;

International Doctorate School in Information and Communication Technologies

DIT - University of Trento

REAL-TIME EVENT CENTRIC DATA INTEGRATION

Majed Ayyad

December 2014

Advisor:
Prof. Fausto Giunchiglia
Università degli Studi di Trento

Abstract

A vital step in integrating data from multiple sources is detecting and handling duplicate records that refer to the same real-life entity. Events are spatio-temporal entities that reflect changes in real world and are received or captured from different sources (sensors, mobile phones, social network services, etc.). In many real world situations, detecting events mostly take place through multiple observations by different observers. The local view of the observer reflects only a partial knowledge with certain granularity of time and space. Observations occur at a particular place and time, however events which are inferred from observations, range over time and space. In this thesis, we address the problem of event matching, which is the task of detecting similar events in the recent past from their observations. We focus on detecting Hyperlocal events, which are an integral part of any dynamic human decision-making process and are useful for different multi-tier responding agencies such as emergency medical services, public safety and law enforcement agencies, organizations working on fusing news from different sources as well as for citizens. In an environment where continuous monitoring and processing is required, the matching task imposes different challenges. In particular, the matching task is decomposed into four separate tasks in which each requiring different computational method. The four tasks are: event-type similarity, similarity in location, similarity in time and thematic role similarity that handles participants similarity. We refer to the four tasks as local similarities. Then in addition, a global similarity measure combines the four tasks before being able to cluster and handle them in a robust near real-time system. We address the local similarity by studying thoroughly existing similarity measures and propose suitable similarity for each task. We utilize ideas from semantic web, qualitative spatial reasoning, fuzzy set and structural alignment similarities in order to define local similarity measures. Then we address the global similarity by treating the problem as a relational learning problem and use machine learning to learn the weights of each local similarity. To learn the weights, we combine the features of each pair of events into one object and use logistic regression and support vector machines to learn the weights. The learned weighted function is tested and evaluated on real dataset which is used to predict the similarity class of the new streamed event.

Keywords

Hyperlocal events, event matching ; similarity measures ; actionable knowledge

Acknowledgments

I wish to express my sincerest gratitude to my thesis supervisor Prof. Fausto Giunchiglia. Every time, I was unsure about the right direction in my research course, a short discussion with him helped me to continue with a perfect reasoning and logic. Without his support, trust and positive thinking, this work wouldn't be completed.

During these years, I spent my time working with different Palestinian municipalities and public safety organizations. I would like to thank every mayor and general director who enabled me to understand their problems closely and allow me to use their data for research purposes.

Special thanks to my friends and colleagues in Palestine, who helped me in building the first prototype of the event matcher. I am in debt to Fahed AL-Khabbas, Mohammad Isleem, Mohannad Sabaana, Tareq Dofish, Raed Zidat and many other colleagues who tested the system and gave me their feedback.

Last but not least, my family contribution to every success I made in this life is uncountable. Their help, patience, sacrifices, and encouragement derive my success during the years. This thesis, is dedicated to my parents, wife and sons.

Contents

List of Tables

List of Figures

1 Introduction

1.1 The Context

In the broad sense, data fusion is the process of utilizing one or more data sources over time to assemble a representation of aspects of interest in an environment [Lambert, 1999]. The term is usually co-located with situation awareness which is "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future" [Endsley and Garland, 2000]. Usually, data fusion problems are studied in three types of environments [Bray, 1999]:

- **Designed World.** In which we have a relatively complete understanding of what exists in that world and how it operates.
- **Real World.** In which we only partially understand the physical phenomena that are being monitored and often have very little control over it.
- **Hostile World.** which is defined for defense applications in which some parts of it we understand and control, but other parts are less understood and controllable.

With [Endsley and Garland, 2000] definition of situation awareness, the role of events become clear. Events reflect the changes in the real world and usually require an action to such changes. In many scenarios, reaction to events is required immediately in near real-time. However, in real world situation, detecting events mostly take place through multiple observations by different observers. For some type of events, like the meteorological events, a sequence of constrained observations, if took place in some order and locations may signal a certain type of events. This order indicates particular properties the required observations must have and how the observations must be temporally and spatially related.

How do we make sense of data fusion ?

In data fusion the main motivating principle is to improve the quality of the information by employing more than one sensor to gather the data [Mitchell, 2007]. Essentially, data fusion doesn't require data interoperability, but making sense of data fusion and enhancing its quality is easier with data interoperability. In real-world environment, we cannot control the data format or the language used by an observer to describe a situation. This process is easy in designed world, but in real-world where sensors are human beings making sense of probed data requires the improvement of the quality at the level of [Mitchell, 2007]:

- **Representation:** a richer abstract and semantic meaning of individual data inputs; certainty (a linear probability of data sets before fusion).
- **Accuracy:** the standard deviation on data after fusion process is smaller than standard deviation by direct source.
- **Completeness**: bringing a complete view on the new informational situation gained by the current knowledge in that situation.

In this thesis, we are interested to improve the quality of data that are mainly probed from human sensors and in particular who report about hyperlocal [1]events such as vehicle accidents, crimes and felonies, traffic jams, floods, sit-in, gatherings and demonstrations, which are occurring at the level of a city, suburb, block, street and even at the level of a building. In this context, the efficient dissemination and processing of hyperlocal events, plays a vital role in many public and private organizations. Addressing this problem effectively has many practical applications. In particular, we envisage the following use cases:

- **Multi-tier responding agencies:** Law enforcement, public safety and homeland security assimilate local events in their decision-making systems to avoid a poor judgment chain from either forming or growing as well as to increase the situation awareness in their areas.

- **Journalists and news agencies:** many organizations or individuals rely on different sources to be instantly informed about breaking events.

- **Multi-national organizations:** situational awareness is a core issue for some international and multi-national organization either for security reasons for their staff or for humanitarian reasons.

- **Citizens:** In some hot places like in Palestinian cities, recently citizens start to use social media to increase the situation awareness before sending their children to school or traveling from one city to another.

A structured approach to a decision making process in a multi-tier agency is depicted in figure (1-2). The figure illustrates the end-to-end information flow from the site of the event until a first responder reaches that site. We can summarize the flow of information in the following main steps:

[1] The word is formed from affixation of the adjective *local* with prefix *hyper-* meaning 'more than usual or

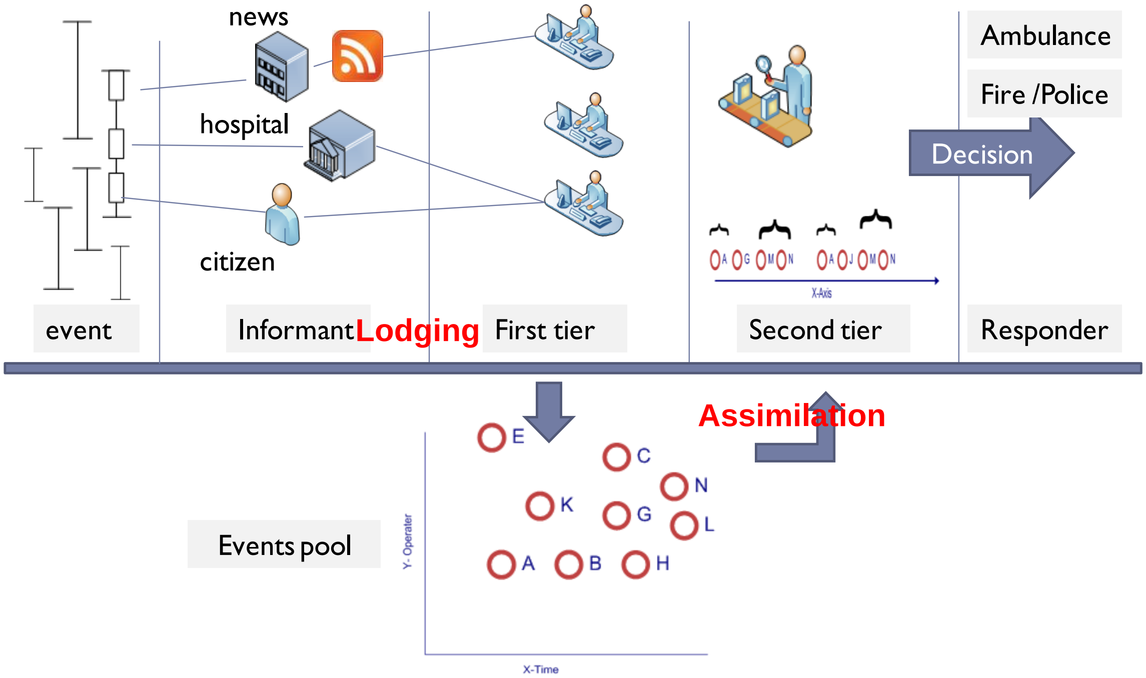


Figure 1-1 Flow of information in multi-tier agencies. First tier workers are called operators; Second tier workers are called commanders

Event occurrence: In real world, events erupt from a wide spectrum of sources. In the Palestinian context, where the events in this thesis are studied, events erupt accidently and frequently. Events such as clashes, demonstrations, confrontations, strikes, sit-ins, stone throwing, shootings, road blockage, breaches impact the life of many Palestinian citizens. Consequently, these events may cause other events such as injury, property damage, fatal or death events. In addition, criminal events, traffic events, meteorological events are also among the main types of events that derive the decision making process by many citizens and organizations.

Event Detection: In this thesis, we confine our study on events that need an action to be taken once an event has been detected. Sometimes this is called "Actionable knowledge". Actionable knowledge has been a hot topic in data mining, where the core idea is to make sense of the mined patterns by enabling the users to utilize them in their decision making. In our context, actions have been classified as:

- **Response:** after the event has been detected an action is needed to deal with the threat, hazard or risk

- **Preventions:** prediction of events based on historical analysis may require a prevention event or a plan to deal with the potential threat, hazard or risk.
- **No action:** With the ability to classify non-priority or non-life threading events, a decision of not taking any action may be considered.

In most cases, citizens are the main source of information about events. The bulk of emergency events are reported through phone calls. These are life-threatening calls and usually have dedicated call lines.

Event Lodging: In order to capture as much events as possible, a dedicated team is allocated to answer phone calls or capture event data from other sources such as RSS feeds, incident reports, street cameras, etc. Due to increasing number of events, usually this team (first tier) is not responsible of analyzing the events. Their main concern is to get as much information as possible from its source. The lodging process is standardized by using agreed vocabularies to describe the event context.

Event Assimilation: A second tier of workers is responsible for processing the lodged events, comparing current events with recent ones, triage events to their severity level, communicate, coordinate, collaborate with other agencies based on the intelligence derived from the lodged events. This step is the fundamental currency that drives the taken decision.

Decision Making: The taken decision depends on several factors which may include the emergency level of the incident, location of the event, available resources, dependency on other agencies, type of action needed, etc.

To illustrate the complexity facing the second tier of workers while processing the lodged events, consider the following example as depicted in figure (1-2). The timeline in this scenario shows the time of receiving the call from an observer. In this example, we have five different events that need immediate action from the responsible agency [fire fighters, police, medical services]. While the time difference between the first and second event is 3 minutes, the worker can use the event type to distinguish between the two events. However, the second, third and fourth events are not that obvious, since a car accident may cause a traffic jam as well as a dispute between drivers or owners of the cars. In this case, the worker needs to reason using the location of each event. The last event in the timeline is also problematic since this event may have two possibilities: a new fire event or a diminishing one. Again the location of this event might give an indication to the worker on how to proceed and which decision to make.

In general and in contrast to technical sensors, humans can cognitively identify and perceive complex events such as a storm or fire which are caused or constructed from different smaller events. A human can specify a relationship between different events based on their spatial, temporal and the mode of participation of different objects. Furthermore, humans can detect how an event evolves or fades through time and space. Despite of this, human capacity is limited, therefore with large volume of incoming events there is a possibility to drop some events, deploy a resource based on false alarms, deploy double resources, or causing a delay in the response time

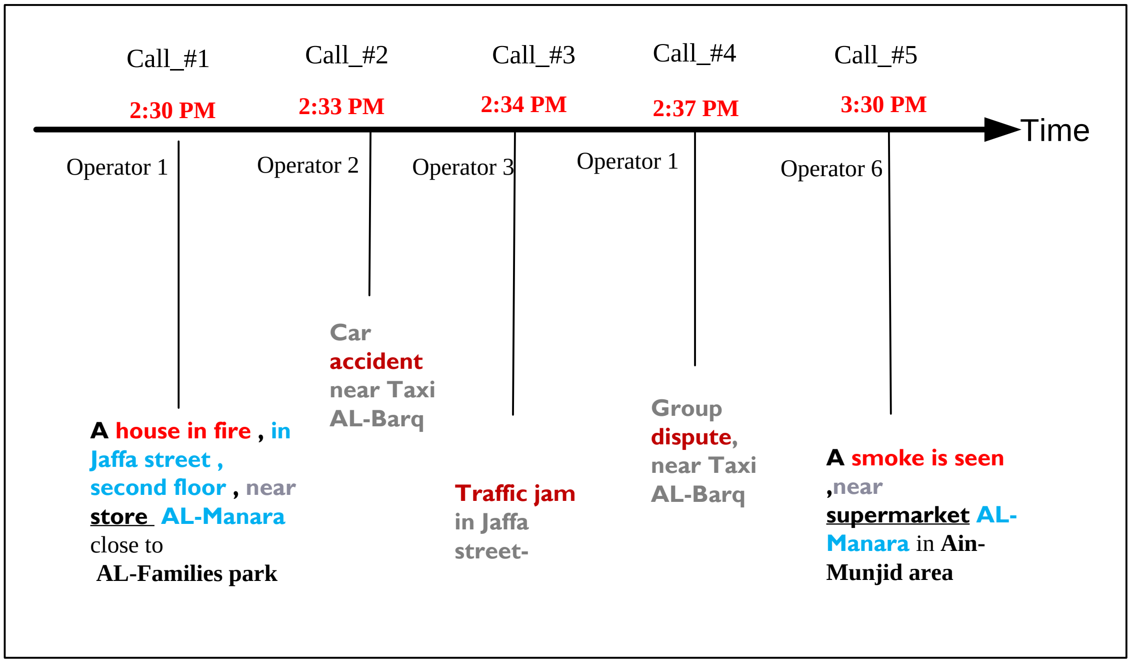

Call_#1
2:30 PM
Operator 1
A house in fire , in Jaffa street , second floor , near store AL-Manara close to AL-Families park
Call_#2
2:33 PM
Operator 2
Car accident near Taxi AL-Barq
Call_#3
2:34 PM
Operator 3
Traffic jam in Jaffa street-
Call_#4
2:37 PM
Operator 1
Group dispute, near Taxi AL-Barq
Call_#5
3:30 PM
Operator 6
A smoke is seen ,near supermarket AL-Manara in Ain-Munjid area
Time

Figure 1-2 Sample of received calls in the same city showing the call number, operator who received the call and the description of the event.

