offered as an integrated component of any mature data warehouse system, such as Microsoft SQL Server, IBM DB2 Data Warehouse Edition, Oracle, and others. Data mining tools require the input data to be consolidated, consistent and clean. OLAP cubes – where the extracted data undergoes exactly this kind of transformation – appear to be perfect candidates to harbor data mining algorithms. In a standard data warehouse system architecture, data mining functionality resides at the upper layer

* This work was partially supported by DFG Research Training Group GK-1042 "Explorative Analysis and Visualization of Large Information Spaces", University of Konstanz.

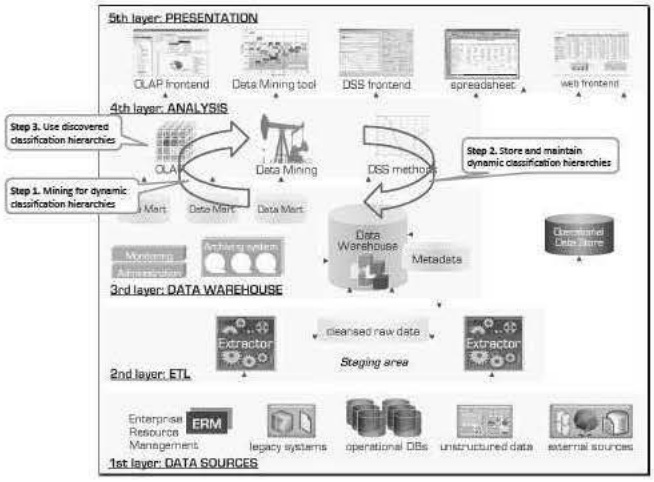

Fig. 1. Integrating a Data Mining Feedback Loop into OLAP Cubes

over the existing cubes and marts of the data warehouse layer as shown in Figure 1. Our proposed contribution is depicted as a 3-step feedback loop between the application and the data warehouse layers in the same figure. In the first step, data mining classification algorithms are applied to cluster dimensional data based on some dynamic characteristics (e.g., to group users by popularity, activity or interest). In the second step, the acquired classification is added as a new aggregation path to the respective dimension, leading to the third step of enabling this new aggregation path in OLAP queries. Introduction of discovered classifications to dimensional hierarchies raises a number of research challenges, such as their maintenance, evolution, temporal validity and aggregation constraints. These issues will be handled later on in this work.

Mining data cubes for dynamic classifications is a popular technique in OLAP applications dealing with customer trending, risk or popularity assessment, etc. However, traditional data mining applications return such classifications as the outcome of the analysis, whereas our approach is to feed this outcome back to the data warehouse as elements of the data model in their own right.

1.1 Tweet Analysis as Motivating Example

Twitter[1] is a popular social network with microblogging service for real-time information exchange. Twitter offers a set of APIs for retrieving the data about its users and their communication. Extreme popularity of the Twitter and the availability of its public stream have resulted in the multiplication of Twitter-related research initiatives as overviewed in the Related Work.

Twitter employs a rather simple data model that encompasses users, their messages (*tweets*), and the relationships between and within those two classes.

[1] `http://twitter.com/`

Users can be friends or followers of other users, be referenced (i.e., tagged) in tweets, be authors of tweets or retweet other users' messages. The third component is the timeline, which describes the evolution, or the ordering, of user and tweet objects. The structure of the original stream explicitly contains a rather small number of attributes usable as measures and dimensions, whereas a wealth of additional parameters, categories and hierarchies can be obtained using different computation methods, from simple derivations to complex techniques of knowledge discovery. Many of the characteristics (e.g., status, activity, interests, popularity, etc.) are dynamic and, therefore, cannot be captured as OLAP dimensions. However, from the analyst's perspective, such characteristics may represent valuable dimensions for the analysis.

The dataset delivered by the Twitter Streaming API is semi-structured using the JavaScript Object Notation (JSON). Each tweet is streamed as an object containing 67 data fields with high degree of heterogeneity. A tweet record encompasses the message itself along with detailed metadata on the user's profile and geographic location. A straightforward mapping of this set of attributes to a multidimensional perspective results in the identification of cubes *Tweet* and *TweetCounters* for storing the contents and the metadata of the messages and for storing the statistical measurements provided with each record, respectively.

