Information Processing in EventWeb through Detection and Analysis of Connections between Events

Tariq Mahmood[1]

Faculty of Computer Science,
Institute of Business Administration,
Karachi, Pakistan

Shaukat Wasi[2]

Mohammad Ali Jinnah University,
Department of Computer Science,
Karachi, Pakistan

Khalid Khan[3]

College of Computing and
Information Sciences,
PAF Karachi Institute of Economics
and Technology, Karachi, Pakistan.

Syed Hammad Ahmed[4]

DHA Suffa University, Department of Computer Science,
Karachi, Pakistan

Zubair. A. Shaikh[5]

Mohammad Ali Jinnah University,
Karachi, Pakistan

Abstract—**Information over the Web is rapidly becoming event-centric with the next age of WWW projected to be an EventWeb in which nodes are inter-connected through diverse types of links. These nodes represent events having informational and experiential information and analysis of these events has a substantial semantic impact regarding enhancement of information search, visualization and story link detection. Information regarding semantics of EventWeb connections is also important for event planning and web management tasks. In this paper, we devise and implement an event algebra for detection and analysis of event connections. As compared to traditional solutions, we process both context-match operators and analytical operators, cater for all event information attributes, and define the strength of connections. We implement a tool to evaluate our algebra over events occurring in the academic domain. We demonstrate an almost perfect precision and recall for context-match operators and high precision and recall for analytical operators.**

Keywords—*EventWeb; information processing; event algebra; operators; link detection; link analysis; information analysis; context-match*

I. INTRODUCTION

The web of documents or web of information is now converging towards a web of events, which has been typically labeled as the EventWeb, in which each node represents an event which has both informational as well as experiential data, and which is connected to other nodes through different types of links, i.e., referential, structural, relational, and causal [1]. The information flow over the web is influenced by the experiences of the users instead of the reporting authorities or agencies. Hence, information over the web is now becoming more event-centric (as compared to document-centric) with events forming the crux of EventWeb. Moreover, detection and analysis of links between events, i.e., semantic extraction of EventWeb, is significant to users from two perspectives. Firstly, this information facilitates and enhances information search, information visualization and tasks related to story link

detection. Secondly, semantic information of connections helps users in decisions related to planning, management and prioritization of events. The context of the events formulates these connections. Data such as location of the event, temporal information, event category, and participants of the event, formulates this context. Events occurring at the same place, time, date or having common participants or same category may have some type of inter-relations or inter-connections. These connections have different strengths depending upon the percentage of the context match and the granularity level of the contextual attributes at which the match takes place. A deeper analysis of the context-based connections is a target of current research in order to explore more connections.

The focus of this paper is to enhance state of the art research in extracting semantic information from EventWeb. Our research objective is to process the contextual event information to detect linkages between events, based on the following three objectives: 1) detecting stories that exist along events but are unseen, 2) enhancing information search and visualization experience over the web, and 3) constructing a formal and extensible representation for representing linkages between events.

To this end, we devise and implement an event algebra for representing and analyzing different connections between events over EventWeb. We generate useful information regarding these connections. Our motivation is to provide a more formal specification of information regarding event descriptions and connections as compared to other state of the art algebras. Our algebra caters for all five event information attributes, i.e., *title*, *location*, *temporal attributes*, *participants*, and *category*. It comprises a number of operators, each of which defines a possible connection between two or more events. These operators represent different types of connections and help in analyzing and producing important information (semantic meaning) from connections. Specifically, *context-match* operators provide a match between the individual contextual attributes of events and *analytical* operators provide

an analytical view over the connections (described in Section 3). Our algebra also detects previously unexplored connections and defines the strength of connections that identifies connections at various levels of strength. Collectively, the aforementioned features are not available in previously existing algebras. Finally, our algebra can also be modeled through a relevant ontology or some other modeling technique.

To evaluate our proposed event algebra, we developed a tool called EventWeb Connection Detector, abbreviated EConnDetect, which implements our algebra operators. We apply each operator to the collection of events to detect the connections existing between the events. For evaluation, we focused on events occurring in an academic (university) environment. Our research question is to determine the frequency of connections that EConnDetect is able to identify correctly from a given set of university events. For this, we initially extracted these events from email inboxes of several students and faculty members (with their consent). Using our previous technologies, we then extracted event information attributes by using finite state machines and then used an event classifier to tag the events with proper categories [1, 2]. We then provided these event attributes as input to EConnDetect, and calculated precision and recall for the identified connections as our evaluation metrics. We obtained an almost perfect precision and recall of 99% and 97% with context match operators, and a high precision and recall of 89% and 78% for analytical operators.

II. RELATED WORK

Detecting linkages between events has been addressed by researchers mainly from two perspectives: linking events on the basis of event information attributes (location, date/time, type etc.) [3, 4, 5, 6, 7, 8, 9, 10, 11, 12] and linking events on the basis of information related to events (pictures, news, posts etc.) [13, 14, 15, 16, 17, 18, 19]. For our literature review, we have considered the works that address linkage detection based on event attributes.

To this end, we have classified the existing research broadly into two classes: ontology-based event representation and algebra-based event representation. Ontology-based solutions [3, 4, 5, 6] are generally aimed at modeling events so that the connections between events can be traced easily. The algebra-based solutions [7, 8, 9, 10, 11, 12] are largely focused on defining operators that represent possible linkages between events. In this paper, we are concerned with algebra-based solutions to linkage detection.

The gap analysis over the existing research in this domain is shown in Table 1, which lists the features or characteristics of our proposed algebra, and mentions the status of existing research works with respect to these features or characteristics. Specifically, we indicate the extent to which the research literature addresses these features through three labels: Addressed (A), Partially Addressed (PA) and Not Addressed (NA). In general, most work in event link detection focuses on historical event analysis, which addresses the problem of detecting and analyzing links between events appearing in near future, along with the events that occurred in past. Also, most works in this area have targeted

events appearing over news wires or articles etc. In our work, we target events appearing over social text streams and WWW in general.

TABLE I. GAP ANALYSIS: A=ADDRESSED, PA=PARTIALLY ADDRESSED, NA=NOT ADDRESSED

Research Work	Features			
	Linkage Generalization	*Linkage Analysis*	*Identifying Inter-relations of Linkages*	*Linkage Strength*
[3]	A	PA	NA	NA
[4,5]	A	PA	NA	NA
[6]	A	NA	NA	NA
[9]	A	A	NA	NA
[12]	NA	PA	NA	NA
[11]	NA	PA	PA	NA
[7,8]	NA	PA	PA	NA
[10]	NA	PA	NA	NA

A. Ontology-based Approaches

In [3], the authors present a Simple Event Model (SEM) for historical event analysis. They use graphs for event representation. The graphs capture four core event attributes (*title*, *place*, *time*, *actors*) along with several properties that help to identify the linkages between events. The core linkages identified through SEM are determined by the level of similarity with the core attributes. SEM also links the events based on event type. It identifies linkages using types of actors, places and events. However, the authors have not identified inter-relationships between linkages that would have helped them to generate more linkages. Moreover, they do not identify the strength of linkages that could have assisted in clarifying historical linkages. We, therefore, consider this handling of link analysis as partial.