To maximize the ability of the second-tier workers (commanders) to identify similar events and refine the intelligence and knowledge from different observations, a utility that can perform the triage and the matching is needed. This will improve the quality of services of the agency and minimize the response time of the first responder. Over the years, the efficiency of public safety organizations has been measured based on measuring the response time, which is measured from the moment of receiving the emergency call until the first responder reaches the scene of the incident or event. Delays in response time has been attributed to many reasons. The main reason that can skew the response time is the time taken by dispatchers to triage incoming calls into the right priorities (high-priority and non-priority calls) and making sure that they are not allocating double resources to the same events and dispatching resources to the nearest point of the event. Furthermore, when resources are not at capacity, delays may be longer specially if the lack of resources are at the equipment and staff level. Table (1-1) list some of the metrics that are usually used to measure the performance.

Table 1-1 Some performance metrics in public safety organizations

Metric	Metric . Current Value	Metric Target Value
Detect false alarms	Less than 5 %	50%
Dropped calls	10 %	Less than 1 %
Avg. Target Response Time Critical Events	12-18 minutes	8 minutes or fewer
Distance from actual event location	500 meter	Less than 100 meter

A solution to this is to build a machinery that can help the worker in finding any similarity relationship between the incoming events based on their timings, locations, event types and other information provided by the informants such as the cause and how the event happened or what instruments are used and by whom. A machine can apply a temporal filter to exclude the events that are not significant and occurred outside the watching window of the worker. A spatial reasoning component is implemented to find any spatial relations between the locations mentioned in the three calls. A possible search is to find any containment relationship between the locations in the event predicates. In our example we have from call_1 : In ’Jaffa Street’ and from call_5 we have In ’Ain Munjid Area’ and thus we may find a containment relationship as in figure (1-3).

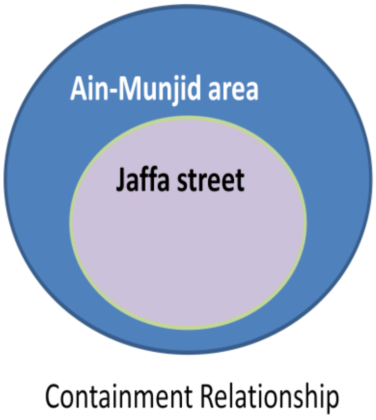


Figure 1-3 Containment relation between two regions

If information about the modus operandi is available, a thematic operators component might be used to compare the how and what facets of events . The situation in call_5 is more difficult because If none of the previous events has a terminates axiom, then call_5 might have two possibilities : a new fire event or a diminishing one.

1.2 The Problem

In this Thesis, we refer to the problem of determining whether two event descriptions (observations) refer to the same underlying entity as an event matching (linkage) problem. We define intuitively the concept matching as the task of linking a pair of events based on a joint relationship. In this context, similarity is the relationship that we would like to use as a link. Similarity indicates how much commonality and differences two stimuli (events) have. The notion of commonality and differences is used by Lin (1997) to define the similarity using an information theoretic approach based on the following three intuitions:

Intuition 1: The similarity between A and B is related to their commonality. The more commonality they share, the more similar they are.
Intuition 2: The similarity between A and B is related to the differences between them. The more differences they have, the less similar they are.
Intuition 3: The maximum similarity between A and B is reached when A and B are identical, no matter how much commonality they share

In [Tversky, 1977] contrast model, the similarity of object A to object B is a function of the features common to A and B (symbolized "A and B"), those in A but not in B (symbolized "A-B") and those in B but not in A (" B-A"). Also the problem of event matching has its roots in philosophy and linguistic which was discussed by [Zacks and Tversky, 2001] [Davidson, 1985] [Quine, 1985] [Davidson, 2001] [Mourelatos, 1978] under the event identification problem. As shown in Table 1. taking for example set 1, two events are similar, from a philosophic point view, if they have the same time and location. The other sets are combination of one or more elements of: time, location, physical object, cause and effect, existential conditions and properties.

Table 1-2 Different criteria for event identification

Criterion	Set 1	Set 2	Set 3	Set 4
Time	X	X		
Location	X			
Physical object		X		
Cause and effect				X
Existential conditions			X	
Properties		X		

Cognitive scientists also studied the processes of similarity judgment. [Larkey and Markman, 2005] identified different roles for similarity which underlies fundamental cognitive capabilities. Their theory is that there are two types of differences between compared items: Alignable differences which are differences between corresponding elements of compared items. For example, an alignable difference between a car and a motorcycle is the number of wheels they have. Nonalignable differences are differences between elements that do not correspond or differences where an element in one representation does not correspond to any element in the other representation. For example, a seat belt is a nonalignable difference between a car and a motorcycle because amotorcycle has no restraining device that corresponds to a car's seat belt. Alignable differences and nonalignable differences are psychologically distinct. Similar items tend to have more alignable differences than dissimilar items[Markman and Gentner, 1993].

The transformation model measures similarity through the use of transformational distance [Hahn and Chater, 1998] [Goldstone, 2004]. The concept of transformational distance is defined as a function of the complexity required to transform the representation of one stimulus into the representation of another. According to Kolmogorov complexity theory [Goldstone, 2004], to which the complexity of a representation is the length of the shortest computer program that can generate that representation. For example, the conditional Kolmogorov complexity between the sequence 1 2 3 4 5 6 7 8 and 2 3 4 5 6 7 8 9 is small, because the simple instructions add 1 to each digit and subtract 1 from each digit suffice to transform one into the other. In other words, the similarity between two entities is the smallest number of operations that a computer program needs to transform one entity into the other.

Having reviewed different approaches from information theory, philosophy, linguistic and cognitive science, we illustrated the complexity of dealing with similarity. In order to clarify similarity assessment framework in the event context, we illustrate the assessment problem using the following example.

Motivating Example

Observer.1 and ***Observer.2*** are looking at an occurring event. Each observer can see the event from a different angle. Let us assume that *Observer.2* is moving, while O*bserver.1* is not. Also let us assume that Observer.1 observed the occurrence 5 minutes after it was observed by Observer.2. The shaded area denoted by view.1 identify the boundary of the region and its environment that can be seen by Observer.1, while view.2 identify the boundary of the region and its environment that can be seen by Observer.2. Both observers are describing the event based on their angle of observation, therefore they are reasoning about the event using what is called by [C. Ghidini and F. Giunchiglia, 2001] [Giunchiglia, 1993] reasoning with viewpoints, and reasoning about belief. The first observer believes that it is a theft, while the second believes that it is a burglary. Although both observers are using the local environment to describe the event

Both observers call the emergency to report about their observations. A controller working on the emergency room log the two phone calls and keep receiving other phone calls from other observers without knowing if these phone calls are for the same event or no.

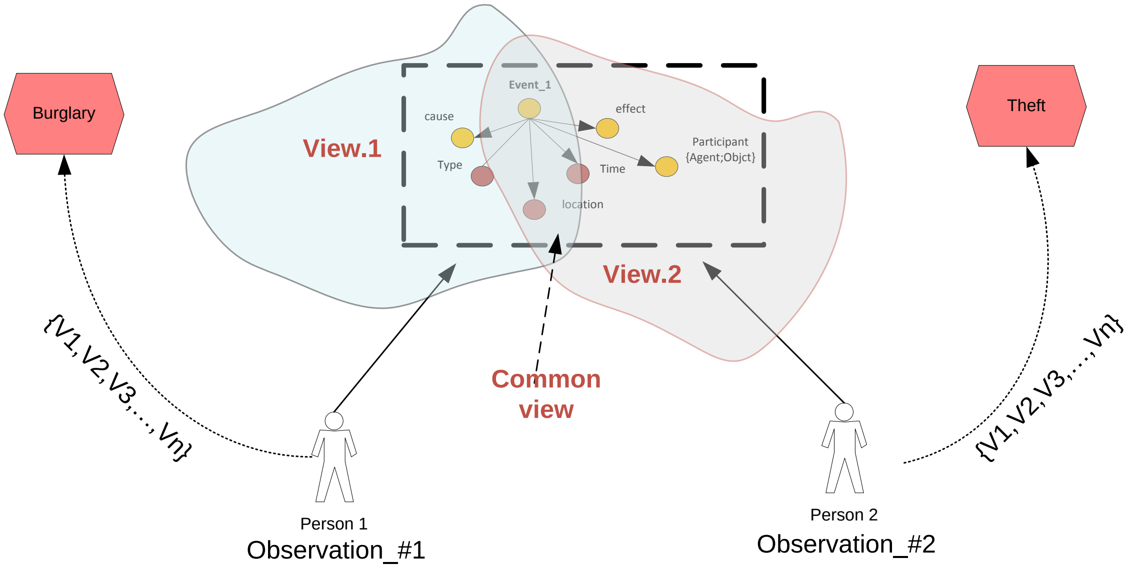


Figure 1-4 Local view of two observations

The log of the phone calls is encoded similar to the following excerpt:

Observation.1

<Type > Burglary

<Location: slot1> Ramallah
<Location: slot2> In city center
<Location: slot3> near supermarket Baghdad

<Date> 12-12-2013
<Time> around 8:30 in the morning

<Agent> {{person1:attrib1,attrib2,…},
{person2:attrib1,attrib2,…}
}
<Recipient:slot1> a car
<Recipient:slot2> {old man:hasage; }

<related-to> event-4

Observation.2

<Type > Theft

<Location: slot1> Ramallah
<Location: slot2> In Tokyo street
<Location: slot3> Not far from AL-Manara square

<Date> 12-12-2013
<Time> in the early morning

<Agent> {person1,person2}

<Recipient:slot1> a white car
<Recipient:slot2> {attrib1,attrib2,… }

<related-to>

Figure 1-5 Context of two observed events

To create meaningful information out of these observations, a second tier of workers try to find which observations are related to the same event before taking any decision. This analysis should be done usually in near real-time. For a person trying to find multiple observations of the same event, this requires comparing and contrasting the components of each new observation with existing ones within a pre-defined time-window.

As illustrated in this example, finding similarity between different observation depends heavily on the context of the observation. We define an event and its context as follow:

Definition 2.1. (Event). An occurrence (behavioral activity or natural phenomenon) happening at a specific time and location. An event entity is a tuple that takes the form:

$$\boldsymbol{E} \;=< \boldsymbol{e_{id}}, \textbf{type, time, Loc, Ctx>} \qquad (1)$$

Where:

- $\boldsymbol{e_{id}}$**,** the unique identifier of an event
- **type**, The type of the occurrence reflects the final type after the analysis of multiple observations.
- **time,** the temporal part of the event. The time of occurrence can be either an instant or a time period. A time period can be with known or unknown ends**.**
- **Loc,** the spatial part of an event. The location of an event can be either a physical or virtual location**.**
- **Cxt,** the event context. Is a set of observations related to a single event.

Definition 2.2 (Context). Is the meta-information taken from the local knowledge of the observer which is related to the detected event, and is represented as a tuple of the form:

$$\textbf{Cxt} = \{\, \boldsymbol{O_1}, \boldsymbol{O_2}, \dots, \boldsymbol{O_n} \} \qquad \textbf{(1)}$$

O* = <ID, observation-data, confidence****)***

O = <Obs_time, obs_loc, obs_type, participant, instrument, recipient, cause, effect, confidence>

- **-Id,**
- **Observation-data is:**
  - **Obs_time,** The time the occurrence observed. Time can be either an instant or a time period. A time period can be with known or unknown boundaries.
  - **obs_loc,** The location of the occurrence from the perspective of the observer.
  - **obs_type,** The type of the occurrence from the perspective of the observer
  - **Participant,** Participant can have different roles such as agents or recipients.
  - **Instrument,**

    - **Cause,** A set of events that might be the reason for this occurrence to happen**.**
    - **Effect,** A set of events that be resulted from this occurrence**.**
- **Confidence**, the level of confidence the observer has in describing the occurrence.

Despite the availability of different models for comparing two objects or entities, selecting one model cannot handle the complexity of comparing a pair of events. Events are complex entities that require employing a similarity framework which can handle:

- Semantic similarity among event-types.
- Spatial similarity among event-locations
- Temporal similarity among event-times
- Feature-based or alignment based similarity among event-participants.

Existing algorithms only handle separately each component. Furthermore, measuring semantic similarity between event types only return a value indicating the degree of similarity between a pair objects. They do not indicate why two objects are similar or not similar. Exploiting the context which includes location, time, environmental conditions, participants, activity, nearby objects, instruments, and nearby people, explains why two events are assigned a particular similarity score and help in detecting errors in the automated similarity measures as well as strengthen our understanding on what factors contribute to similarity between events.

Problem Definition

Consider an observation stream as a time ordered series of observation records $O = \{o_1, o_2, o_3, \dots, o_n\}$ and a stream of events $\mathrm{E} = \{e_1, e_2, e_3, \dots, e_m\}$, where o_i has the form

$$o_i \ =\{\ o_{RDF}^{type}, o_{RDF}^{Time}, o_{RDF}^{location}, o_{RDF}^{participant}, o_{RDF}^{description}, 0_{RDF}^{state}, o_{RDF}^{cause}\}$$

Consider a delta-based time sliding window model W = $\{\ \mathrm{TS}_{i-b+1}, \dots, \mathrm{TS}_{i-1}, \mathrm{TS}_i\}$, where TS_i is the latest time slot and TS_{i-b+1} is the first time slot in the window and the first to be evicted when the time shifts by b to the new slot TS_i .

Hence the event matching -problem is to group the events arriving in the last b time periods of the stream S into a set of clusters $\mathrm{C} = \{\ \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \dots, \mathrm{C}_n\}$ such that each cluster C_i is associated with only similar events.

1.3 The Solution

We consider the problem of determining whether a pair of events, (e_i, e_j) belong to the same class or not, as a pairwise binary classification problem. The main objective of pairwise classification is to infer the similarity relation between two events. We have two classes: similar and not similar. To learn the similarity relation between a pair of events, we trained a classification function g from a set of training examples where for each pair (e_i, e_j) of the example, we know if the pair belongs to the same class ($y_{ij} \coloneqq 1|similar$) or not ($y_{ij} \coloneqq -1|$ not similar).

$$g(e', e) = \begin{cases} 1\ if\ e', e\ belong\ the\ the\ same\ class \\ \quad\quad -1\ (otherwise) \end{cases}$$

As shown in figure 1-6, given a dataset of similar pairs and non-similar pairs of events and a feature representation that characterize these relations. We can infer a model that if given a new pair of events, can predict the relation between them.