1.2 Related Work

The work related to our contribution can be subdivided into two major sections: 1) research on integrating data warehousing and mining and 2) knowledge discovery from Twitter data.

A pioneering work on integrating OLAP with data mining was carried out by Han [5] who proposed a theoretical framework for implementing OLAP mining functions. His *mining then cubing* function is a predecessor of our approach. The idea is to enable application of OLAP operators on the mining results. An example of implementing such functionality can be found in the Microsoft SQL Server and is known as *data mining dimensions* [10]. The latter contain classifications obtained by applying clustering or other algorithms on the original cube and can be materialized and used (with some limitations) just like ordinary dimensions for OLAP. Usman et al. review the research literature on coupling OLAP and data mining in [17] and propose a conceptual model for combining enhanced OLAP with data mining systems. The urge to enhance the analysis by integrating OLAP and data mining was expressed in multiple publications in the past. Significant works in this area include [6], [18], [4], and [3]. It was Han et al. [6] who introduced the concept of integrating OLAP and data mining called Online Analytical Mining (OLAM).

Research contributions related to the Twitter analysis mostly focus on improving the search and navigation in a huge flow of messages as well as on discovering valuable information about the content and the users. We are more interested in the latter types of works. In 2007 Java et al. [8] presented their observations of the microblogging phenomena by studying the topological and geographical properties of Twitter's social network. They came up with a few categories for

Twitter usage, such as daily chatter, information and url sharing or news reporting. Mathioudakis and Koudas [14] proposed a tool called Twitter Monitor for detecting trends from Twitter streams in real-time by identifying emerging topics and bursty keywords. Recommendation systems for Twitter messages are presented by Chen et al. [2] and Phelan et al. [16]. Chen et al. studied content recommendation on Twitter to better direct user attention. Phelan et al. also considered RSS feeds as another source for information extraction to discover Twitter messages best matching the user's needs. Michelson and Macskassy [15] discover main topics of interest of Twitter users from the entities mentioned in their tweets. Hecht et al. [7] analyze unstructured information in the user profile's location field for location-based user categorization. While most Twitter-related contributions focus on mining or enhancing the contents of tweets, improving the frontend or generating meaningful recommendations, we exploit the advantages of coupling the OLAP technology with data mining to enable aggregation-centric analysis of the Twitter data.

2 Conceptual Modeling of Dynamic Elements

Data in a data warehouse is structured according to the aggregation-centric multidimensional data model, which uses numeric measures as its analysis objects [1]. A fact consists of one or multiple measures along with their descriptive properties referred to as *dimensions*. Values in a dimension can be structured into a *hierarchy* of granularity levels to enable drill-down and rollup operations.

The terms *fact* and *measure* are often used as synonyms in the data warehouse context. We distinguish between those terms to account for facts without measures. According to Kimball [9], a fact is given by a many-to-many relationship between a set of attributes. There exist many-to-many mappings in which no attribute qualifies as a measure. A classical example is an event record, where an event is given by a combination of simultaneously occurring dimensional characteristics. We use the notion *non-measurable fact type* introduced in [12] for facts with no measures. Back to the Twitter scenario, a non-measurable fact type could be used to capture the tweeting events with user, message and time/date as its dimensions.

A dimension is a *one-to-many* characteristic of a fact and can be of arbitrary complexity, from a single data field to a collection of related attributes, from uniform grain to a hierarchical structure with multiple alternative and parallel hierarchies [11,13]. OLAP does not support definition of dynamic, non-strict, or fuzzy dimension hierarchies. However, the extended Dimensional Fact Model (x-DFM) [12] makes provisions for modeling such hierarchy types at the conceptual level. We adopt the x-DFM notation for the concepts introduced in this section.