The works of Ilaria Corda et al. [4, 5] also use event ontology to analyze the historical event collection to unveil connections between events. Their objective is to represent essays describing the history of events. For this, they propose the concept of semantic trajectories which represent sequences of events. In a semantic trajectory, any two events which occur consecutively are linked to each other by some semantic link. This link is characterized by one or more attribute(s) that are common in both events. The authors tackle linkage generalization by using a set of different attributes. Moreover, the semantic links are sequenced in the trajectory based on chronological ordering of occurrence. The authors have dealt with analysis over temporal linkages but have not addressed any other linkage type or the strength of linkages. They have also not addressed the identification of inter-relationships of linkages.

In [6], the authors developed the LODE model for representing events with an objective to perform historical event analysis. It covered the four W's (*when*, *where*, *who*, and *what*) to represent and link events. However, the authors do not attempt to identify more linkages, or the inter-linkages between relationships, or the strength of relationships.

B. *Event Algebra-based Approaches*

Allen presented a seminal work for defining relations between events through an algebra [20]. He defined thirteen temporal relations, i.e., the relations were defined based on temporal attributes of the events. Hence, Allen's algebra only addressed one attribute of the event for representing linkages. Moreover, the event algebra by Chakravarthy et al. was developed for detecting composite events in active databases [11]. This algebra contained operators that could be applied over events to detect links between primitive events. Here, the authors identified only temporal linkages and do not address the identification of linkage strength.

Nagargadde and Sridhar developed an algebra to identify links between events in the sport of Cricket [9]. Their objective was to link basic events to formulate a "meta" or derived event. The authors used three attributes, i.e., *time*, *space* and *label* to detect linkages between events. This work focuses on meta events like "Run Out" and "LBW" which refer to basic events like "ball hits the wicket" and "ball hits the player's leg" respectively. . Hence, the authors use temporal and spatial attributes at a low granularity level. They perform linkage analysis to extract meta events; however inter-relations between linkages and the linkage strength have not been explored.

The EVA algebra detects composite events related to a specific domain [7, 8]. At any time, an event is defined to be a transition in state of an object. Composite events are identified by linking the transition in a specific state of an object at that time. This algebra comprises sequence operators and operators that link primitive events on the basis of time and the attribute over which the transition has occurred. Hence, this algebra is focused over temporal attributes. The authors perform analysis to identify sequences of events and to further identify inter-relations among temporal linkages. However, the authors do not cater for the strength of the linkages.

Uma and Aghila have proposed operators for identifying temporal patterns of events [12]. They have extended Allen's algebra [20] by using an event as a reference for relation between two other events. The events are linked based on the temporal attributes only. The authors have modeled the temporal info using hand-coded rules for identifying the linkages and have suggested that these rules can be fed to ontology for detecting event linkages. Finally, Rink et al. use textual graph patterns for detecting casual relationships between events [10]. These patterns facilitate analysis over the relationships but do not cater for inter-relationships or the strength of linkages.

A. *Major Contributions Compared to Related Work*

As compared to the related work, our algebra has the following major contributions:

- Our algebra uses five attributes for linking events which haven't been used collectively in any work, i.e., title, location, temporal attributes, participants and category

- Our algebra captures all possible connections that may exist between two events, while each related work identifies only a specific type of linkage, e.g., composition, dependency, or temporal sequence

- Our algebra contains operators that provide an *analytical* view over linkages between events; this feature is not offered by any related work

- Our algebra defines the criteria for identifying the strength of event connections and provides definitions of the operators for various strength levels. Both of these features haven't been proposed or implemented in any related work.

III. EVENT ALGEBRA FOR DETECTION AND ANALYSIS OF CONNECTIONS

In this section, we detail our proposed algebra for detecting and analyzing connections between events. As mentioned in Section 1, our algebra has two types of operators, i.e., *context match* operators and *analytical* operators. In Table 2, we give the complete list of these operators and describe them later on in this section. These operators identify inter-event connections and also provide an analytical view over these connections. This view aids in determining the strengths of connections and the prioritization of events.

The analytical operators are composite operators and are defined with two or more simple (context-match) operators. For example, *co-location*, *homology*, *analogy*, *concurrency* and *title-alike* are simple operators that check for matches over location, participants, event category, time/date and title respectively. *Duplication* is a composite operator that comes into a "true" state if the events are collocated, homologous, analogous, concurrent, and have the same title. Figure 1 depicts the composition of all analytic operators in our algebra. Here, composite (analytical) operators are shown in the column on the left, and the simple operators defining these composite operators are shown in the right column. The arrows represent the "definition" relationship, e.g., M-Participation is defined by Co-location, Analogy and Participation. An analytic operator can be used to produce event recommendations for a user or to provide the prioritization aspect. For example, consider that two events *E1* and *E2* are sub-events of an event *E* and a user's (*U1*) previous event participation history shows that she always attends events similar to *E1*. Now, if *U1* has another event *E3* that is overlapping with event *E2*, a recommendation system can recommend *U1* to attend *E2* instead of *E3* in order to avoid missing *E1*. Similarly, suppose that *U1*'s participation history shows she mostly attends events in which another user *U2* is also present. Now for *U1*, all the events in which *U2* is participating are of high priority.

TABLE II. EVENT ALGEBRA OPERATORS

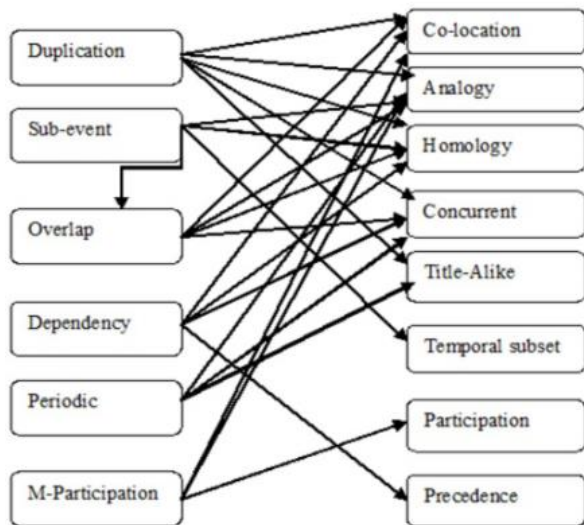| Operator Name	Symbol	Operator Type	Description	Operator's effect
Concurrency	<<>>¹	Simple	The time interval or point is same	Context Match
Precedence	<	Simple	One after another	Context Match
Temporal Subset	Ξ	Simple	[Start time(E1)>start time(E2) and (End time (E1)<End time (E2)	Context Match
Analogy	<<>>ᵉ	Simple	Same Type or category	Context Match
Co-location	<<>>ˢ	Simple	Same Place	Context Match
Homology	<<>>ᵖ	Simple	Same participants	Context Match
Title-Alike	<<>>¹	Simple	Same title	Context Match
Participation	∋	Simple	A person participating in an event	Context Match
Duplication	≡	Composite	All attributes are same	Context Match
Overlap	∞	Composite	Time interval overlaps	Analytical
Dependency	⇒	Composite	Dependent over an event	Analytical
Sub Event	∋	Composite	Part of mega event	Analytical
Periodic	⊂	Composite	Repeated after specific intervals	Analytical
M-Participant	∋ᴹ	Composite	Participation in two or more related events	Analytical

Fig. 1. Composition of Analytical Operators.