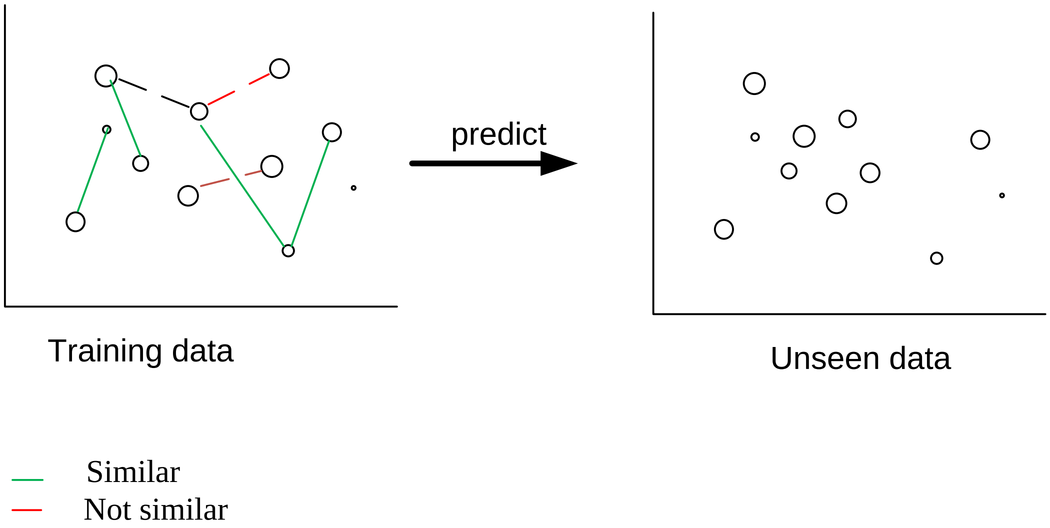


Figure 1-6 Relation learning example

We decompose the matching task into three sub-tasks:

1. **Feature selection:** we use the similarity measures as the features of similar or non similar events.
2. **Learning task:** from the training set, we learned a metric so the prediction model could be used to infer the relation between a new pair of events. The output of this

phase is a similarity matrix which is used to assign the new event to a new or existing cluster.

3. **Validation**: we validate the model using real-data set .

For metric learning, we consider the two events as two records and follow the procedure of record linkage problem. The theory and techniques of record linkage date back to pioneering work by Fellegi and Sunter [Fellegi and Sunter, 1969] in their seminal paper "A Theory for Record Linkage". In relational management database system, a record linkage problem is addressed by applying different similarity algorithms [Elmagarmid et al., 2007] [Banu, 2012] at three different levels:

- Record level
- Field level
- Index level

We also show the system architecture used to automatically compare events on a stream and identifies past events similar to newly detected ones. The events we monitor are local in contrast to global events, that is, they happen at a specific region in a given time period.

1.4 The Contribution

In this thesis, we provide the following contributions:

The main contribution of this thesis is represented in chapter 3 and chapter 4. Mainly the work on identifying suitable and adequate similarity measures for each element of the observed event. In essence, this thesis includes the following important contributions:

- Provides adequate type, spatial, temporal and thematic role similarity functions. The design of these similarity functions considers similarity knowledge combined from cognitive point of a view as well as functional point of view. Similarity measures in addition to semantic similarity and relatedness considers:
    - Location relations (topology, orientation and direction)
    - Temporal relations (linguistic terms and fuzzy intervals)
    - Causes and effects
    - Agent
    - Patient
    - Functions
    - Participant features
    - Instruments

- A quantitative analysis of different similarity measures and their limitations to be used in finding similar events. We analyzed the adequacy of existing similarity measure for the task of learning the weights of event types . For other aspects or facets of the event, we discussed the concept of similarity from numerous view-points and their computational approach, in particular, the alignmenet model, transformational model and relational model, of similarity.

- A computation framework to calculate similarity is presented using supervised learning approaches. Mainly similarity between pairs of events are learned using logistic regression and support vector machines.

- We also evaluate our approach and show that the approach is applicable to real-life scenarios and applications.

1.5 Structure of the Thesis

The thesis is organized as follows.

Chapter 2 introduces the state of the art covering the topics of similarity measures and learned metrics.

Chapter 3 is divided into four main sections covering: type-similarity; location similarity ; time similarity and thematic role similarity.

Chapter 4 provides a computational framework to learn similarity weights and describes the architecture of the system for event matching.

Chapter 5 describes the evaluation measures to assess and select the model as well as methods of collecting and validating the data used in our experiments.

Chapter 6 shows the results of our experiments.

Chapter 7 provides a review of related work

Chapter 8 summarizes the work and gives outlook for future work.

2 State of the Art

In the previous chapter, we explored the main theories and broad definition of similarity between two stimuli or objects. In this chapter, we will introduce the approaches and techniques to measure and learn the similarity. In section (2.1), the notion of similarity, its definition and different related similarity measures will be introduced focusing on the following dimensions: concept similarity, spatial similarity, temporal similarity and attributal similarity. In particular, this will cover the four main dimensions of any event. In the second section, we will introduce the learning theory and two algorithmic approaches used to learn a model. In particular, we will introduce logistic regression and support vector machine.

2.1 Similarity Measures

As argued by [Goodman, 1972] [Medin et al., 1993] there is no global agreement on how similarity is measured or defined. Goodman argues that the similarity of A to B is an ill-defined unless one can say in what respects. To define a frame of reference to the task of finding similar events, we argue that two events are similar based on the similarity between their types, spatial, temporal and participants aspects. Since we are comparing a pair of events using their contexts then we confine our literature review to the similarity measures that are related to the context elements:

- **Time,** The time the occurrence observed. Time can be either an instant or a time period. A time period can be with known or unknown boundaries.
- **Location,** The location of the occurrence from the perspective of the observer.
- **Type,** The type of the occurrence from the perspective of the observer
- **Participant,** Participant can have different roles such as agents or recipients.

A similarity measure is a function which computes the degree of similarity between pair of objects. Although there is no universal agreement as to a definition of similarity, its range manifestations map to the range [-1,1] or [0,1].

Definition 2.1[Balcan, 2008] A similarity function over X is any pairwise function $\mathrm{K}: X \times X \rightarrow [-1,1]$. Where K is a symmetric similarity function if $\mathrm{K}(x, x') = \mathrm{K}(x', x)$ for all x, x'

Besides the formalism introduced in Definition 2.1, other mathematical ways to represent similarity can be defined using distance notation and ranking [Richter,1992] . A ranking similarity is relative similarity between two pairs.

Ranking. For two pairs x,y and z,w, SIM(x,y,z,w) means that y is at least as similar to *x* as *z* is to w. This is equivalent to $\mathrm{SIM}(\mathrm{x},\mathrm{y}) \leq SIM(z,w)$

Distance. A function *d*(*x*, *y*): X × *X*→ R+ measuring the distance between *x* and *y*. A function $\mathrm{d}(\mathrm{x},\mathrm{y}) \rightarrow\ R^{+}$, is commonly called a distance measure if it satisfies the following properties:

Non-negativity:

$$\mathrm{D}(\mathrm{A},\mathrm{B}) \geq 0$$

- Identity of indiscernibles:

$$\mathrm{D}(\mathrm{A},\mathrm{B}) = 0\ ;\ \ \mathrm{if}\ \mathrm{A} = \mathrm{B}$$

- Symmetry:

$$\mathrm{D}(\mathrm{A},\mathrm{B}) = \mathrm{D}(\mathrm{B},\mathrm{A})$$

- Subadditivity (triangle inequality):

$$\mathrm{D}(\mathrm{A},\mathrm{B}) +\ \mathrm{D}(\mathrm{B},\mathrm{C}) \geq \mathrm{D}(\mathrm{A},\mathrm{C})$$

2.1.1 Taxonomy Based Similarity Models

Many similarity measures have been proposed based on the availability of comprehensive taxonomies, ontologies or lexical databases such as [WordNet, 2010] or the Gene Ontology [GO, 2000] in bioinformatics. A vast amount of existing similarity measures use WordNet as the basis to compute similarity between concepts. Measuring the similarity or distance between concepts is based on measuring the semantic similarity or semantic relatedness between two concept words or phrases. The difference between semantic similarity and semantic relatedness is explained by is [Resnik, 1995] as "Semantic similarity represents a special case of semantic relatedness: for example, cars and gasoline would seem to be more closely related than, say, cars and bicycles, but the latter pair are certainly more similar".

Most similarity measures using WordNet or any other similar structure use the following terms to quantify the similarity between two concepts. For any two concepts $c_1\ and\ c_2$, the following terminology is used:

Dist (c_1, c_2): The length of the shortest path from synset c_1 to synset c_2 .

LCS (c_1, c_2): The lowest common subsumer of c_1 and c_2 . The Least Common Subsumer of two concepts A and B is "the most specific concept which is an ancestor of both A and B", where the concept tree is defined by the is-a relation

Depth(c_1): the length of the path to synset (c_1 from the global root entity, and depth(root)=1.

deep_max: the max depth(ci) of the taxonomy

hypo(c): the number of hyponyms for a given concept c.

node_max: the maximum number of concepts that exist in the taxonomy.

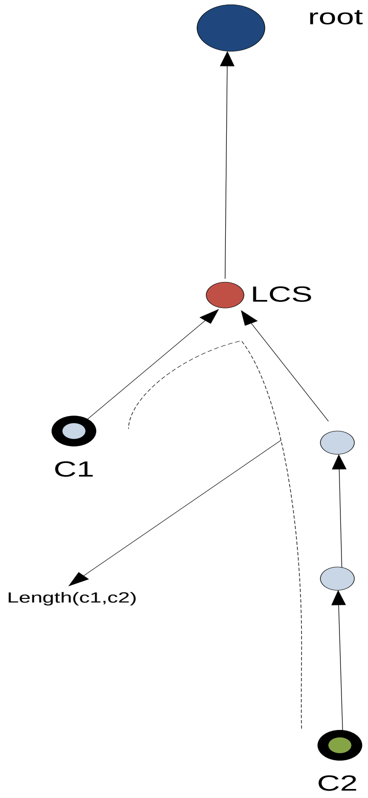


Figure 2-1 Node count terminology

Wu and Palmer's Similarity Measure. [Wu and Palmer,1994]

The similarity between a pair of concepts is calculated using the formula:

$$\mathrm{sim}_{\mathrm{wup}}(C_1, C_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3}$$

Where

- N3 is the number of nodes from the most least common subsumer (LCS) of $C_1$and C_2 to the root.
- N1 is the number of nodes on the path from C_1 to the node of least common subsumer (LCS)
- N2 is the number of nodes on the path from C_1 to the node of the least common subsumer (LCS)
- The score in wup is 0 < score <= 1.
- When two concepts are the same, the score is one

Wup depends on the depth of the nodes

Leacock and Chodorow's Similarity Measure [Leacock and Chodorow, 1998]

$$\mathrm{sim}_{\mathrm{LC}}(\mathrm{C}_1, \mathrm{C}_2) = -\log\frac{\mathrm{Dist}(\mathrm{c}_1, \mathrm{c}_2)}{2 * \mathrm{D}}$$

Where, $Dist(c_1, c_2)$ is the shortest path between the synsets and D the total depth of the of the taxonomy. The measure us node-counting for finding the $Dist(c_1, c_2)$

PATH Similarity Measure [Rada et al.,1989]

This module computes the semantic relatedness of word senses by counting the number of nodes along the shortest path between the senses in the 'is-a' hierarchies of a taxonomy.

$$\mathrm{Sim}_{\mathrm{PATH}}(\mathrm{C}_1, \mathrm{C}_2) = \frac{1}{\mathrm{Dist}(\mathrm{c}_1, \mathrm{c}_2)}$$

The measure also uses node-counting scheme

Resnik Similarity Measure [Resnik, 1995]

Resnik showed that semantic similarity depends on the amount of information that two concepts have in common, this shared information is given by most least common subsumer (LCS) that subsumes both concepts. If LCS does not exist then the two concepts are maximally dissimilar.

Resnik semantic similarity is defined as:

$$\mathrm{sim}_{\mathrm{res}}(\mathrm{C}_1, \mathrm{C}_2) = -\log\mathrm{P}(\mathrm{lcs}(\mathrm{c}_1, \mathrm{c}_2))$$

Where, the information content can be quantified as the negative of the log likelihood,

$$\mathrm{IC}_{\mathrm{LCS}} = -\log \mathrm{P(c)}$$

the probabilities of concepts in the taxonomy is estimated using the formula:

$$\mathrm{P(c)} = \frac{\sum_{\mathrm{w} \in \mathrm{W(c)}}^{\mathrm{n}} \mathrm{count(w)}}{\mathrm{N}}$$

Where, where W(c) is the set of words (nouns) in the corpus whose senses are subsumed by concept c, and N is the total number of word (noun) tokens in the corpus that are also

present in WordNet. A snippet from information content file [http://ws4j.googlecode.com/svn-history/r3/trunk/edu.cmu.lti.ws4j/src/main/resources/ic-semcor.dat]

```
wnver::eOS9lXC6GvMWznF1wkZofDdtbBU
1740n 128767 ROOT
1930n 69661
2137n 59062
2452n 3669
2684n 39997
3553n 32734
3993n 0
4258n 20896
4475n 20800
5787n 0
5930n 0
6024n 0
6150n 0
6269n 8
6400n 0
6484n 87
7347n 19753
7846n 19196
15388n 1124
```

the probabilities of concepts in the taxonomy were estimated from noun frequencies gathered from the one-million-word Brown Corpus of American English. Frequency counts are based on the number of senses a word has. Because Resnik measure is using as a corpus to calculate the information content, it is sometimes classified under the corpus based similarity models.