Figure 2 shows an example of modeling a cube for storing user activity statistics in x-DFM. A fact type is represented as a graph centered at the fact type node (*TweetCount*), which includes the measures (*#friends*, *#followers*, *#status*, *#favorited* and *#listed*) and a degenerated (i.e., consisting of a single data field) dimension (*FactID*). Dimensions are modeled as outgoing paths of the fact

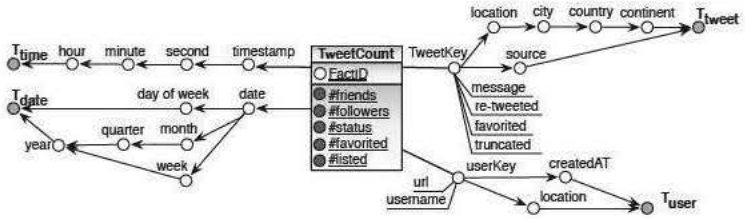

Fig. 2. Fragment of the tweet record in x-DFM

type node with edges as "rolls-up-to" relationships between hierarchy levels. Multiple aggregation paths are possible within a dimension, all converging in an abstract $\top$ node, which corresponds to the aggregated value *all*. A level node in a dimension consists of at least one key attribute, but may include further attributes shown as underlined terminal nodes.

In general, a datacube can be extended by adding new elements of type a) *measure*, b) *dimension*, or c) *hierarchy level*. Adding a new element can be rather trivial if its value is derived from the values of other elements within the same fact entry. We are interested in discovering non-trivial and hidden relationships in the dataset such as those that cannot be expressed by a derivation formula. Our approach is to apply data mining algorithms for discovering clusters or rules useful for defining new elements in the cube. For this purpose, the input set has to be transformed into a representation more generic than the one offered by the multidimensional model. The goal is to treat all elements symmetrically as potential input fields for discovering new categories. To achieve this, we "homogenize" the graph model of the cube and get rid of different types of nodes and edges based on the observation that all edges are of type "many-to-one" or "one-to-one" and all nodes are of type attribute.

Figure 3 (a) shows the transformed graph from Figure 2, describing the cube in terms of attributes and hierarchical relationships between them. The new graph is centered at the fact identifier attribute *FactID*, which uniquely identifies each fact entry (this may be an artificially generated attribute). The obtained representation of a data cube is suitable for specifying the input set for data mining algorithms by selecting a relevant subgraph and extracting the data behind it.

Consider an example of adding a dynamic category *re-tweet activity* to the *user* dimension defined as the frequency of re-tweeting relative to the period ellapsed since the creation of the user's account. This category should assign each user into one of four clusters: *mature-active*, *new-active*, *mature-passive*, and *new-passive* for users who registered long ago or recently and who re-tweet more or less frequently, respectively. Neither the time elapsed since the user registration nor the frequency of re-tweeting is explicit in the data set, but both are derivable from other data fields.

Figures 3 (b), (c), and (d) demonstrate the steps of obtaining the new aggregation path. Figure 3 (b) shows the subgraph relevant for discovering the desired category. Figure 3 (c) shows the derivation of the required fields *time elapsed*

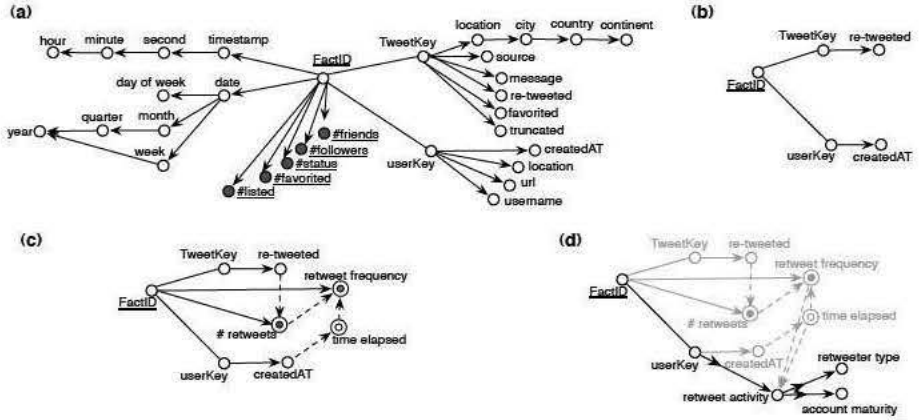

Fig. 3. Stages of acquiring new hierarchy levels

as the difference between the current and the account creation date, cumulative measure *# retweets* as the number of messages with the *re-tweeted* value set to true, and, finally, *retweet frequency* as *# retweets* divided by *time elapsed*. Figure 3 (d) shows the result of adding *re-tweet activity* as a hierarchy level to the *user* dimension. In the conceptual model, a discovered or derived category can be treated just as an ordinary one. For instance, we added parallel hierarchy levels *retweeter type* with member values *active* and *passive* and *account maturity* with member values *new* and *mature* on top of *re-tweet activity*. So far we have considered the presentation of discovering new structural elements at the the conceptual level in order to provide an abstract, generic and implementation-independent view on the data. However, there are significant differences in the behavior of static and dynamic elements in terms of their maintenance and usage in OLAP queries, as elaborated in the next section.