Since the operators represent various connections, therefore the strength of an operator is actually the strength of the connection that is represented by the specific operator. As discussed above, the granularity level of the contextual attributes helps in identifying the strength of the operators. Here, the granularity level means the level of detail (or depth) for an attribute. For example, in case of location, we assume that *city*, *area* in city and the (exact) *spot* may be available; however the attributes *country*, *region*, and *continent* may also be taken into account. In our algebra, we have considered five levels of granularity for any attribute, but some attributes may have more depth. The change in granularity will not affect the algebra or the analysis process and hence, any level of granularity can be used for any attribute.

We assume an event title to have at most three words, i.e., a granularity level of 3. We will perform the match for these words and the strength of the match will depend upon the level of granularity at which the match is achieved. For location, we have defined three levels of granularity as mentioned above, i.e., *city*, *area* and *spot*. In case of temporal attributes, since the events are real-world events and not real-time events, therefore we will only provide time in *hours* and *minutes*. For describing the date of an event, we have used *year*, *month* and *day of the month*. Yet, our own event extraction component extracts dates from various types of phrases and converts them to a canonicalized format. While producing the definitions of the connection operators and defining the strength criteria, we have ignored those cases in which the connection cannot exist or may occur rarely.

For applying the algebra operators, we have devised a 5-Tuple description of an event. An event E is described as $E = (L; S; T;C; P)$ where L, S, T, C and P represent title, location, temporal attributes, event category and participants respectively. L, S and T are composite elements. L is further composed of three sub-elements and is defined as a 3-tuple $< w1; w2; w3 >$, where $w1$, $w2$ and $w3$ are the labels depicting the granularity levels. For example, in a title "Bubble-up Cricket Tournament"; w1 = "Bubble-up", w2 = "Cricket" and w3 = "Tournament". Similarly, S is defined as a 3-tuple $< s1; s2; s3 >$, where $s1$, $s2$ and $s3$ represent city, area and spot respectively. T is also defined by a 3-tuple $< dts; dt f; ts; t f >$, where dts, $dt f$, ts and $t f$ represent start date, end/finish date, start time and end/finish time of an event respectively. The terms dts and $dt f$ are further composed of the triplet $(y; m; d)$ where y, m and d represent year, month and day respectively. Similarly, ts and $t f$ are defined by a pair $(h; m)$ where h and m represent hour and minutes respectively. We represent the relationship between an event and its attributes with question mark (?), and between an element of the event description and its sub-elements by dot (.). For example, for an event *E1*, the location is represented as $E1?S$ and $S:s1$ represents the city name.

We now move towards a formal description of our operators. Each operator has its own criteria for strength. The symbols for the operators have superscripts and subscripts. The superscript contains the symbol representing the event attribute and the subscript contains the symbols representing the strength of the operator. Attribute symbols, *l*, *s*, *p*, *t*, and *c* stand for *title=label*, *location=site*, *participants*, *time* and *eventtype=category* respectively. For strength, we have used three symbols, , χ and φ, representing the levels of strength of operator in descending order respectively. We need to specify that the notation used for representing our algebra is our own selection. Specifically, we use mathematical symbols to represent the algebra operators. While choosing a symbol to represent an operator, we have tried to select a symbol which, in mathematics, is used to represent a similar relationship. For example, we use ≺ for representing the precedence relationship. In mathematical equations, the same symbol is used to represent precedence. Also, the matching of string operators is done using an equality operator.

A. Event Algebra-based Approaches

We have defined eight context-match or simple operators. These operators provide a match between the individual contextual attributes of events. The definitions provided below for these operators are obvious and clearly represent the semantics of the operators.

1) Analogy: An event E1 is analogous to another event E2 if E1 has same type or category as E2. The strength of the connection is either at the highest level when the types of the events match, or it's null, i.e., there is no analogy connection when the events' types are different. This connection is represented by Equation 1.

$$E1 <<>>c\ E2 \tag{1}$$

2) Homology: An event E1 is homologous to another event E2 if E1 and E2 have same or common participants. The homologous connection has two strength levels. In case all participants are the same in both events, the events are completely homologous. Otherwise, if the participants of one event are a proper subset of the other event then the homology is weak.

$$E1 <<>>^p_\varphi E2\ if(E1 \rightarrow p = E1 \rightarrow p) \tag{2}$$

$$E1 <<>>^p_\chi E2\ if(E1 \rightarrow p \approx E1 \rightarrow p) \tag{3}$$

$$E1 <<>>^p_\varphi E2\ if(E1 \rightarrow p\ E1 \rightarrow p) \tag{4}$$

3) Co-Location: An event E1 is co-located to another event E2 if E1 and E2 occur at the same location. The strength of the connection depends upon the granularity level at which the match takes place. A match at only the top level of granularity means a lower level of strength, and a match at the lowest granularity level means highest strength or an exact match. The co-location connection is represented by Equation 5, Equation 6 and Equation 7, representing high, average and low levels of strength for co-location connection respectively.

$$E1 <<>>s\ E2\ if(E1 \rightarrow S{:}s1 = E1 \rightarrow S{:}s1)\ \wedge$$

$$(E1 \rightarrow S{:}s1 = E1 \rightarrow S{:}s1)\ \wedge$$

$$(E1 \rightarrow S{:}s1 = E1 \rightarrow S{:}s1) \tag{5}$$

$$E1 <<>>s\ \chi\ E2\ if[f(E1 \rightarrow S{:}s1 = E1 \rightarrow S{:}s1)\ \wedge$$

$$(E1 \rightarrow S{:}s3 = E2 \rightarrow S{:}s3)g\ \_\ f(E1 \rightarrow S{:}s1 = E2 \rightarrow S{:}s1)\ \wedge$$

$$(E1 \rightarrow S{:}s2 = E2 \rightarrow S{:}s2)g] \tag{6}$$

$$E1 <<>>s\ \varphi\ E2\ iff(E1 \rightarrow S{:}s1 = E2 \rightarrow S{:}s1)\_(E1 \rightarrow S{:}s2 = E2 \rightarrow S{:}s2)\ \_\ f(E1 \rightarrow S{:}s3 = E2 \rightarrow S{:}s3)g \tag{7}$$