Jiang and Conrath’s Similarity Measure [Jiang and Conrath, 1997]:

Jiang and Conrath's measures semantic distance between two concepts taking into consideration both the information content and edge-counting. Therefore, sometimes this method is classified under hybrid methods that combines both: information content and edge-counting. The distance is calculated by the following formula :

$$\mathrm{Dist}_{\mathrm{JC}}(\mathrm{C}_1, \mathrm{C}_2) = \mathrm{IC}(\mathrm{c}_1) + \mathrm{IC}(\mathrm{c}_2) - 2 \times \mathrm{IC}(\mathrm{LSC}(\mathrm{c}_1, \mathrm{c}_1))$$

$$\mathrm{Dist}_{\mathrm{JC}}(\mathrm{C}_1, \mathrm{C}_2) = 2\log \mathrm{p}(\mathrm{LSC}(\mathrm{c}_1, \mathrm{c}_1) - (\log \mathrm{p}(\mathrm{c}_1) + \log\ \mathrm{p}(\mathrm{c}_2))$$

Therefore the similarity is

$$\mathrm{sim}_{\mathrm{JC}}(\mathrm{C}_1, \mathrm{C}_2) = \frac{1}{2\log \mathrm{p}(\mathrm{LSC}(\mathrm{c}_1, \mathrm{c}_1) - (\log \mathrm{p}(\mathrm{c}_1) + \log\ \mathrm{p}(\mathrm{c}_2))}$$

Lin's Similarity Measure [Lin, 1998]

Lin similarity measure is based on the following three intuitions as a basis to his model :

1. The similarity between arbitrary objects *A* and *B* is related to their commonality; the more commonality they share, the more similar they are.
2. The similarity between *A* and *B* is related to the differences between them; the more differences they have, the less similar they are.
3. The maximum similarity between *A* and *B* is reached when *A* and *B* are identical, no matter how much commonality they share.

$$\mathrm{sim}_{\mathrm{lin}}(\mathrm{C}_1, \mathrm{C}_2) = \frac{2 \times \log\ \mathrm{p}(\mathrm{LSC}(\mathrm{c}_1, \mathrm{c}_1)}{\log\ \mathrm{p}(\mathrm{c}_1) + \log\ \mathrm{p}(\mathrm{c}_2)}$$

Lesk Similarity Measure [lesk, 1985]

Lesk proposed that the relatedness of two words is proportional to to the extent of overlaps of their dictionary definitions. The adapted leask [Banerjee and Pedersen, 2002] LESK measure is based on adapted uses WordNet as the dictionary for the word definitions. A combination score

$$\mathrm{sim}_{\mathrm{lesk}} = \max(\mathrm{combinatios} = \prod_{\mathrm{i}=1}^{\mathrm{N}} |\mathrm{W}_{\mathrm{i}}\ |)$$

The combinations are calculated from the overlap between the two concepts synset glos, hypo gloss and hype gloss.

2.1.2 Spatial Similarity Models

There are substantial work on similarity between geo-concepts [Schwering and Raubal, 2005] [Shariff et al., 1998] [Rodríguez and Egenhofer, 2003] [Rodríguez et al., 1999] [Rodríguez and Egenhofer, 2004]. Shariff et al. developed a model defining the geometry of spatial natural-language relations following the premise *topology matters*, *metric refines* [Shariff et al., 1998]. [Schwering and Raubal, 2005] show that people's choice of spatial relations to describe two objects differs depending on the meaning of objects, their function, shape and scale. The matching-distance measure [Rodríguez and Egenhofer, 2004] computes similarity between geo-concepts by combining different weighted similarity functions from the sub-classes of the main concept which are part, functions and attributes. The distance-matching measure is based on the comparison of distinguishing features and uses the shortest path for determining the distinguishing features in an entity class's definition.

While measuring similarity between geo-concepts is an important aspect, we need to focus on this thesis on measuring the proximity of two places or locations rather than computing similarity between their classes. Therefore in the rest of this section, we will focus on reviewing related literature that measures similarity between locations based on their topological, orientation and directional relations.

[Freksa, 1992b] created the conceptual neighborhood network based on Allen's 1-D interval relations. The conceptual neighborhood approach is based on the transformation model, in which similarity is measured according to the distance between two concepts in a network. Using the conceptual neighborhood, [Egenhofer and Al-Taha, 1992] worked on spatial relation similarity for the topological relations. They derived gradual changes of the topological relationship based on Egenhofer's 9-intersection model. They created a conceptual neighborhood of the topological relationship and calculated the distance as table 2-1 illustrates.

Table 2-1 The Topology distance between the eight topological relationships for two spatial regions

	Disjoint	Meet	Equal	Inside	coverdBy	Contains	Covers	overlap
Disjoint	0	1	6	4	5	4	5	4
Meet	1	0	5	5	4	5	4	3
Equal	6	5	0	4	3	4	3	6
Inside	4	5	4	0	1	6	7	4
coverBy	5	4	3	1	0	7	6	3
Contains	4	5	4	6	7	0	1	4
covers	5	4	3	7	6	1	0	3
overlap	4	3	6	4	3	4	3	0

[Papadias and Dellis, 1997] extended this model into a higher dimensional space to address spatial relationship similarity on topology, direction and metric distance. For higher dimensions they consider a relation set r which represents a disjunction of relations. The distance between a relation set r and a primitive relation R is the minimum distance between any relation of the relation set and R:

$$d(r,R) = min_{R_k \in r}(d(R_k, R))$$

Topology-Direction-Distance (TDD) [Li and Fonseca,2006]

The TDD spatial similarity model utilizes a similarity measure that integrates four similarity models which are the geometric model, the feature contrast model, the transformation model, and the structure alignment model. The TDD model builds on the Conceptual Neighborhood Approach [Freksa, 1992] [Egenhofer and Al-Taha,1992]. The level of comparison is taken at two levels :

1. Scene level : for a scene the spatial or non-spatial relationship is measured. The spatial relationships are measured using the following relations: topological, directional, metric distance and distribution. The non-spatial relationship is measured using attribute distance.
2. object level : for objects the attributes of the objects in the scene are measured. Object attributes are measures using types of objects and attribute comparison.

$$\mathrm{Sim} = \left(\mathrm{C_{scene}} + \mathrm{C_{object}}\right) - (\mathrm{D_{scene}} + \mathrm{D_{object}})$$

$$\mathrm{C_{scene}} = (\mathrm{C_{topological}} + \mathrm{C_{directional}} + \mathrm{C_{metricDistance}})$$

$$\mathrm{C_{object}} = (\mathrm{C_{geometric}} + \mathrm{C_{thematic}})$$

The final similarity is a weighted measure

$$S(A,B) = \theta * commonality - D(A,B)$$

and

$$D(A,B) = \alpha * (aligned\_difference) + \beta\ (non\_aligned\ difference)$$

by default the model gives different weights for each parameter

$$\theta = 1.5\ ;\ \alpha = 1\ ; or\ \beta = 0$$

The Topological Relationship

The computational framework is based on the transformation cost, but unlike the traditional transformation which assumes that transformation across all edges is the same, the TDD considers two types of transformation: inter-group and intra- group. If two nodes belong to different groups, the transformation cost is called inter-group cost; otherwise, the transformation cost is called intra- group cost. Directed by this principle, in Figure (2-2), adapted from [Li and Fonseca, 2006], the inter- group cost is set as 3, while the intragroup cost is set as 2 with an exception of transforming from *contain* to *contain&meet*. Nodes of *contain* and *contain&meet* can be considered as a sub-group within the group of *overlap in different levels*, hence the transformation cost is set as 1 which is one degree less than the intra- group cost.

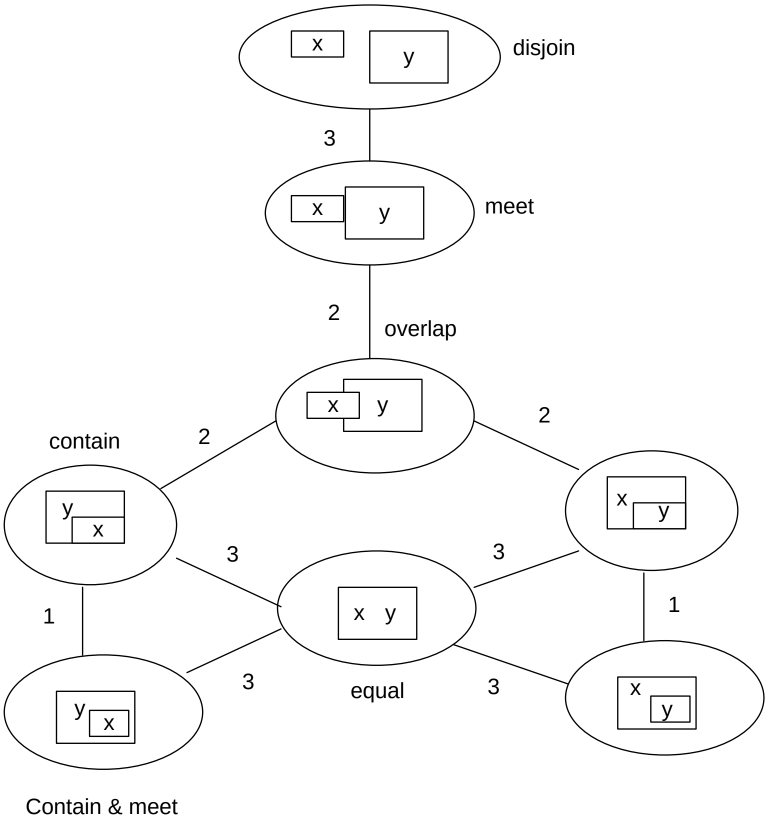
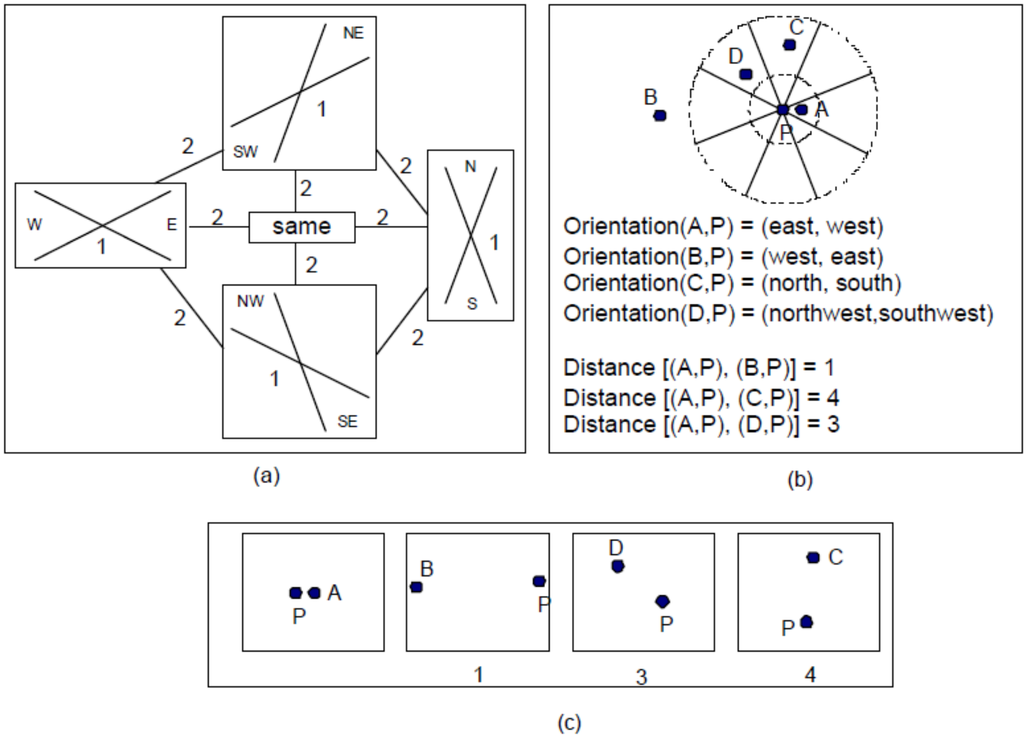


Figure 2-2 Conceptual neighborhood network of topological relationships -polygons

Directional Relationship

In the TDD model,using the transformation cost from one node to another in the p/2 directional network as shown in figure 2-3, which constitutes of 5 nodes {east, west}, {northeast, southwest}, {north, south}, {northwest, southeast}, and {same}. The cost of the transformation from one node to its neighbor is 2. The cost for switching the direction inside a node is 1 as.



Orientation(A,P) = (east, west)
Orientation(B,P) = (west, east)
Orientation(C,P) = (north, south)
Orientation(D,P) = (northwest,southwest)

Distance [(A,P), (B,P)] = 1
Distance [(A,P), (C,P)] = 4
Distance [(A,P), (D,P)] = 3

Figure 2-3 (a) direction network; (b) pattern examples; (c) ranking of similarity for the patterns in (b)

Distance calculation

Using a metric distance network of four nodes ({*equal*, *near*, *medium*, *far*}) as shown figure 2-4, the transformation cost is set as 1. If in one scene, the metric distance between the two objects is near, while in the other scene, the metric distance between the two objects is far, the transformation cost is 1 + 1 = 2.

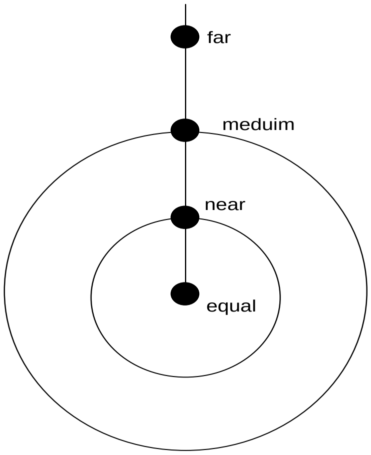


Figure 2-4 Metric distance network

2.1.3 Temporal Similarity Models

In this section we introduce a special type of similarity measures to compare two time intervals of fuzzy characteristics. It is very common that users or observers describe the time of an event using a fuzzy temporal term such as “in the early morning” and “ around 8:30”. In the literature there is a substantial work on comparing fuzzy objects, based on fuzzy-set-theoretical concepts.

We confine our review here to methods using generalized fuzzy numbers, which is a common approach to represent time intervals and time instants. A generalized fuzzy number $\mathrm{A} = (\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{w})$, where $0 \le \mathrm{a} \le \mathrm{b} \le \mathrm{c} \le \mathrm{d} \le 1$ and $0 \le \mathrm{w} \le 1$, is a fuzzy subset of the real line R with membership function μ_A which has the following properties[chen and chen, 2003] :

1. μ_A is a continuous mapping from R to the closed interval [0,w]
2. $\mu_A(\mathrm{x}) = 0$ for all $x \in (-\infty, a\,]$
3. μ_A is strictly increasing on [a,b]
4. $\mu_A(\mathrm{x}) = \mathrm{w}$ for all $x \in [b, c]$, where w is a constant and $0 \le \mathrm{w} \le 1$
5. μ_A is strictly decreasing on [c,d]
6. $\mu_A(\mathrm{x}) = 0$ for all $x \in (d, \infty)$

In a generalized fuzzy number, if μ_A is linear in [a,b] and [c,d] then it is called a generalized trapezoidal fuzzy number.

Similarity measures between generalized fuzzy Numbers

For any 2 trapezoidal fuzzy numbers $A = (a_1, a_2, a_3, a_4)$ and $B = (b_1, b_2, b_3, b_4)$, there exists different approaches to find similarity between fuzzy numbers .