3 Maintenance Strategies for Dynamic Categories

Classically, dimensions in a data cube correspond to non-volatile characteristics of the data. This property ensures consistency and validity of pre-aggregation. In reality, however, the instance or even the structure of a dimension may evolve in time. The problem of *Slowly Changing Dimensions* (SCD)[9] is well elaborated in the literature, with various strategies proposed for maintaining the up-to-date or the historical view, or even both. More sophisticated strategies employ some kind of multiversioning to preserve various states of the aggregates. Dynamic dimensions proposed in our work may be considered a special case of SCD, in which the changes occur in a predictable fashion: discovered categories reflect a particular state of the cube and as such, have to re-computed on a regular or ad-hoc basis to stay consistent with the evolution of the underlying data set. Preservation of all previous states of a dynamic dimension appears crucial for correct aggregation. With this scheduled update behavior, the SCD methodology

Type 4 [9] appears an appropriate implementation option. This method offers unlimited history preservation by creating multiple records for a given natural key and storing the temporal validity bounds for each entry in history tables.

Another challenge is the recomputation of the dynamic category itself. Frequent and complete recomputation may impose an unaffordable burden on the system. A performance gain can be achieved by re-using the outcome rules of the data mining routines used for discovering the category. In our example, we could use the previously established threshold values for *account maturity* and *retweet frequency* for refreshing the assignment if *user* entries to *re-tweet activity*. This way, the data does not need to be mined repeatedly and the maintenance is reduced to simple computations and adjustments within the existing clusters.

A problem specific only to discovered categories is how to assign new member values in a dimension to the parent values of such a category. Depending on the definition of the discovered relationship, either a default assignment should be provided (for example, newly registered users are most likely to fit into *new and passive* cluster of re-tweet activity), or, if the rules of the dynamic assignment are available, these can be applied for assigning the new values.

Finally, there is a problem of quering the data along dynamic categories in the presence of its multiple versions of a dimension hierarchy. In our scenario, it is important to ensure correct analysis by matching the timeframes of the queried facts and those of the applied dimension hierarchies. For example, if we analyse user activity patterns in 2010 by applying the re-tweet activity hierarchy computed in 2012, we will obviously end up with historically incorrect aggregate values. A consistent result can be achieved by the matching the temporal characteristic of each fact entry with the matching version of the dynamic dimension hierarchy. The SCD implementation of Type 2 offers exactly this type of matching for ensuring historically correct aggregation.

4 Demonstration

We implemented the data warehouse for Twitter analysis using the Microsoft SQL Server system with its powerful set of analysis services including OLAP and data mining. We see a big gain in the ability to employ the existing DW technology and tools for enabling discovered dimensions. The dataset for the experiments was obtained via the Twitter Streaming API, which provides 10% of the total public stream of Twitter. We proceed by presenting two cases of discovering new categories in the process of analyzing events on Twitter.

Case 1 - Spatio-temporal analysis of tweeting during the Super Bowl 2012[2]. 2012's Super Bowl XLVI has been of much interest to many, not only sports fans but also to the social network analysts, as it was the top tweeting event to date, with its record value of 12,233 tweets per second. Tweets relevant to this event and with time-bounds of the game were extracted. One task was to find the top (i.e., with the highest number of tweets sent) tweeting cities in the

[2] The Super Bowl is the annual championship game of the National Football League (NFL), the highest level of professional American football in the United States.