4) Concurrency: The definition of concurrency between two events E1 and E2 is separately given for different granularity levels as we have extended the actual definition of concurrency for our purpose. Specifically, if E1 and E2 occur in the same day, month, and year and the time interval also overlaps, or matches exactly, then the events are concurrent with high strength; this strong concurrency is represented by Equation 8. Equation 9 represents concurrency with average strength. The reason for a lower level of strength is that the time does not match exactly and the duration of one event falls within the duration of other event. The lowest level of concurrency is represented by Equation 10 that captures the case where only year and month or month and day match for two events.

$$E1 <<>>t\ E2\ if\ (E1 \rightarrow T{:}dts = E1 \rightarrow T{:}dt\ f\ )^\wedge$$

$$(E1 \rightarrow T{:}dts = E2 \rightarrow T{:}dts)^\wedge\ [f(E1 \rightarrow T{:}ts = E2 \rightarrow T{:}ts)\ \wedge$$

$$(E1 \rightarrow T{:}t\ f = E2 \rightarrow T{:}t\ f\ )g\_f(E1 \rightarrow T{:}ts < E2 \rightarrow T{:}ts)\ \wedge$$

$$(E1 \rightarrow T{:}t\ f > E2 \rightarrow T{:}t\ f\ )g\_\ f(E1 \rightarrow T{:}ts > E2 \rightarrow T{:}ts)\ \wedge$$

$$(E1 \rightarrow T{:}t\ f < E2 \rightarrow T{:}t\ f\ )g] \tag{8}$$

$$E1 <<>>t\ \chi\ E2\ if(E1 \rightarrow T{:}dts = E2 \rightarrow T{:}dts)\ \wedge$$

$$f(E1 \rightarrow T{:}dt\ f < E2 \rightarrow T{:}dt\ f\ )\ \_\ (E1 \rightarrow T{:}t\ f > E2 \rightarrow T{:}ts)g \tag{9}$$

$$E1 <<>>t\ \varphi\ E2\ if[f(E1 \rightarrow T{:}dts{:}y = E2 \rightarrow T{:}dts{:}y)\ \wedge$$

$$(E1 \rightarrow T{:}dt\ f{:}y = E2 \rightarrow T{:}dt\ f{:}y)g\_f(E1 \rightarrow T{:}dts{:}m = E2 \rightarrow T{:}dts{:}m)\ \wedge\ (E1 \rightarrow T{:}dt\ f{:}m = E2 \rightarrow T{:}dt\ f{:}m)g\_f(E1 \rightarrow T{:}dts{:}d = E2 \rightarrow T{:}dts{:}d)\ \wedge\ (E1 \rightarrow T{:}dt\ f{:}d = E2 \rightarrow T{:}dt\ f{:}d)g] \tag{10}$$

5) Temporal Subset: Two events E1 and E2 are temporal subsets of an event E', if the time intervals of E1 and E2 fall within the time interval of E'. Equation 11 represents the temporal subset connection.

$$E'\Xi_t\ E1\ \text{¡}E2\ if[[(E' \rightarrow T{:}dts = E1 \rightarrow T{:}dt = E2 \rightarrow T{:}dt)\ \wedge f(E' \rightarrow T{:}ts \le min(E1 \rightarrow T{:}dts;\ E1 \rightarrow T{:}dts))g\wedge f(E' \rightarrow T{:}ts \ge max(E1 \rightarrow T{:}dt\ f;\ E2 \rightarrow T{:}dt\ f\ ))g] \tag{11}$$

6) Precedene: Two events E1 and E2 are temporal subsets of an event E', if the time intervals of E1 and E2 fall within the time interval of E'. Equation 11 represents the temporal subset connection.

$$E1 \prec_t E2\ if[f(E1\ !\ T{:}dt\ f \le E2\ !\ T{:}dts)g\ \_\ f(E1\ !\ T{:}dt = E2\ !\ T{:}dt)\ \wedge\ (E1\ !\ T{:}t\ f \le E2\ !\ T{:}ts)g] \tag{12}$$

7) Title-Alike: The titles of two given events E1 and E2 may match exactly or partially. The exact match does not mean that the events are same, as two events with the same name may occur at different locations with different time intervals. Specifically, if the title of E1 and E2 match at all three levels of granularity, i.e., spot, area and city match exactly, then it means that the title of the events match exactly (Equation 13). In all other cases, where the titles match at any granularity level, the events are said to have a Title-Alike connection with a low strength (Equation 14).

$E1 <<>>l \ E2$ iff$(E1 \ ! \ L{:}l1 = E2 \ ! \ L{:}l1)$ ^ $(E1 \ ! \ L{:}l2 = E2 \ !$
$L{:}l2)$g ^ $(E1 \ ! \ L{:}l3 = E2 \ ! \ L{:}l3)$g $\hspace{2cm}$ (13)
$E1 <<>>s$
$\varphi \ E2$ if[f$(E1 \ ! \ L{:}l1 = E2 \ ! \ L{:}l1)$ ^ $(E1 \ ! \ L{:}l2 = E2 \ ! \ L{:}l2)$g_
f$(E1 \ ! \ L{:}l1 = E2 \ ! \ L{:}l1)$ ^ $(E1 \ ! \ L{:}l3 = E2 \ ! \ L{:}l3)$g_
f$(E1 \ ! \ L{:}l3 = E2 \ ! \ L{:}l3)$ ^ $(E1 \ ! \ L{:}l2 = E2 \ ! \ L{:}l2)$g] $\hspace{1cm}$ (14)

8) Participation: The participation operator represents the link between a person and an event. A person P is said to have a connection with an event E, if P is in the participants' list of E. Participation operator is a simple operator and is represented by Equation 15).

$E \ni A$ $\hspace{6cm}$ (15)

B. Analytical Operators

Analytical operators provide analytical perspective to the connections. We have defined six analytical operators. As illustrated in Figure 1, the definitions of these operators depend on context-match operators for providing the analytical view. We formalize these definitions as follows.

1) Duplication: Two events are duplications of each other or related to each other through the duplication operator, if they have the same values for all event attributes. Given two events E1 and E2, the rule for the duplication operator is given by the Equation 16.

$E1 \equiv E2$ iff$(E1 <<>>l \ E2)$ ^ $(E1 <<>>t \ E2)$ ^ $(E1 <<>>p \ E2)$ ^
$(E1 <<>>c \ E2)$ ^ $(E1 <<>>s \ E2)$g $\hspace{2cm}$ (16)

2) Overlap: Two events E1 and E2 are said to overlap if they are homologous, analogous, concurrent and co-located. The strength of the connection varies with respect to various combinations and strengths of context match operators. Equation 17, Equation 18 and Equation 19 describe these dynamics of the overlap operator.