- **Chen similarity measure** [Chen, 1996]

$$S(A,B) = 1 - \frac{\sum_{i=1}^{4} |a_i - b_i|}{4}$$

- **Hsieh and Chen similarity measure** [Hsieh and Chen, 1999]

$$S(A,B) = \frac{1}{1 + d(A,B)}$$

where,

$$d(A,B) = P(A) - P(B)$$

and

$$P(A) = \frac{a_1 + 2a_2 + 2a_3 +, a_4}{6} ; P(B) = \frac{b_1 + 2b_2 + 2b_3 +, b_4}{6}$$

- **Simple center of gravity method (SCGM)** [Chen and Chen, 2003]

$$S(A,B) = \left[1 - \frac{\sum_{i=1}^{4} a_i - b_i}{4}\right] * (1 - |x_A^* - x_B^*|)^{B(S_A, S_B)} * \frac{\min (y_A^*, y_B^*)}{\max (y_A^*, y_B^*)}$$

Where,

$$y_A^* = \begin{cases} \frac{w_{A\left(\frac{a_3 - a_2}{a_4 - a_1} + 2\right)}}{6} \, if \; a_1 \neq a_4 \\ \frac{w_A}{2} \; if \; a_1 = a_4 \end{cases}$$

$$x_A^* = \frac{y_A^* \, (a_3 + a_2) + (a_4 + a_1)(w_A - y_A^*)}{2w_A}$$

$$B(S_A, S_B) = \begin{cases} 1 \; if \; S_A + S_B > 0 \\ 0 \; if \; S_A + S_B = 0 \end{cases}$$

2.1.4 Feature-Based Similarity Models

The feature model is based on a set-theoretic representational model . As shown in figure (2-5), A-B and B-A are the set of unique features for each object, where $A \cap B$ is the set of common features shared between the two objects. Similarity measures of feature models underlie the assumption that similarity of concepts increases the more common and the less distinct features these concepts have. The most prominent representatives of the feature-matching model is Tversky's contrast and ratio model [Tversky, 1977] .

Tversky's “Contrast Model” assumes that the similarity of object a to object b is a function of the features common to a and b ("A and B"), those in a but not in b (symbolized "A-B") and those in b but not in a (" B-A"). In this model we have three components as illustrated in Figure (5-2) : common features of *A* and *B*, distinct features of *A* not in *B*, distinct features of *B* and not in.

A similarity measure based Tversky's model is given as

$$\mathbf{S(a,b) = xf(a\ and\ b) – yf(a\text{-}b) – zf(b\text{-}a).}$$

Here, S is an interval scale of similarity, f is an interval scale that reflects the salience of the various features, and x, y and z are parameters that provide for differences in focus on the different components.

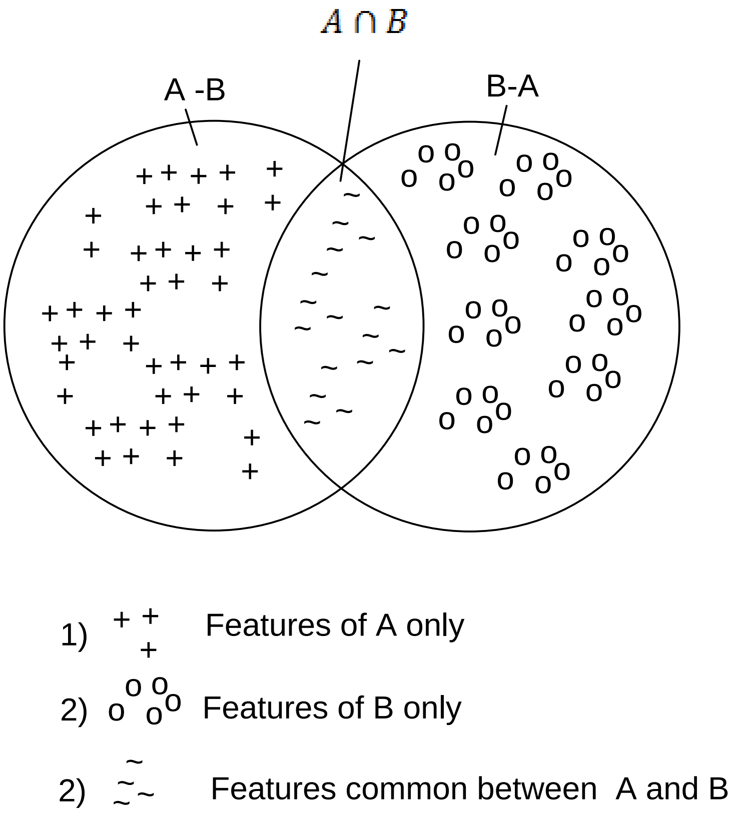


Figure 2-5 Representation of two objects that each contains its own unique features and also contains common features

Tversky also proposed the ratio model as another matching function based on the combination of $f(A \cap B), f(A - B), and\ f(B - A)$. The Ratio model is defined as follows:

$$S(A, B) = \frac{f(A \cap B)}{f(A \cap B) + \alpha f(A - B) + \beta f(B - A)}$$

Similar to this approach, the Matching-Distance Similarity Measure (MDSM) was proposed by [Rodríquez and Egenhofer, 2004] which was developed for similarity measurement of geospatial terms. This category of models was based on the ratio model that extends the original feature model by introducing different types of features and applying them to terms. There are also other similarity functions based on set theoretic measures such as Jaccard coefficient, Overlap coefficient, and Dice coefficient.

2.2 Learning Models

The General Setting for Statistical Learning Problems from examples comprises three components [Vapnik, 1999] :

1. A generator of random vectors, drawn independently from a fixed but unknown distribution P(x) ;
2. A supervisor that returns an output vector for every input vector, according to a conditional distribution function1P(y|x), also fixed but unknown;
1. A learning machine capable of implementing a set of functions . $\mathrm{f}(\mathrm{x}, \alpha), \alpha \in \Lambda$

The problem of learning is that of choosing from the given set of functions $\mathrm{f}(\mathrm{x}, \alpha), \alpha \in \Lambda$, the one which predicts the supervisor's response in the best possible way. The selection is based on a training set of random independent identically distributed (i.i.d.) observations drawn according to

$$\mathrm{P}(\mathrm{x}) = \mathrm{P}(\mathrm{x})\mathrm{P}(\mathrm{y}|\mathrm{x})$$

For our task, our approach is to learn a model from examples of event pairs which are labeled similar (+1), and ones that are labeled dissimilar (-1) . The objectives in learning similarity are:

- To develop a similarity classifier, that is, when given a novel pair of events, as accurately as possible, predicts the label of similarity {-1,1} for this pair.

- To provide a framework for similarity search form past events, without the need to apply similarity classifier to every possible pair of events.

The function chosen by the learning machine is denoted by $\mathrm{f}(\mathrm{x}; \theta)$ where θ is a parameter vector that should be learned to fit the data.

Since we consider the problem of determining whether a pair of events, (e_i, e_j) belong to the same class or not, as a pairwise binary classification problem. In the following sections we will introduce two approaches that are commonly used to in classification problems.

2.2.1 Logistic Regression

Logistic regression is part of a category of generalized linear models. The logistic regression model extends the linear regression model by linking the range of real numbers to the range 0-1. It is a type of multivariate regression that has a predictive model that can be used when the target variable is a categorical variable. The technique aims at modeling the relationship between a set of independent variables and the probability that a case is a member of one of the categories of the dependent variables. There are two types of logistic regression: Binary logistic regression which is used for two groups and Multinomial Logistic Regression that can be used with more than two groups. In this thesis we consider only binary logistic regression.

Logistic regression has many uses [Garson, 2009] It is used to predict a dependent variable on the basis of continuous and/or categorical independents; to determine the percentage of variance in the dependent variable explained by the independents; to rank the relative importance of independents; to assess interaction effects; and to understand the impact of covariate control variable.

1. A logistic regression model is used when the outcome variable is dichotomous.
2. Logistic regression uses binomial distribution.
3. Logistic regression does not assume a linear relationship between the dependents and the independents
4. The dependent variable in the logistic regression analysis need not be normally distributed (but does assume its distribution is within the Poisson, binomial or gamma).
5. Logistic regression coefficients estimate the odds ratio for each of the independent variables used in the model
6. The models predicts the probability within a population of an individual becoming or not becoming a case
7. Tabachnick and Fidell (2001) indicate that logistic regression is a good model when using different types of predictor variables. In this case, continuous and categorical variables were used in building a predictive model.

2.2.2 Logistic Regression Model

The basic assumption with logistic regression (binary output) is that if we have an experiment with X;y, where X the dataset of experiments and y is the binary outcomes. For each experiment $x_i \in X$ the outcome is either $y_i = 1$ or 1 or $y_i = 0$. We want to model the conditional probability Pr(Y = 1|X = x) as a function of x; any unknown parameters in the function are to be estimated by maximum likelihood.

Since the response variable (y_i) for logistic regression is always binary (assuming only two values), its distribution is binomial.

$$y_i \sim B(n_i,\pi_i),(i=0 \text{ } or \text{ } 1)$$

$\mathrm{n_i}$ is the numbers of Bernouli trials and $\mathrm{\pi_i}$ is the probability of being in the success group $\mathrm{y_i} = 1$, and $(1 - \mathrm{\pi_i})$ is the probability of being in the group $\mathrm{y_i} = 0$. The binomial distribution has distribution function

$$f_i(y_i) = \binom{n_i}{y_i} \pi_i^{y_i}(1-\pi_i)^{n_i-y_i}$$

taking natural log on the equation above and let

$$\theta_i = \ln\frac{\pi_i}{1-\pi_i}$$

then the unkonwn probability π_i is equal to

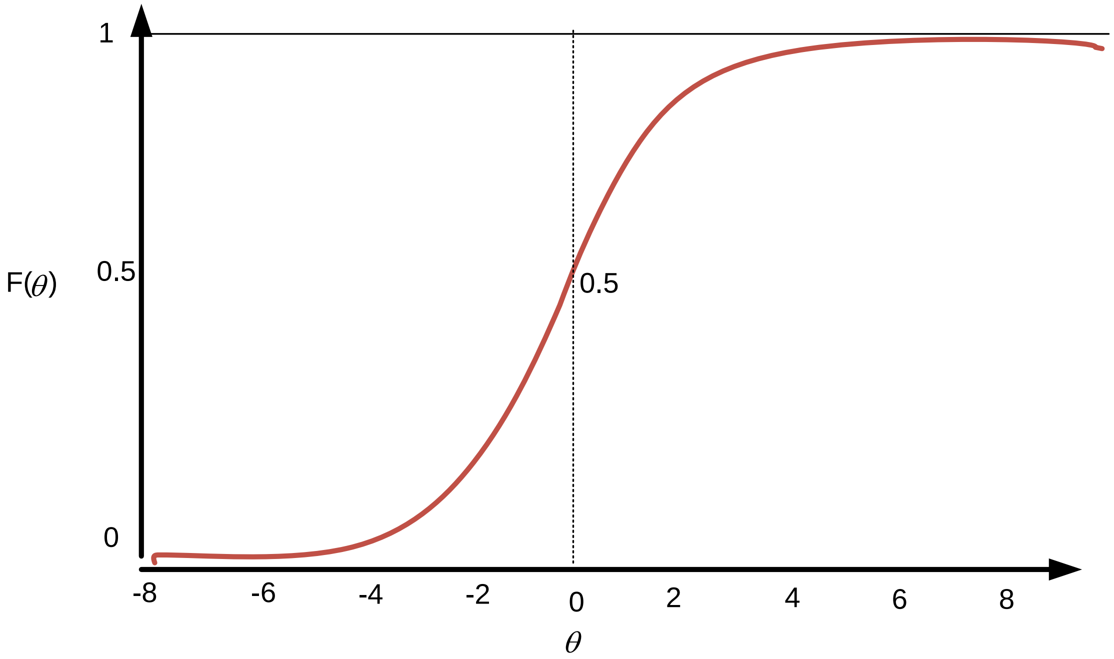
$$\pi_i = \frac{e^{\theta_i}}{1+e^{\theta_i}}$$

Let the variable $\boldsymbol{\theta_i}$ given by

$$\theta_i = \beta_0 + \beta_1 x_1 + \beta_0 x_2 + \cdots + \beta_k x_k$$

$$\boldsymbol{f(\theta_i) = \frac{1}{1+ e^{-\theta_i}}}$$

$$f(\theta_i) = \pi_i = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k)}}$$



2-6 Logistic regression function

As shown in figure 2-6, the logistic regression function takes as an input, any value from negative infinity to positive infinity, whereas the output is confined to values between 0 and 1. The variable θ is a measure of the contribution of all the risk factors used in the model, while f(θ)represents the probability of a particular outcome, given that set of risk factors.

The central mathematical concept that underlies logistic regression is the logit—the natural logarithm of an odds ratio. The logit is the natural logarithm (ln) of odds of Y, and odds are ratios of probabilities (π) of Y happening. Logistic regression applies the logit transformation to the dependent variable. In essence, the logistic model predicts the logit of Y from X [Peng et al., 2002b].

Odds of an event are the ratio of the probability that an event will occur to the probability that it will not occur. If the probability of an event occurring is π, the probability of the event not occurring is (1- π). Then the corresponding odds is a value given by

$$Odds(case = 1; event) = \frac{\pi}{1 - \pi}$$

With logistic regression the mean of the response variable π in terms of an explanatory variable x is modeled relating π and x through the equation $\pi = \ \alpha + \beta x$.

In this formula, if the value of x is large then the value of $\alpha + \beta x$ is large making the value of π not in the range 0 and 1. Which is not accepted since the probability should be between 0 and 1. The solution for this problem is to transform the odds using the natural logarithm [Lee and Ingersoll, 2002]. With logistic regression we model the natural log odds as a linear function of the explanatory variable

$$\theta_i = \ln \frac{\pi_i}{1-\pi_i} = \log\,(odds)$$

and recall that

$$\theta_i = \beta_0 + \beta_1 x_1 + \beta_0 x_2 + \cdots + \beta_k x_k$$

then

$$\theta_i = \ln \frac{\pi_i}{1-\pi_i} = \log(odds) = \beta_0 + \beta_1 x_1 + \beta_0 x_2 + \cdots + \beta_k x_k = logit(\pi)$$

This indicates that the independent observations variables are linearly related to the logit of the dependent. (Menard, 2001).