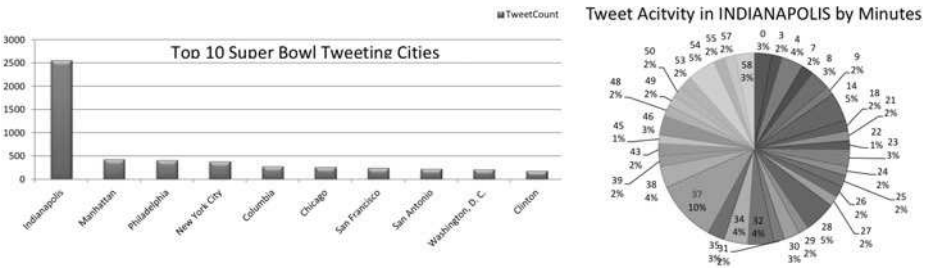

Fig. 4. Twitter Activity during the Superbowl 2012

US during Super Bowl championship. For this purpose, the *FactCount* cube was extended by the measure *TweetCount* and a hierarchical dimension *geolocation*. The input facts were filtered to the tweets originating from the *USA*.

The tweet activity of top 10 cities is plotted in the chart on the left hand side of Figure 4. *Indianapolis*, the city that hosted Super Bowl 2012 championship, remained the most active city during this game with 2559 tweets. The game venue has capacity for 70,000 spectators, which is a contributing factor to make *Indianapolis* the top tweeting city. One other task was to see peak activity along the timeline for the city hosting the championship. The chart on the right hand-side in Figure 4 plots Twitter activity by minutes only for Indianapolis where most tweets were sent in the 37th minute.

Case 2 - Types of Twitter users by geographic regions. Users on Twitter engage in many activities including 1) posting tweets 2) (un-)marking tweets as favorite 3) making other users friend, and 4) (un-)following other users. Our task was to explore geographical regions based on such activities. Figure 5 depicts the outcome of this analysis.

The first pie-chart shows the distribution of Activity (tweeting / status updates) by continent. South America with 37% share is the top active continent followed by North America with 26%. Please note that users on Twitter can exclude location specific data from the tweet. BLANK represents such tweets in the chart. The second pie-chart plots regions by favoriting activity. North Americans lead the way with 32% share and are followed by Asians with 26%. The third pie-chart shows regions by friendship. North American have most friends with 34% share. The last chart shows regions by number of followers with South America having 40% of the total and North America having about 26% of the total number followers, respectively. The mining structure consisted of fields *UserID, User-Created-At, Language* from *UserDIM* dimension and all the measures in the fact table. The mining model, however, contains *User-Created-At*(a Date field) and *StatusCount*. Microsoft Clustering Algorithm was configured to use scalable K-Means method and to have 4 clusters as to correspond to *Active & New*, *Passive & New*, *Active & Mature* and *Passive & Mature* categories.

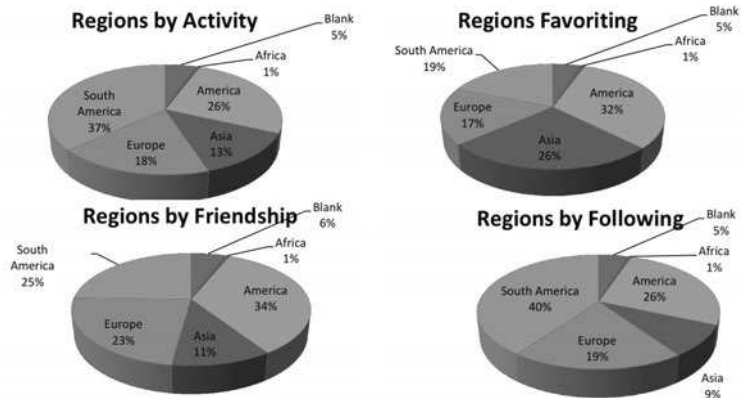

Fig. 5. Exploration of geographical regions by user activity

The presented cases demonstrate the advantages of coupling OLAP with data mining for discovering and analyzing dynamic data characteristics. Re-using the mining results as aggregation paths in OLAP queries enables new insights, which could not be obtained without the feedback loop at the level of data modeling.

5 Conclusions and Future Work

In this work we proposed to extend the classical approach to modeling OLAP dimensions by the inclusion of dynamic categories and hierarchies discovered from the data through the application of data mining algorithms and other computations. The discovered classifications reflect "hidden" relationships in the data set and thus represent new axes for exploring the cube's measures. We handled the process of adding discovered categories at the conceptual modeling level by transforming the cube schema into a homogeneous graph consisting of attribute nodes and hierarchical relationships between them. This representation allowed us to treat measures and dimensions symmetrically for the purpose of discovering interesting relationships and grouping options.