$E1$ on $E2$ if[$(E1 <<>>c \ E2)$ ^ f$(E1 <<>>p \ E2)$ _ $(E1 <<>>\chi p$
$E2)$g^
f$(E1 <<>>t \ E2)$ _ $(E1 <<>>t\chi \ E2)$g ^ $(E1 <<>>s \ E2)$] $\hspace{0.5cm}$ (17)
$E1$ on
$\chi \ E2$ if[f$(E1 <<>>\varphi p \ E2)$ _ $(E1 <<>>\chi p \ E2)$g^
f$(E1 <<>>t \ E2)$ _ $(E1 <<>>t\chi \ E2)$g^
f$(E1 <<>>s \ E2)$ _ $(E1 <<>>\chi s \ E2)$g] $\hspace{1.5cm}$ (18)
$E1$ on $E2$ if[$(E1 <<>>c \ E2)$ ^ f$(E1 <<>>\varphi p \ E2)$ _ $(E1 <<>>\chi p$
$E2)$g^
f$(E1 <<>>t \ E2)$ _ $(E1 <<>>t\chi \ E2)$g ^ $(E1 <<>>\varphi s \ E2)$] $\hspace{0.5cm}$ (19)

3) Dependency: A dependency between two events E1 and E2 may exist due to various reasons. The reason for occurrence of this dependency relates to its strength. The dependency operator checks for the existence of precedence, homology and co-location connections and based on their existence (or non-existence), defines its own existence. It also considers the strengths of its three simple operators. If precedence, homology, and co-location connections have high strengths, then dependency also exists with high strength. If the events are not co-located but have high precedence, and are homologous with an average strength, then the dependency exists with an average strength. The same scheme is applied for low dependency with a homology with low strength. Equation 20, Equation 21 and Equation 22 represent these dynamics of the dependency operator.

$E1 \ ) \ E2$ iff$(E1 \prec t \ E2)$ ^ $(E1 <<>>s \ E2)$ ^ $(E1 <<>>p \ E2)$g
$\hspace{5cm}$ (20)
$E1 \ )$
$\chi \ E2$ iff$(E1 \prec t \ E2)$ ^ $(E1 <<>>\chi p \ E2)$g $\hspace{2cm}$ (21)
$E1 \ )$
$\varphi \ E2$ iff$(E1 \prec t \ E2)$ ^ $(E1 <<>>p \ E2)$g $\hspace{2.5cm}$ (22)

4) Sub-Event: Two events E1 and E2 are said to be sub-events of a mega-event EM under 3 conditions: i) E1 and E2 have common participants such that the union of both sets of participants equals the participant set of the mega-event, ii) E1 and E2 are of the same type as the mega-event, and iii) the time intervals of E1 and E2 fall within the time interval of the mega-event. Equation 23 mathematically describes these definitions:

$E_M \ni E1jE2$ if[$(E1 \ \text{on}_c \ \varphi \ E2)$ ^ $(E_M <<>>c \ E1)$ ^

f$E_M <<>>p \ (E1 \ [\ E2)$g ^ $(E_M \Xi_t \ E1jE2)$] $\hspace{1.5cm}$ (23)

5) Periodic: If two events E1 and E2 have a similar title, similar type, and similar dates then the events are strong candidates for being incidences of a same event E+ that occurs periodically. We represent this mathematically in Equation 24.

$E_+ @ E1jE2$ iff$(E1 <<>>l \ E2)$ ^ $(E1 <<>>c \ E2)$ ^ $(E1 <<>>t\varphi$
$E2)$g $\hspace{5cm}$ (24)

6) M-Participation: If a person P is participating in two events E1 and E2 and the events have a connection between them, then P is said to have M-Participation with E1 and E2. The strength of M-Participation depends upon the strength of the eventevent connection. If E1 and E2 have the same type, or are held at the same location, then the connection between the events and the participant is good, but if E1 and E2 have the same type and the same location, then the connection has a higher strength. These dynamics are represented in Equation 25 and Equation 26.

f$E1; E2$g $\ni \chi M \ P$ if[$(E1 \ni P)$ ^ $(E2 \ni P)$ ^ f$(E1 <<>>s \ E2)$ _
$(E1 <<>>c \ E2)$g] $\hspace{4cm}$ (25)

f$E1; E2$g $\ni M \ P$ if[$(E1 \ni P)$ ^ $(E2 \ni P)$ ^ f$(E1 <<>>s \ E2)$ _ $(E1$
$<<>>c \ E2)$g] $\hspace{4.5cm}$ (26)

IV. Description of Econndetect

As mentioned in Section 1, we have implemented a Java-based tool called EConnDetect which inputs a set of events that are provided through a text file, and outputs event connections along with the strength of connections. The tool's GUI offers two primary functions, i.e., analysis and viewing of events. The analysis function detects connections in input data, and outputs connection details in an interactive tabular form. Figure 2 shows an EConnDetect snapshot of the collection of events used in our evaluation (described in next section). Here, column names represent event attributes, e.g., title, location, participants etc. Also, Figure 3 presents a snapshot of the output produced by EConnDetect. Here, the output comprises the ID of the linkage, IDs of the two linked events, the linkage found between the events and the strength of the identified linkage. All information shown in Figure 2 and Figure 3 is self-explanatory and hence we do not describe it in detail.

Fig. 2. Collection of Events used for Evaluation of Proposed Event Algebra.

Fig. 3. A Snapshot of Connections found by EconnDetect.

V. Algebra Evaluation with Econndetect

We evaluated EConnDetect over events occurring in the academic (university) domain. Our data set of university events contains professional and social events; however some events also fall in the category of personal events. This work accomplishes the development of one of the components of our context-based event detection model. Therefore, we have sampled the event email data set constructed for our two previously developed components [21, 2]. Our motivation for sampling the university data set is:

- The email data set contains an extensive variety of event related communications.

- Three general classifications, i.e., Personal, Professional and Social, categorize all types of events in the email data set (social, educational, formal, personal, friendship, family and official events).

- The email data set provides a diverse user base.

We identified two types of users, i.e., Type A (Students) and Type B (Faculty). We categorized the interaction between these users into four types:

- T ype-I: A to A (Personal/Social interaction from Students to Students)

- T ype-II: A to B (Professional/Social interaction from Students to Faculty)

- Type-III: B to A (Personal/Professional/Social interaction from Faculty to Students)

- T ype-IV: B to B (Personal/Professional/Social from Faculty to Faculty).

The selection of our dataset towards the university email corpus itself posed some interesting problems such as the categorization of the variety of event types, spanning from one-one lunch invitations to faculty meetings, office meetings, discussions with the supervisor, wedding invitations, educational and social seminars, conferences, picnic, and sport events etc. that were covered in our dataset [2].

We sampled approximately 100 university events for evaluation of our proposed algebra. This event data was extracted from email inboxes of six students and four faculty members with their consent. The emails were collected in a time span of about six months. The data was extracted using an Event Information Extraction System (EIES) that uses part of speech (POS) based Finite State Machines (FSM) for extraction of event information (title, location, temporal information and participants) from emails [21]. We then used our event email classifier to label the extracted events with an event category (Social, Professional, or Personal) [2]. Finally, the collected contextual event information is stored in CSV format, and fed as input to EConnDetect.