Under the logistic regression model, the parameters β_0 and β are estimated by the method of maximum likelihood of observing the sample values [Menard, 2001]. Maximum likelihood will provide values of β_0 and β which maximize the probability of obtaining the data set. Assuming the likelihood of the parameters is given by

$$L(\theta) = p(y|X;\theta)$$

$$= \prod_{i=1}^{m} p(y^{(i)}\,|x^{(i)};\theta)$$

$$= \prod p(x_i;\theta)^{y_i}\,(1-p(x_i;\theta)^{1-y_i}$$

Since it is easier to work with the log likelihood

$$l(\theta) = l(\beta_0,\beta) = \sum_{i=1}^{n} y_i \log p(x_i) + (1-y_i)\log 1 - p(x_i)$$

To find the values of (θ), we take the partial derivative of the log likelihood with respect to the parameters, set the derivatives equal to zero, and solve. But since this a transcendental equation, and there is no closed-form solution, we can however approximately solve it numerically.

$$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} (y_i - p(x_i; \beta_0, \beta))x_{ij}$$

We can learn the weights either by using gradient descent or Newton's Method

2.2.3 Regularization

It is well-known that regularization is required to avoid over-fitting, especially when there is a only small number of training examples, or when there are a large number of parameters to be learned and the degree of over-fitting depends on several factors [Ng,2005].:

- Number of training examples—more are better
- Dimensionality of the data—lower dimensionality is better
- Design of the learning algorithm—regularization is better

There are three types of regularizations: L0, L1 and L2 [Hastie et al.,2001] [Ng,2005]. L1 regularized logistic regression requires a sample size that grows logarithmically in the number of irrelevant features and L2 regularized logistic regression, under rotationally invariant algorithms, required a sample size that grows linearly in the number of irrelevant features.

L0 norm (sum of non zero entries

$$\|\theta\|_0 = \sum_{i=1}^{n} |\theta_i|^0$$

L1 norm (sum of non zero entries ; L1 norm drives many parameters to zero

$$\|\theta\|_1 = \sum_{i=1}^{n} |\theta_i|^1$$

L2 norm (sum of non zero entries ;L2 norm does not achieve the level of sparseness as L1

$$\|\theta\|_2 = \sum_{i=1}^{n} \theta^2$$

There are other strategies to produce sparse models such as elastic net regularization [Zou and [Zou and Hastie, 2005] and LASSO [Tibshirani, 1996]. The elastic net regularization tries to combine the best of L1 and L2 by using a shrinkage and selection method that produces a sparse model with good prediction accuracy, while encouraging a feature grouping effect. LASSO tries to get the best of the Ridge regression which is a continuous process that tries to shrink the coefficients of the features of less importance which have no effect on the actual output and the subset selection which is a discrete process; its regressors are either retained or totally excluded from the model.

2.2.4 Batch vs. Stochastic Gradient Descent

Logistic regression (LR) learns weights so as to maximize the likelihood of the data .

$$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} (y_i - p(x_i; \beta_0, \beta)) x_{ij}$$

$$\beta_{new} := \beta_{old} - \alpha \frac{1}{m} \sum_{i=1}^{n} (y_i - p(x_i; \beta_0, \beta)) x_{ij}$$

In this thesis we will use gradient descent to learn the weights. Gradient descent is divided into two categories: stochastic (also called on–line) and batch (also called off–line) learning. Stochastic is chosen either because of the very large data set(or may be redundant) training set. On the other hand Batch training is fast for small training set. The following procedure illustrated the difference between the two methods .

```
Batch mode Gradient Descent

until [number of iterations or other criteria]

    1. Compute the gradient
           $\beta_{new} := \beta_{old} - \alpha \frac{1}{m} \sum_{i=1}^{n} (y_i - p(x_i; \beta_0, \beta)) x_{ij}$
    2. End
```

```
Stochastic mode Gradient Descent

Do until [chosen stop criteria]

 For each training example $x_i \in X$
    1. Compute the gradient
          $\beta_{new} := \beta_{old} - \alpha \frac{1}{m} \sum_{i=1}^{n} (y_i - p(x_i; \beta_0, \beta)) x_{ij}$
    2. End
```

2.2.5 Kernel Methods

Another approach to data classification is to treat the given data as inner products in some Hilbert space. Support vector machine (SVM), which is based on Vapnik's statistical learning theory [Vapnik, 1999] utilizes kernel methods, and maximum margin classifiers for classification-based learning. In the following sub-sections, we provide a summary of these concepts and how they could be applied to learn the similarity between a pair of events. Like logistic regression, it requires a set of training examples with each marked as belonging to one of the categories. What makes SVMs different and more efficient is the use of kernel trick which maps the inputs into higher-dimensional feature.

The basic idea in kernel methods [Hofmann et al.,2008] is to map data from the input space into a high dimensional space (some Hilbert space) by means of a feature map. Since the feature map is normally chosen to be nonlinear, a linear model in the feature space corresponds to a nonlinear rule in the original domain.

Most data analysis methods outside kernel methods use feature mapping to do a prediction. For each x in the set of objects concerned by the learning problem each object is represented by a set of features $\emptyset(\mathrm{x}) \in \mathcal{F}$, with $\mathcal{F}$ a high dimensional feature space. however, in kernel methods instead of mapping $\emptyset : \mathcal{X} \to \mathcal{F}$, a real valued comparison function $\mathrm{k} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is used which is equivalent to representing the data set of objects by $\mathrm{N} \times \mathrm{N}$ similarity matrix of pairwise comparisons. The kernel function k is defined as follows:

Definition 2.1 A function $\mathrm{k} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a positive definite kernel iff it is symmetric, that is $k(x, x') = k(x', x)$ for any two objects $x, x' \in \mathcal{X}$, and positive semi-definite that is

$$\sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j k(x_i, x_j) \geq 0$$

For any N >0 and any choice of real numbers $c_1, \dots, c_N \ \in \mathbb{R}$

A kernel function can be seen as the dot product of the feature representation of two objects x, x'

$$k(x, x') = \emptyset(x)^T \emptyset(x') \text{ for any } x, x' \ \in \ \mathcal{X}$$

Examples of kernel functions:

- Linear kernel (identity kernel) : $\mathrm{K(x, x') = (x}^T\mathrm{x')}$
- Polynomial kernel with degree *d*: $\mathrm{K(x, x') = (x}^T\mathrm{x' + 1)^d}$
- Radial basis kernel with width σ: $\mathrm{K(x, x')} = e^{\frac{\|\mathrm{x-x'}\|^2}{\sigma^2}}$
- Sigmoid kernel with parameter a and r: $\mathrm{K(x, x') = tanh(}a\mathrm{x}^T\mathrm{x' + r)}$

2.2.6 Support Vector Machines

Since we need to solve a binary classification problem, in the coming section we will focus only presenting SVM mathematical foundation for the binary classification case. Our goal is to solve a binary classification problem by using a linear model in the Hilbert space. The linear model is represented by the following formula:

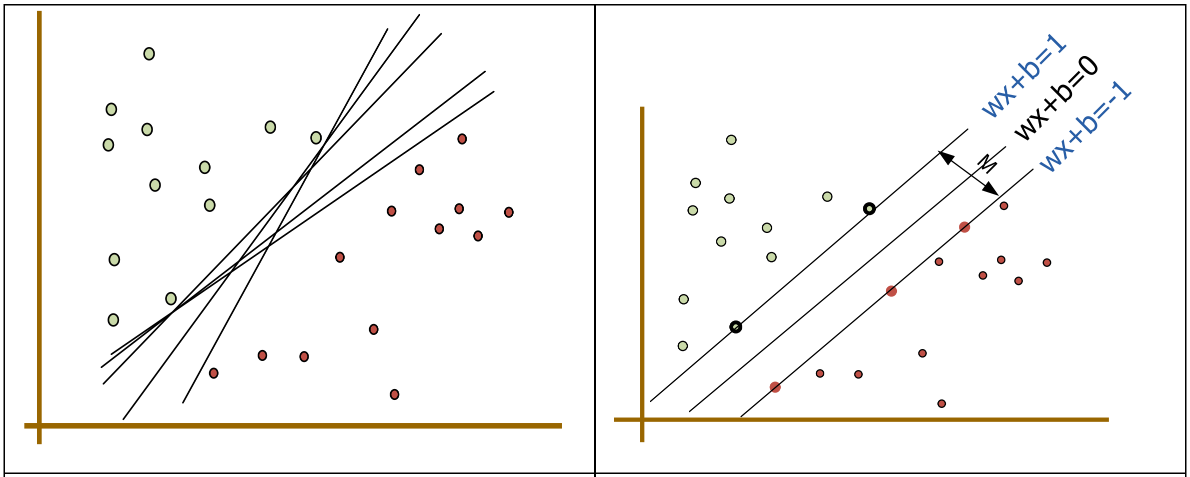
$$y(x) = \theta^T \emptyset(x) + \ b$$

Where θ and b are the parameters,$\emptyset(x)$ is the feature representation set of N objects $x_1, x_2, \dots, x_N$ and y(x) is the output of the prediction that depends on the sign of y(x), where $y \in \{-1,1\}$

2.2.6.1 Binary Linearly Separable Case

In the linearly separable case, there exists one or more hyperplanes that may separate the two classes represented by the training data with high ccuracy. As show in Figure (2-7):

(a) shows many separating hyperplanes (in the case of a two-dimensional input the hyperplane is simply a line). The main question is how to find the optimal hyperplane that would maximize the accuracy on the test data. The intuitive solution is to maximize the gap or margin separating the positive and negative examples in the training data. The optimal hyperplane is then the one that evenly splits the margin between the two classes.



a) More than one solution. Different hyperplanes could classify the data

b) The hyperplane that maximizes the margin between the two classes

Figure 2-7 LINEARLY SEPARABLE CASE

(b), the data points that are closest to the separating hyperplane are called *support vectors*. In mathematical terms, the problem is to find $f(\mathbf{x}) = (\mathbf{w}^T\mathbf{x}_i + b)$ with maximal margin, such that:

$\mathbf{w}^T\mathbf{x}_i + b = 1$ for data points that are *support vectors*

$\mathbf{w}^T\mathbf{x}_i + b > 1$ for other data points

Assuming a linearly separable dataset, the task of learning coefficients **w** and *b* of support vector machine $f(\mathbf{x}) = (\mathbf{w}^T\mathbf{x}_i + b)$ reduces to solving the following constrained optimization problem:

find **w** and *b* that minimize: $\frac{1}{2}\|\mathbf{w}\|^2$

subject to: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \quad \forall i$

Note that minimizing the inverse of the weights vector is equivalent to maximizing $f(\mathbf{x})$.

This optimization problem can be solved by using the Lagrangian function defined as:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} - \sum_{i=1}^{N} \alpha_i [y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) - 1], \text{ such that } \alpha_i \geq 0, \forall i$$

where $\alpha_1, \alpha_2, \ldots \alpha_N$ are Lagrange multipliers and $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots \alpha_N]^T$.

The support vectors are those data points $\mathbf{x}_i$ with $\alpha_i > 0$, i.e., the data points within each class that are the closest to the separation margin.

Solving for the necessary optimization conditions results in

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

where, $$\sum_{i=1}^{N} a_i y_i = 0$$

By replacing $\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$ into the Lagrangian function and by using $\sum_{i=1}^{N} a_i y_i = 0$ as a new constraint, the original optimization problem can be rewritten as its equivalent *dual problem* as follows:

Find $\boldsymbol{\alpha}$ that maximizes $$\sum_i \alpha_i - \tfrac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

subject to $$\sum_{i=1}^{N} \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad \forall i$$

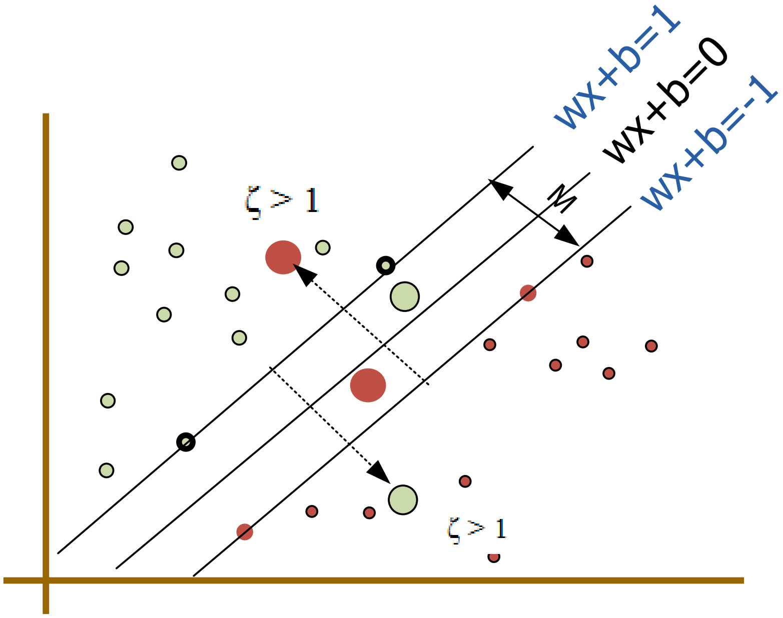
The optimization problem is therefore a convex quadratic programming problem which has global minimum.

2.2.6.2 Binary Non-Linearly Separable Case

In the non-linearly separable case, it is not possible to find a linear hyperplane that separates all positive and negative examples. To solve this case, the margin maximization technique may be relaxed by allowing some data points to fall on the wrong side of the margin, i.e., to allow a degree of error in the separation. ***Slack Variables*** $\boldsymbol{\xi_i}$ are introduced to represent the error degree for each input data point. Figure 2-8 demonstrates the non-linearly separable case where data points may fall into one of three possibilities:

1. Points falling outside the margin that are correctly classified, with $\xi_i = 0$
2. Points falling inside the margin that are still correctly classified, with $0 < \xi_i < 1$
3. Points falling outside the margin and are incorrectly classified, with $\xi_i = 1$

FIGURE 2-8 – SVM NON-LINEARLY SEPARABLE CASE



If all slack variables have a value of zero, the data is linearly separable. For the non-linearly separable case, some slack variables have nonzero values. The optimization goal in this case is to maximize the margin while minimizing the points with $\xi_i \neq 0$, i.e., to minimize the margin error.

$$\begin{aligned}
&\mathbf{w}.\mathbf{x}^c + b \geq +1 - \xi^c \quad \textit{for positive cases}\\
&\mathbf{w}.\mathbf{x}^c + b \leq -1 + \xi^c \quad \textit{for negative cases}\\
&\textit{with} \;\; \xi^c \geq 0 \quad \textit{for all } c\\
&\textit{and} \;\; \frac{\|\mathbf{w}\|^2}{2} + C\sum_c \xi^c \quad \textit{as small as possible}
\end{aligned}$$

In mathematical terms, the optimization goal becomes:

$$\text{find } \mathbf{w} \text{ and b that minimize:} \qquad \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i^{\,2}$$

$$\text{subject to: } \; y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i$$

where C is an user-defined parameter to enforce that all slack variables are as close to zero as possible. Finding the most appropriate choice for C will depend on the input data set in use.