We tested our approach on the dataset of the Twitter's public stream focusing the analysis on the metadata represented by over 60 data fields about the message and its author. The presented application scenarios demonstrate how the original data can be enriched by discovered knowledge about the dynamic characteristics of the data set, such as activity and popularity of the users, Twitter usage patterns by geographical distribution, emergence and dissemination of events, etc. In contrast to the standard application of data mining tools where the outcome is used as the final result, we provide a feedback loop to integrate the obtained groupings into the data cube as additional aggregation paths. We expect our approach to enhancing multidimensional cubes with dynamic hierarchy paths to be a valuable contribution for numerous OLAP applications.

References

1. Chaudhuri, S., Dayal, U., Ganti, V.: Database technology for decision support systems. Computer 34(12), 48–55 (2001)
2. Chen, J., Nairn, R., Nelson, L., Bernstein, M.S., Chi, E.H.: Short and tweet: experiments on recommending content from information streams. In: Proc. CHI, pp. 1185–1194. ACM (2010)
3. Dehne, F., Eavis, T., Rau-Chaplin, A.: Coarse Grained Parallel On-Line Analytical Processing (OLAP) for Data Mining. In: Alexandrov, V.N., Dongarra, J., Juliano, B.A., Renner, R.S., Tan, C.J.K. (eds.) ICCS 2001. LNCS, vol. 2074, pp. 589–598. Springer, Heidelberg (2001)
4. Dzeroski, S., Hristovski, D., Peterlin, B.: Using data mining and OLAP to discover patterns in a database of patients with y-chromosome deletions. In: Proceedings of the AMIA Symposium, p. 215. American Medical Informatics Association (2000)
5. Han, J.: OLAP mining: An integration of OLAP with data mining. In: Proc. of the 7th IFIP 2.6 Working Conf. on Database Semantics, DS-7 (1997)
6. Han, J., Chee, S., Chiang, J.: Issues for on-line analytical mining of data warehouses. In: Proc. of the Workshop on Research Issues on Data Mining and Knowledge Discovery, Seattle, Washington, pp. 2:1–2:5 (1998)
7. Hecht, B., Hong, L., Suh, B., Chi, E.H.: Tweets from justin bieber's heart: the dynamics of the location field in user profiles. In: Proc. CHI, pp. 237–246 (2011)
8. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis, pp. 56–65. ACM (2007)
9. Kimball, R.: The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses. John Wiley & Sons, Inc., New York (1996)
10. MacLennan, J., Tang, Z., Crivat, B.: Mining OLAP Cubes, ch. 13, pp. 429–431. Wiley Publishing (2008)
11. Malinowski, E., Zimányi, E.: Hierarchies in a multidimensional model: From conceptual modeling to logical representation. Data & Knowledge Engineering 59(2), 348–377 (2006)
12. Mansmann, S.: Extending the OLAP Technology to Handle Non-Conventional and Complex Data. PhD thesis, Konstanz, Germany (2008)
13. Mansmann, S., Scholl, M.H.: Empowering the OLAP Technology to Support Complex Dimension Hierarchies. International Journal of Data Warehousing and Mining 3(4), 31–50 (2007) (Invited Paper)
14. Mathioudakis, M., Koudas, N.: Twittermonitor: trend detection over the twitter stream. In: Proceedings of the 2010 International Conference on Management of Data, pp. 1155–1158. ACM (2010)
15. Michelson, M., Macskassy, S.A.: Discovering users' topics of interest on twitter: a first look. In: Proceedings of the Fourth Workshop on Analytics for Noisy Unstructured Text Data, AND 2010 (in Conjunction with CIKM 2010), Toronto, Ontario, Canada. ACM (October 26, 2010)
16. Phelan, O., McCarthy, K., Smyth, B.: Using twitter to recommend real-time topical news. In: Proceedings of the Third ACM Conference on Recommender Systems, pp. 385–388. ACM (2009)
17. Usman, M., Asghar, S., Fong, S.: A conceptual model for combining enhanced OLAP and data mining systems. In: Fifth International Joint Conference on INC, IMS and IDC, NCM 2009, pp. 1958–1963. IEEE (2009)
18. Zhu, H.: On-line analytical mining of association rules. PhD thesis, Simon Fraser University (1998)