EConnDetect applies our algebra operators to input data. For each operator, we have developed a simple function. Each function outputs a Boolean value that depicts the existence or non-existence of connection, and strength value that depicts the strength of the relationship, in case the connection exists. The function L represents the relationship between the operator, event pairs and the output obtained as a result of the application of the operator over the event. The domain and range for L are given by Equation 27.

$$L : OS \times ES \rightarrow OS \times B \times S \, t \tag{27}$$

When an operator *o* 2 *OS* is applied to events (*E*1; *E*2) 2 *ES* ,the output is a triplet (*o*; *bool*; *str*), where *o* 2 *OS* , *bool* takes values from (*true*; *f alse*) and *str* represents the strength (; χ; φ) of the connection. The algorithm representing the process of identifying connections is provided in Algorithm 5.1 and sample pseudo codes for colocation and dependency operators are provided in Algorithm 5.2 and Algorithm 5.3 respectively2. These pseudo codes are implementations of equations that represent the connection operators. For detecting the existence of a connection between two events, the events are fed as input to the operator's function. The conditions, represented by the operator's equation, that indicate the existence of the connection are applied over the event attributes. The functions return Boolean values to indicate the existence or non-existence of a particular connection. If a connection is found to exist between two events, then the strength of the connection is also returned by the function.

Algorithm 5.1: *ConnectionDetection(ES; OS)*
for each *Oᵢ* 2 *OS* **comment:** Simple Operators are applied first followed by Composite ones
do{
for each *Eⱼ* 2 ES j Flag(*Eⱼ*) , processed
     do{
     for each *Eₖ* 2 ES ; j , k
         do{
            Apply *Oᵢ* to (*Eⱼ*; *Eₖ*)
         }
         Flag(*Eⱼ*) = processed
         comment: An event is marked as it is processed for link detection } }

Algorithm 5.2: *Co-Location(E1; E2)*

comment: {*E*1; *E*2} 2 *ES*
if {(*E*1 → *S*:*s*1 = *E*2 → *S*:*s*1) ^
(*E*1 → *S*:*s*2 = *E*2 → *S*:*s*2) ^ (*E*1 → *S*:*s*3 = *E*2 → *S*:*s*3)}
then return (1;*Co - Location*; *High*)
else if {[{({(*EE*11→→*SS*:*s*:*s*11==*EE*22→→*SS*:*s*:*s*1) 1)^^((*EE*11→→*SS*:*s*:*s*23==*EE*22→→*SS*:*s*:*s*2) 3)}]}_
then return (1;*Co - Location*; *Average*)
else if {(*E*1 → *S*:*s*1 = *E*2 → *S*:*s*1) _ (*E*1 → *S*:*s*2 = *E*2 → *S*:*s*2) _ (*E*1 → *S*:*s*3 = *E*2 → *S*:*s*3)}
then return (1;*Co - Location*; *Low*)
else return (0;*Co - Location*)

Algorithm 5.3: *Dependency(E1; E2)*
comment: {E1; E2} € ES
if{
[({(*Co precedence - Location* (*E*1(;*EE*12) ; *E*=2)
(1 =;(1 *Precedence* ;*Co - Location* ; *High*;)) *High* ^)) _
(*Homology*(*E*1; *E*2) = (1; *Homology*; *High*))}]
}
then return (1; *Dependency*; *High*)
else if {{(*precedence*(*E*1; *E*2) = (1; *Precedence*; *High*)) ^

(*Homology*(*E*1; *E*2) = (1; *Homology*; *Average*))}}
then return (1; *Dependency*; *Average*)
else if {{{(*precedence*(*E*1; *E*2) = (1; *Precedence*; *High*)) ^
(*Homology*(*E*1; *E*2) = (1; *Homology*; *Low*))}}
then return (1; *Dependency*; *Low*)
else return (0; *Dependency*)

The co-location operator, provided in Algorithm 5.2, matches the location attributes of the two events that are fed as input. The location attributes are matched at each granularity level. As defined in a previous section, we have assumed 3 levels of granularity for the location attribute and therefore the location attribute S comprises of a 3-tuple < *s*1; *s*2; *s*3 >. Hence *s*1, *s*2 and *s*3 for *E*1 are matched with *s*1, *s*2 and *s*3 for *E*2. If any of the three tuple variables is matched, the function returns 1, indicating the existence of co-location connection. The strength of the connection is determined based on number of tuple variables that match for both events. For a complete and single match, the strength is designated as high and low respectively, and for a two-tuple variable match, the strength is designated as average.

The dependency operator in Algorithm 5.3 is a composite operator and it requires the output of three other operators; precedence, homology and co-location. The precedence operator indicates that ending of one event before or along with the beginning of another event. Homology and co-location operators define same participants and same location respectively. The dependency operator checks for the existence of precedence, homology and co-location connections and based on their existence or non-existence announces it's on existence. It also considers the strengths of the three simple operators that are involved in the process. If precedence, homology, and co-location connections are found with high strengths then the dependency is also found to exist with high strength. If the events are not co-located but have high precedence and are homologous with an average strength then the dependency exists with an average strength. The same scheme is applied for low dependency with a homology with low strength. For rest of the cases, 0 is returned indicating no dependency.

C. Results of Evaluation over University Events

As mentioned in Section 1, our research question is to determine the frequency of events which EConnDetect is able to identify correctly. For this, we calculate the precision and recall parameters, for both context-match and analytic operators. Our collected email dataset comprised social events like sporting events, Mela (a type of celebration event), quiz competition, and farewell dinner. It also comprised several professional events like conference, seminars, and meetings. Finally, personal events like "lunch" or "tea" were also found in the collection.

TABLE III.   SAMPLE EVENTS FROM OUR COLLECTED EVENTS

Event ID	Event Title	Location	Date	Time	Participants	Category
E-1	Meta 2013	Main Campus FAST-NU Karachi	10 January 2013	9:00 AM – 4:00PM	Faculty –CS Students-CS Faculty-EE Student-EE	Social
E-2	Seminar	City Campus FAST-NU Karachi	1 January 2012	1:00PM - 2:00PM	Faculty BBA Students BBA	Professional
E-3	Gaming Competition	Main Campus FAST-NU Karachi	10 January 2013	1:00PM - 2:00PM	Faculty-CS Students-CS Students-EE	Social
E-4	Robocup Competition	Main Campus FAST-NU Karachi	10 January 2013	10:00 AM- 1:00PM	Faculty-EE Students-EE Students-CS	Social
E-5	Cricket Match	Main Campus FAST-NU Karachi	3 March 2013	1:00 PM - 4:00PM	Faculty-CS Students-CS	Social
E-6	Lunch	Main Campus FAST-NU Karachi	2 January 2013	2:00 PM	Shaukat, Imran, Hammad, Tariq Jawwad, Zubair	Personal
E-7	Procom	Main Campus FAST-NU Karachi	20 January 2012	9:00 AM – 4:00 PM	Faculty-CS Students-CS Faculty-EE Students-EE Director Campus	Professional
E-8	MS Thesis Evaluation	City Campus FAST-NU Karachi	15 June 2012	9:00AM- 2:00PM	Faculty-CS Faculty-EE	Professional
E-9	Procom	Main Campus FAST-NU Karachi	20 June 2013	9:00 AM-4:00 PM	Faculty-CS Students-CS Faculty-EE Students-EE Director Campus	Professional

Out of approximately 100 events, we manually assigned Professional tag to 55 events, Personal tag to 18 events, and Social tag to remaining 27 events. Although 100 events indicate a limited data set, EConnDetect discovered more than 10000 connections for this dataset. This number is large because there were many repeated connections; most connections are commutative and were counted from both sides. All the context match operators produced good results except concurrency and temporal subset operators; some concurrent events with a high strength of concurrency were identified by the system as concurrent with a lower level of strength. For analytical operators, EConnDetect failed to identify several connections and this was either because of misidentification of concurrency and temporal subset operators, or on our own choice of strength level of the simple operator used in the definition of a given analytical operator.