As in the linearly separable problem, this optimization problem can be converted to its dual problem:

find $\boldsymbol{\alpha}$ that maximizes $$\sum_i \alpha_i - \tfrac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j \mathbf{x}_i{}^T \mathbf{x}_j$$

subject to $$\sum\nolimits_{i=1}^{N} \alpha_i y_i = 0,$$ $$0 \le \alpha_i \le C, \quad \forall i$$

In order to solve the non-linearly separable case, SVM introduces the use of a mapping function Φ: $R^M \rightarrow F$ to translate the non-linear input space into a higher dimension feature space where the data is linearly separable. **Error! Reference source ot found.** presents an example of the effect of mapping the nonlinear input space into a higher dimension linear feature space.

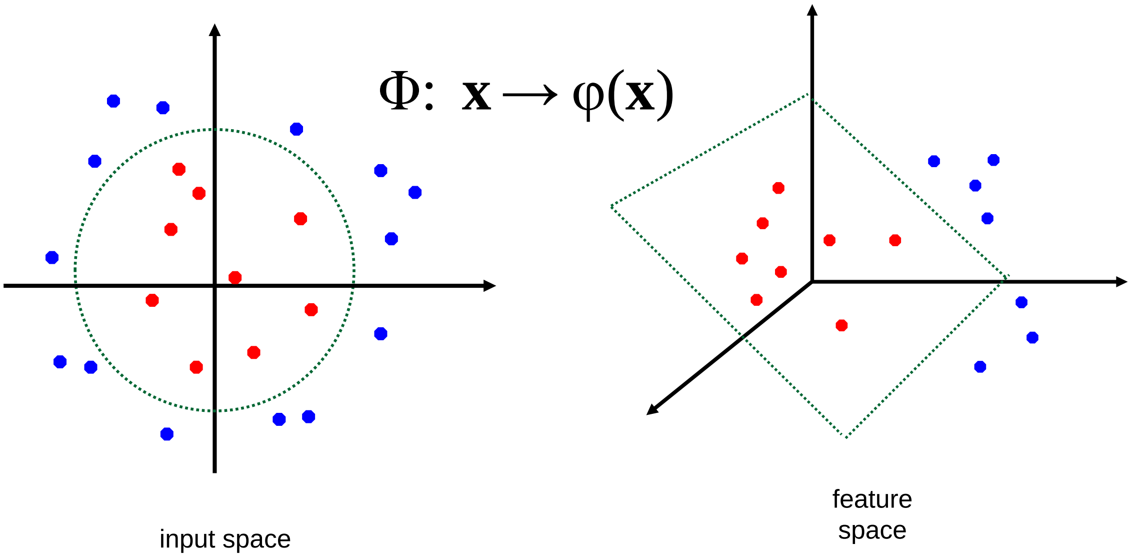


Figure 2-9 SVM mapping from input to feature space

The dual problem is solved in feature space where its aim becomes to:

find $\boldsymbol{\alpha}$ that maximizes
$$\sum_i \alpha_i - \tfrac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}_j)$$

subject to $$\sum\nolimits_{i=1}^{N} \alpha_i y_i = 0,$$ $$0 \le \alpha_i \le C, \quad \forall i$$

the resulting SVM is of the form:

$$f(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}_i) + b = \sum_{i=1}^{N} \alpha_i y_i \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}) + b$$

2.2.7 Pairwise Support Vector Machines

It's also possible to use other kernel functions to solve specific problems related to pairwise prediction, where the input is two instances and the output is the relationship between them. The input for pairwise Support Vector Machines is a pair of entities (a,b). As formalized by [Brunner et al., 2012], the binary Pairwise Support Vector Machine is: given a training data $((a_i, b_j), y_i)$, where y_i has binary values and the pair (a_i, b_i), is classified as +1 or -1), i=1,…,n, j=1,…,n and the mapping function $\emptyset$, then the Pairwise SVM method finds the optimal hyperplane:

$$w^T \emptyset(a_i, b_j) + \ b = 0$$

which separate the points in two categories. One of the solutions is based on the dual formalism of the optimization problem described in the previous sub-section. In this case the decision function is:

$$f((a_1, b_1) := \sum_{(i,j) \in I} \gamma_{ij} K(((a_i, b_i), (\, a_1, b_1)) + b$$

where $b \in \mathbb{R}\ and\ \gamma_{ij}\ \in \mathbb{R}\ for\ all\ (i, j) \in I.$

The construction of pairwise kernels K are based on simple kernels k [Roche-Lima et al.,2014]. Some examples of pairwise kernel functions are [Brunner et al., 2012]:

- symmetric direct sum pairwise kernel
$$K_{SD}((a, b), (c, d)) := \frac{1}{2}(\langle a, c\rangle + \langle a, d\rangle + \langle b, c\rangle + \langle b, d\rangle$$
- metric learning pairwise kernel
$$K_{ML}((a, b), (c, d)) := \frac{1}{4}(\langle a, c\rangle - \langle a, d\rangle - \langle b, c\rangle + \langle b, d\rangle)^2$$
- Tensor learning pairwise kernel
$$K_{TL}((a, b), (c, d)) := \frac{1}{2}(\langle a, c\rangle\langle b, d\rangle + \langle a, d\rangle\langle b, c\rangle$$
- asymmetric tensor pairwise kernel
$$K_{AT}((a, b), (c, d)) := \frac{1}{4}(\langle a, c\rangle\langle b, d\rangle - \langle a, d\rangle\langle b, c\rangle)^2$$

3 Framework of Event Similarity Measures

The focus of this chapter is on developing similarity measures for the main components of an event. We discuss in details similarity between event-types, similarity in time, similarity in location, similarity in participants . This chapter establishes the basis for next chapter which will discuss the algorithmic framework of how to combine the individual similarity measures developed in this chapter and how to use them in one architecture.

3.1 Event Type similarity

As discussed in the previous chapter, many of existing similarity measures are devoted to measure the similarity between concepts. In this section, we will test the adequacy of these measures for our task and discuss their pros and cons. An event-type (concept), taken from an observation description reflects the type from the perspective of the observer. An observer describes the event based on her partial knowledge of the situation. For some types of events like the meteorological events, a sequence of observations, if took place in some order and locations may signal a certain type of events. However, a single observation may represent only a local view of the global event.

To be able to reason about event-type similarity, we need to find the relation between each pair as shown in figure(3-1).

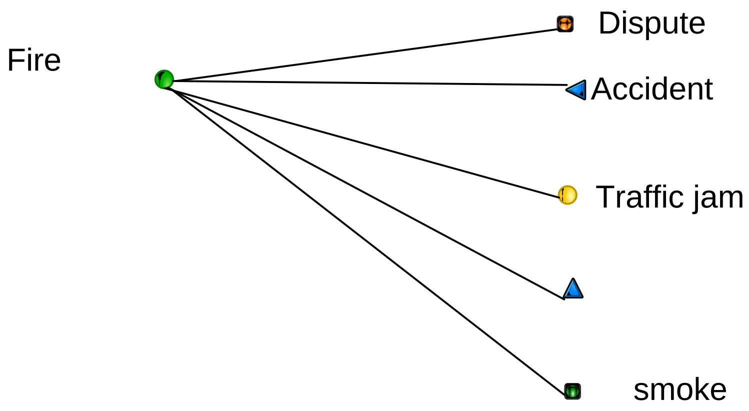


Figure 3-1 Sample of event types occuring in a time window

Having a pair of event-types, we simply can calculate the similarity using any of existing similarity measures listed in the previous chapter. Almost the majority of similarity measures use the lexical database ‘WordNet’ to calculate this similarity. In the next section, we elaborate more about the organization of events in WordNet before using the measure to do the calculation.

3.1.1 Events in WordNet

Understanding the organization of events in WordNet is crucial, because many similarity measures depend on this structure (path between concepts or path between each concept and their least common subsumer (LCS) to calculate similarity. WordNet has four Synsets for the term event which are used to describe physical phenomenon, psychological feature, circumstance and a phenomenon that is caused by some previous phenomenon [WordNet, 2010]. As shown in figure (3-3), all Sysnsets meet at classifying events as a sub-class from entity.

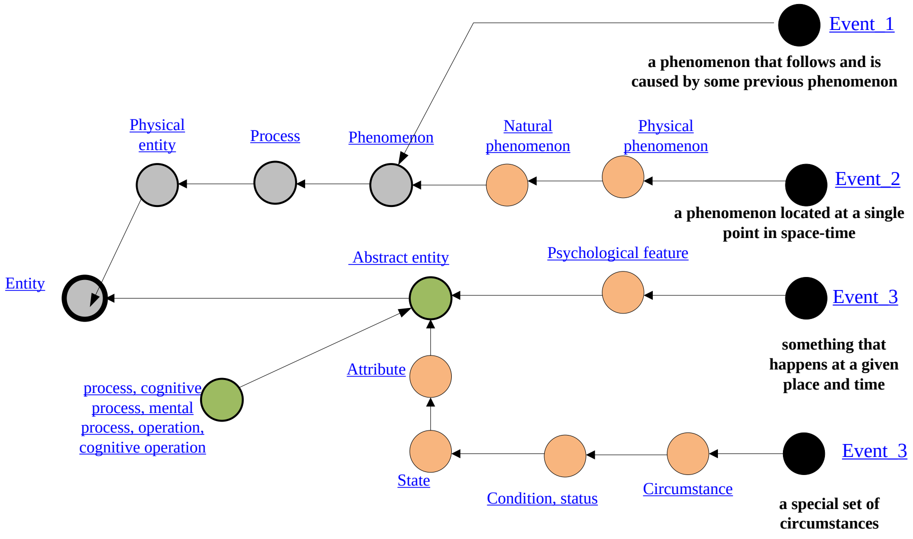

Event_1
a phenomenon that follows and is caused by some previous phenomenon
Physical entity
Process
Phenomenon
Natural phenomenon
Physical phenomenon
Event_2
a phenomenon located at a single point in space-time
Entity
Abstract entity
Psychological feature
Event_3
something that happens at a given place and time
process, cognitive process, mental process, operation, cognitive operation
Attribute
State
Condition, status
Circumstance
Event_3
a special set of circumstances

Figure 3-2 Event Synsets in WordNet

WordNet classifies for example ‘storm’ under natural phenomenon (all phenomena that are not artificial) to distinguish them from psychological features, which is in turn a distinguishing between physical and abstract entities. However, a less clear distinction is given to different types of events that are performed or caused by humans which are acts that are further classified as an activity, or an action or a process. Figure (3-4) illustrates different examples that come under the Synset (Event: something that happens at a given place and time).

Under the “act” Synset (something that people do or cause to happen), WordNet groups different types of events. A “Piracy” event is classified as an “Activity”, a “knife fight” is classified as “group action”, the “looting” event is classified as an “action”. Procedures done by human beings like “ calculating, fingerprinting and experimental procedure” are grouped under “Activity”. In WordNet activities, actions, group actions and procedures are grouped together because they are a kind of psychological feature that arouses an organism to action toward a desired goal. One may justify this based on considering that they have a reason for the action which gives a purpose and a direction to behavior.

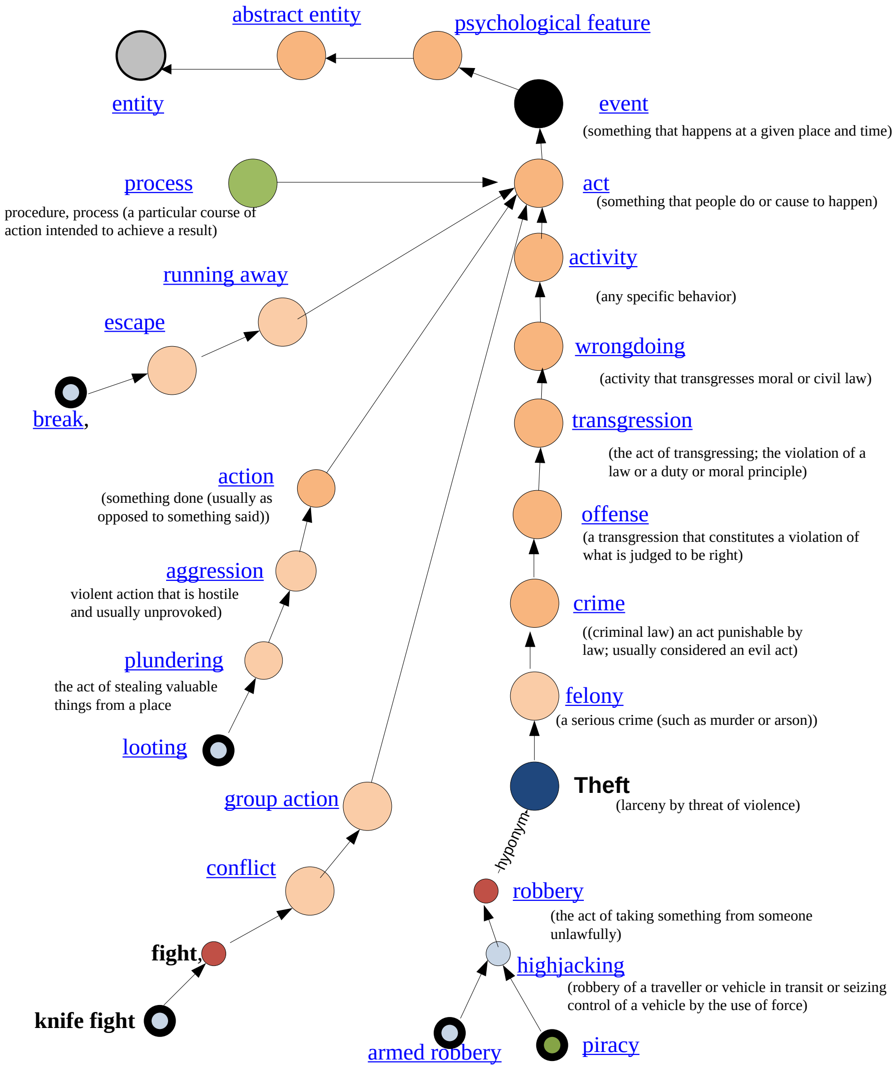

abstract entity
psychological feature
entity
event
(something that happens at a given place and time)
process
procedure, process (a particular course of action intended to achieve a result)
act
(something that people do or cause to happen)
activity
(any specific behavior)
running away
escape
break,
wrongdoing
(activity that transgresses moral or civil law)
transgression
(the act of transgressing; the violation of a law or a duty or moral principle)
action
(something done (usually as opposed to something said))
offense
(a transgression that constitutes a violation of what is judged to be right)
aggression
violent action that is hostile and usually unprovoked)
crime
((criminal law) an act punishable by law; usually considered an evil act)
plundering
the act of stealing valuable things from a place
felony
(a serious crime (such as murder or arson))
looting
Theft
(larceny by threat of violence)
group action
hyponym
conflict
robbery
(the act of taking something from someone unlawfully)
fight,
highjacking
(robbery of a traveller or vehicle in transit or seizing control of a vehicle by the use of force)
knife fight
armed robbery
piracy

Figure 3-3 Different types of human events

We can conclude that some concepts that are cognitively recognized as events, such as weather events, are not classified as events in WordNet. Also some concepts like ‘doctor’ is classified as an event, when it refers to a child's play where children take the roles of physician or patient or nurse and pretend they are at the physician's office.