In Table 3, we show some sample university events from our event collection with their attributes, and in Table 4, we show connection information for a few connections found by applying algebra operators over pairs of events shown in Table 2. Here, columns Event 1 and Event 2 represent the pairs of events.

TABLE IV.   SAMPLE EVENT CONNECTIONS RELATED TO PAIRS OF EVENTS (SHOWN IN COLUMNS EVENT 1 AND EVENT 2) FOUND BY APPLYING VARIOUS ALGEBRA OPERATORS SHOWN IN TABLE 2

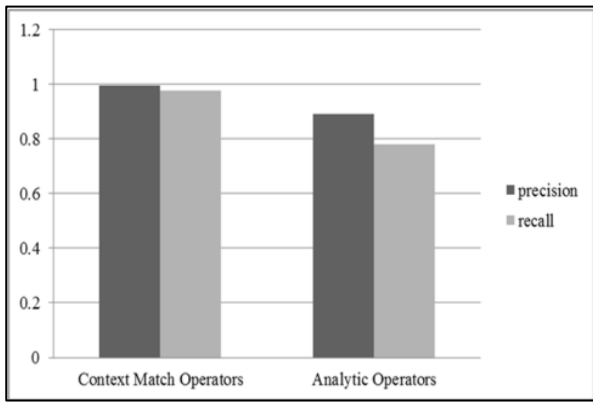| Event 1 ID	Event 2 ID	Connections	Strength
E3	E1	E3 is a sub-event of E1	High
E4	E1	E4 is sub-event of E1	High
E2	E7	Analogous Co-Located	High Average
E7	E8	Homologous Analogous Co-Location	Low High Average
E7	E9	Periodic Events	High
E3	E4	Overlapping Events	High
E1	E5	Homologous Analogous Co-Location	Low High High

Fig. 4.   Precision and Recall for Context-Match and Analytical Operators.

For instance, $E3$ (a gaming competition) and $E4$ (robocup competition) is connected to $E1$ (Mela) as a sub-event because both competitions were part of the Mela₃. Through manual activity, we determined that our university event log contained a total of 1300 event connections. Out of these, EConnDetect identified 1267 connections giving an overall recall of 97.5%. Amongst 1267 connections, EConnDetect failed to identify only 44 connections giving an overall accuracy of 96.5%. Out of 1300 connections, 1227 were context matches while the remaining 73 were analytical matches. In Figure 4, we show the precision and recall values for both context-match and analytic operators.

Our system achieved high precision values, i.e., almost 100% for context-match and around 90% for analytic operators. We also achieved a recall of almost 100% for context matches but relatively less for analytic operators (around 80%). This comparatively reduced performance because identification of connections that require an analysis is more complex as compared to those that require context matches. In Figure 5, we show the number of connections detected through each operator of our proposed algebra. As compared to other operators, the analogy, co-location, and homology operators detected most connections (around 300) followed by the precedence operator (around 150). The reason is that the attributes forming the base of these connections are common among various events. For example, all events held in a university premises are linked based on the common location attribute, even if the events have no other inter-relationship. We can apply similar arguments for the analogy and homology operators.
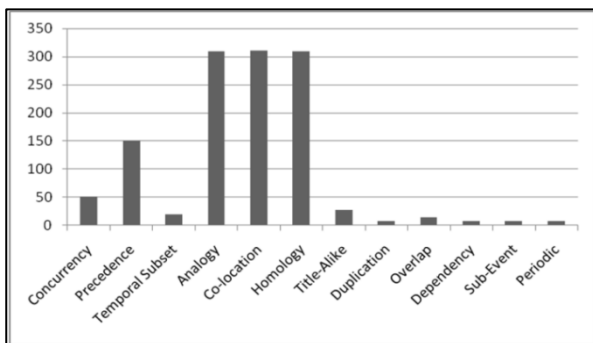

Fig. 5.   Frequency of Connections detected through each Operator of Proposed Event Algebra.

VI. Practical and Theoretical Implications and Limitations

Our proposed algebra provides a chronological chain of events with respect to common location(s), participant(s), or categories through context match operators. It also identifies a set of spatial, temporal or categorical linkages between events at a more granular level through analytical linkage operators. In our opinion, these results can be useful in three practical situations. Firstly, they can provide the extra information required to effectively detect unseen stories from a stream of events in a First Story Detection scenario [22, 23]. Secondly, our results can be used to effectively recommend future events and also assign priorities to events in an event-based recommendation scenario [24, 25]. Thirdly, the chronological events and analytical linkages together provide a visualization dimension in the contextual representation of the events (which is our proposed future work direction).
[26].

With respect to theoretical implication, we have provided a formal method of representing linkages between events through an event algebra. Our algebra has a rich, formal vocabulary of operators representing linkages. We can extend it to support new types of event information attributes and event linkages. We can also use it to represent events occurring in any application domain or appearing in any textual media. The core requirement is only to extract the event attributes for the specific domain and identify the possible linkages. Since we have selected the symbols for linkage operators from the set of mathematical operators based on similarity in semantics, therefore it is not difficult to find a meaningful symbol for a new operator. Our context match operators form the primitive group of operators that define any new spatial, temporal or categorical linkage. Finally, we have defined all existing complex analytical operators in our algebra using the primitive context match operators.

Our work has limitations with respect to the size of the processed data. If the data size exceeds the Megabyte level, then we will need to use big data analytics techniques to process our event algebra. For this, we plan to use MongoDB, a world-renowned database for big data as our backend storage. Through the Python programming language, we will encode the EventWeb queries to run over MongoDB by using PyMongo API. If the data is growing exponentially, we can also use the clustered version of MongoDB to store data over a distributed cluster. Another limitation of our work is that we cannot claim our proposed event algebra to function in any selected domain. It is possible that there are domains where we may need to propose more operators. This can only be done at application time.

VII. Conclusions and Future Work

Extracting important information regarding connections between events appearing over EventWeb is an existing requirement of the Web community. In this paper, we have developed an event algebra that extracts semantic information from EventWeb by identifying potential connections between events. We were unable to locate any related research work

that collectively incorporates all features of our algebra for information extraction. Specifically, we consider all five event information attributes (title, location, temporal attributes, participants, and category) and we detect previously unexplored connections. We also provide an analytical view over the connections through analytical operators, and we define the strength of connections. We have implemented a tool to implement the operators proposed in our algebra. We applied these operators to a set of detected events in the university domain to extract information regarding connections between these events. Out of 1300 connections that existed among 100 events, our tool correctly identified 1267 linkages. Hence, we achieved a precision and recall of 99% and 97% respectively for context match operators. Similarly, the precision and recall values for analytic operators were found to be approximately 90% and 80% respectively. We believe that these results validate the effectiveness of our algebra.

Our research has addressed the task of information processing in EventWeb, by linking events from multiple unique dimensions. Firstly, we address the two-fold objective behind linking events: 1) assistance in event planning and management, and 2) enhancing the experience of information search, visualization and story link detection over EventWeb. Secondly, in order to meet these objectives significantly, we attempt to identify all possibly existing connections. Moreover, we also cater for connections that required an analysis, along with the simple connections. We formulate all this through a robust event algebra that can is extensible for other domains. Finally, we develop a tool with simple user interface and validate our algebra effectively.

As future work, we are currently working to develop a contextual representation of event information components and connections. We also aim to extend our algebra to identify chains of connections. Moreover, we plan to use the analytical view provided by our operators in conjunction with the previous event participation history of the user for tagging events with priorities and generating recommendations. This will be part of a recommendation system attached to the link detection component, and that will deliver recommendations over event participation to support planning and management of events. We also aim to assign priorities to events by using the analytical operators along with the event participation history. In this context, we can use a participation history and a weightage feature-based approach for identifying the prioritized events. Finally, we intend to evaluate our algebra with events of other domains in the future.

REFERENCES

[1] S. Wasi, Z. A. Shaikh, S. Qasmi, H. Sachwani, R. Lalani, A. Chagani, Event information extraction system (eiee): Fsm vs hmm, World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering 7 (1) (2013) 5–12.

[2] S. Wasi, S. I. Jami, Z. A. Shaikh, Context-based email classification model, Expert Systems 33 (2) (2016) 129–144, eXSY-Oct-13-189.R4.

[3] W. R. van Hage, D. Ceolin, The simple event model, in: Situation Awareness with Systems of Systems, 2013, pp. 149–169.

[4] I. Corda, B. B. Bennett, V. Dimitrova, A logical model of an event ontology for exploring connections in historical domains (2011).

[5] B. B. Ilaria Corda, Vania Dimitrova, An ontological approach to unveiling connections between historical events, in: The 18th International Conference on Knowledge Engineering and Knowledge Management, Galway City, Ireland, 2012.

[6] R. Shaw, R. Troncy, L. Hardman, LODE: Linking Open Descriptions of Events, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 153–167.

[7] A. Hinze, A. Voisard, A parameterized algebra for event notification services., in: TIME, 2002.

[8] A. Hinze, A. Voisard, Eva: An event algebra supporting adaptivity and collaboration in event-based systems, Tech. Rep. TR-09-006, International Computer Science Institute, USA (2009).

[9] A. Nagargadde, V. Sridhar, K. Ramamritham, Representation and processing of information related to real world events., Knowl.-Based Syst. 20 (1) (2007) 1–16.

[10] B. Rink, C. A. Bejan, S. M. Harabagiu, Learning textual graph patterns to detect causal event relations., in: H. W. Guesgen, R. C. Murray (Eds.), FLAIRS Conference, AAAI Press, 2010.

[11] S. Chakravarthy, V. Krishnaprasad, E. Anwar, S.-K. Kim, Composite events for active databases: Semantics, contexts and detection, in: Proceedings of the 20th International Conference on Very Large Data Bases, VLDB '94, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1994, pp. 606–617.

[12] V. Uma, D. G. Aghila, Article: Temporal pattern mining and reasoning using reference event based temporal relations (retr), International Journal of Computer Applications 36 (3) (2011) 45–49, published by Foundation of Computer Science, New York, USA.

[13] X. Zhao, P. Jin, L. Yue, Discovering topic time from web news, Inf. Process. Manage. 51 (6) (2015) 869–890.

[14] M. Ruocco, H. Ramampiaro, Geo-temporal distribution of tag terms for event-related image retrieval, Inf. Process. Manage. 51 (1) (2015) 92–110.

[15] L. Derczynski, D. Maynard, G. Rizzo, M. van Erp, G. Gorrell, R. Troncy, J. Petrak, K. Bontcheva, Analysis of named entity recognition and linking for tweets, Inf. Process. Manage. 51 (2) (2015) 32–49.

[16] M. N. Hila Becker, L. Gravano, Event identification in social media, in: Twelfth International Workshop on the Web and Databases (WebDB 2009), Rhode Island, USA, 2009.

[17] H. Sayyadi, M. Hurst, A. Maykov, Event detection and tracking in social streams., in: E. Adar, M. Hurst, T. Finin, N. S. Glance, N. Nicolov, B. L. Tseng (Eds.), ICWSM, The AAAI Press, 2009.

[18] H. Khrouf, R. Troncy, EventMedia : Visualizing events and associated media, in 10th International Semantic Web Conference, Demo Session, ISWC, Eurecom, 2011.

[19] R. Troncy, B. Malocha, A. T. S. Fialho, Linking events with media, in: Proceedings of the 6th International Conference on Semantic Systems, I-SEMANTICS '10, ACM, New York, NY, USA, 2010, pp. 42:1–42:4. doi:10.1145/1839707.1839759.

[20] J. F. Allen, Maintaining knowledge about temporal intervals, Commun. ACM 26 (11) (1983) 832–843.

[21] J. S. Shaukat Wasi, Z. A. Shaikh, Contextual event information extractor for emails, Sindh University Research Journal 43 (1) (2011) 103–108.

[22] S. Petrovic, M. Osborne, V. Lavrenko, Streaming first story detection with application to twitter, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10, Association for Computational Linguistics, Stroudsburg, PA, USA, 2010, pp. 181–189.

[23] S. Petrovic, M. Osborne, V. Lavrenko, Using paraphrases for improving first story detection in news and twitter, in: Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 3-8, 2012,

Montreal, Canada, 2012, pp. 338–346. URL http://www.aclweb.org/anthology/N12-1034

[24] A. Q. Macedo, L. B. Marinho, R. L. Santos, Context-aware event recommendation in event-based social networks, in: Proceedings of the 9th
ACM Conference on Recommender Systems, RecSys '15, ACM, New York, NY, USA, 2015, pp. 123–130. doi:10.1145/2792838.2800187.

[25] X. Ji, Z. Qiao, M. Xu, P. Zhang, C. Zhou, L. Guo, Online event recommendation for event-based social networks, in: Proceedings of the 24th International Conference on World Wide Web, WWW '15 Companion, ACM, New York, NY, USA, 2015, pp. 45–46. doi:10.1145/2740908.2742742.

[26] N. Gronau, M. Neta, M. Bar, Integrated contextual representation for objects' identities and their locations, Journal of Cognitive Neuroscience, 20 (3) (2008) 371–388.