3.1.2 Analysis of similarity measures

With the variety of given similarity measures, different approaches and strategies are adopted to evaluate which measure captures most the similarity between two objects or two concepts. As indicated by [Lin, 1997], the problem with similarity measures is that each of them is tied to a particular application or assumes a particular domain model. Therefore, Lin proposed a theoretical examination of the properties of the similarity measure, and whether they comply with the similarity intuitions he proposed. Resnik (1995), for example, compared the results with human judgments. Other researchers, examine the fitness of the similarity measure based on the domain or application needs. This approach is widely used in biological research [Lord et al, 2003]. Measuring the similarity or distance between concepts is based on measuring the semantic similarity or semantic relatedness between two words. The difference between semantic similarity and semantic relatedness is explained by is [Resnik,1995] as "Semantic similarity represents a special case of semantic relatedness: for example, cars and gasoline would seem to be more closely related than, say, cars and bicycles, but the latter pair are certainly more similar". While similarity only considers subsumption relations to assess how two objects are alike, relatedness takes into account a broader range of relations (e.g., part-of).

To motivate our need to re-evaluate existing semantic measures, we took different pairs of events and lookup their semantic scores using different existing measures listed in chapter 2. The results are shown in Table (3-1).

Example # 1: Semantic similarity

In the first example, we consider only concepts having subsumption realtions as shown in figure (3-4)

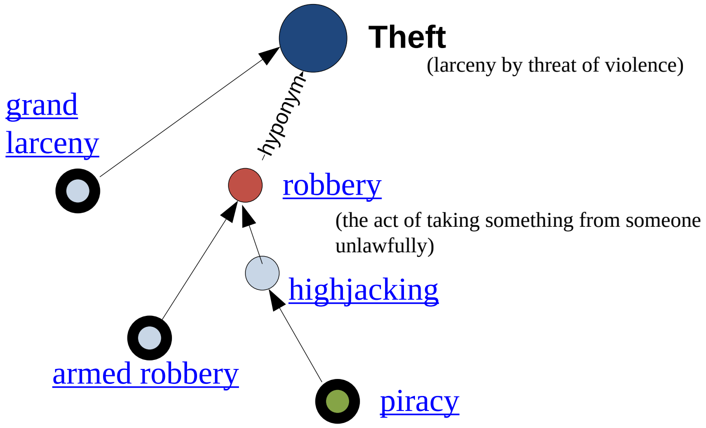

Theft
(larceny by threat of violence)
hyponym
grand larceny
robbery
(the act of taking something from someone unlawfully)
highjacking
armed robbery
piracy

Figure 3-4 Example of subsumptions relations

The results of similarity measures between a pair of concepts is shown in table (3-1) and figure (3-5).

Table 3-1 Similarity measures for a set of pairs with subsumption relations

Pair	wup	jcn	lch	lin	res	path	lesk	hso
theft#n#1, robbery#n#1 w1>w2	0.9600	0.9885	2.9957	0.9488	9.3679	0.5000	155	4
robbery#n#1, highjacking#n#1 w2>w1	0.9630	0.0000	2.9957	0.0000	10.3795	0.5000	410	4
highjacking#n#1, piracy#n#1 w2>w3	0.9655	0.0000	2.9957	0.0000	0.0000	0.5000	1059	4
robbery#n#1, armed_robbery#n#1 w2>w5	0.9630	0.9885	2.9957	0.9488	9.3679	0.5000	155	4

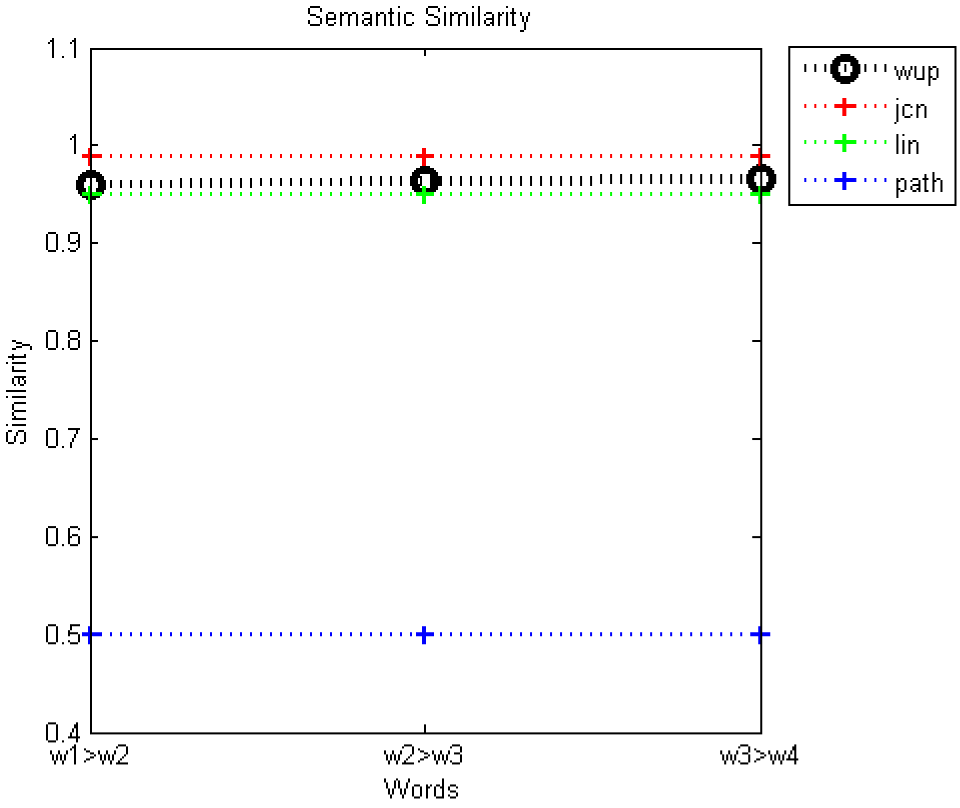


Figure 3-5 Comparison of similarity measures for a set of pairs with subsumption relations

To calculate the similarity measures, we used the online library WS4J (WordNet Similarity for Java) which measures the semantic similarity/relatedness between words.

Example # 2. Semantic relatedness

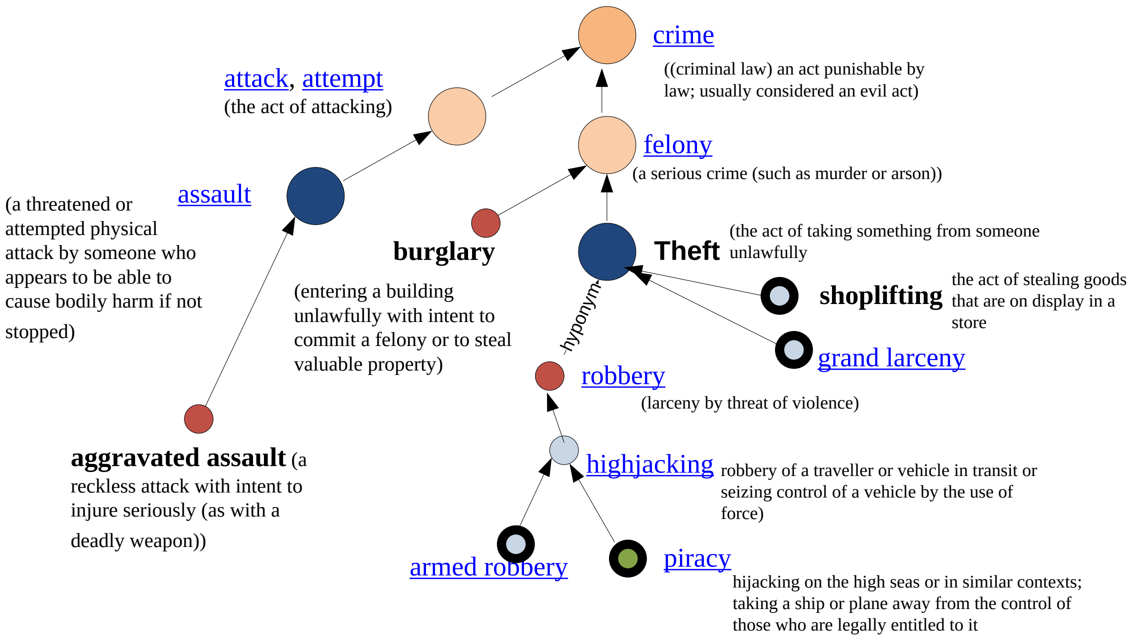
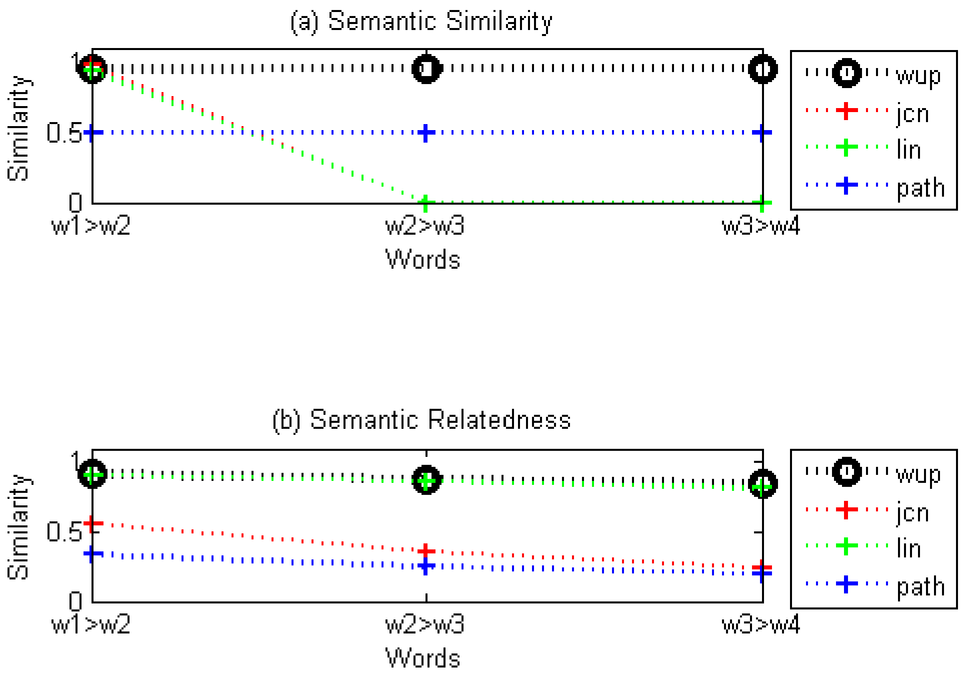

crime
((criminal law) an act punishable by law; usually considered an evil act)
attack, attempt
(the act of attacking)
felony
(a serious crime (such as murder or arson))
assault
(a threatened or attempted physical attack by someone who appears to be able to cause bodily harm if not stopped)
burglary
(entering a building unlawfully with intent to commit a felony or to steal valuable property)
Theft
(the act of taking something from someone unlawfully
hyponym
shoplifting
the act of stealing goods that are on display in a store
grand larceny
robbery
(larceny by threat of violence)
highjacking
robbery of a traveller or vehicle in transit or seizing control of a vehicle by the use of force)
aggravated assault (a reckless attack with intent to injure seriously (as with a deadly weapon))
armed robbery
piracy
hijacking on the high seas or in similar contexts; taking a ship or plane away from the control of those who are legally entitled to it

Figure 3-6 Example of semantic relatedness relations

Pair	wup	jcn	lch	lin	res	path	lesk	hso
theft#n#1, burglary#n#1	0.9167	0.5613	2.5903	0.9111	9.1267	0.3333	78	5
robbery#n#1, burglary#n#1	0.8800	0.3580	2.3026	0.8673	9.1267	0.2500	7	4
burglary#n#1, armed_robbery#n#1	0.8462	0.2393	2.0794	0.8137	9.1267	0.2000	0	3
robbery#n#1, aggravated_assault#n#1	0.7692	0.0000	1.7430	0.0000	7.6882	0.1429	7	0

As shown in Figure


(a) Semantic Similarity

(b) Semantic Relatedness

Discussion

Based on the calculation shown in table (3-1) and table (3-2), we notice the following:

1. In jiang (jcn), Leakcock and Chodorow (lch), lin and resnik (res) the similarity measure in the is-a hierarchy is the same for different pairs as shown by the similarity of the two pairs:

   (theft#n#1, robbery#n#1) is the same as (robbery#n#1, armed_robbery#n#1) ;

2. It is hard to explain the high similarity and its meaning in **Wu & Palmer's (wup)**.
3. The path similarity is constant for all pairs of the same length between their nodes
4. The zero value in similarity measures using information content doesn't mean zero similarity, rather it means that the information content which is calculated based on the frequency of the word in the corpus is not found. The information content measure relies on corpora analysis, and sparse data problem is not avoided
5. For the same pairs there is a big difference in the values between different measures, if we take the normalized measures only, those giving values in [0,1] as an example, will produce values ranging from 0.333 to 0.91:

Pair	wup	jcn	lin	path
theft#n#1, burglary#n#1	0.9167	0.5613	0.9111	0.3333

If we take the average, it seems a bad idea specially when the value of the measure equals "0". Also taking the highest value may mislead the comparison as some measures are creating higher similarities above the average of all other similarity measures.

Also as indicated by [Budanitsky and Hirst, 2006] evaluating WordNet-based measures of lexical semantic relatedness, results show considerable differences in the performance of proposed measures. For measures depending on information content, as mentioned by [Wang et al., 2004] [Lee et al., 2008] may be inaccurate due to shallow annotations.

There are alternative methods to measure the similarity between concepts. One method is to use the features based model to calculate similarity. In the following section, we represent a method based on lattice approach. Both feature based and lattice based similarity suffer from computational complexity. If features are not available, similarity computation is not applicable.

3.1.3 Event Lattice

First we examined the components that are considered by a human to assign a type to an event. Working with a group of experts, who are working on daily basis with different types of events we find that representing an event using a lattice better help users to differentiate between one type from another. The main intuition behind using an event lattice is to explicitly describe the overlap between different concepts. As shown in Figure (3-8)

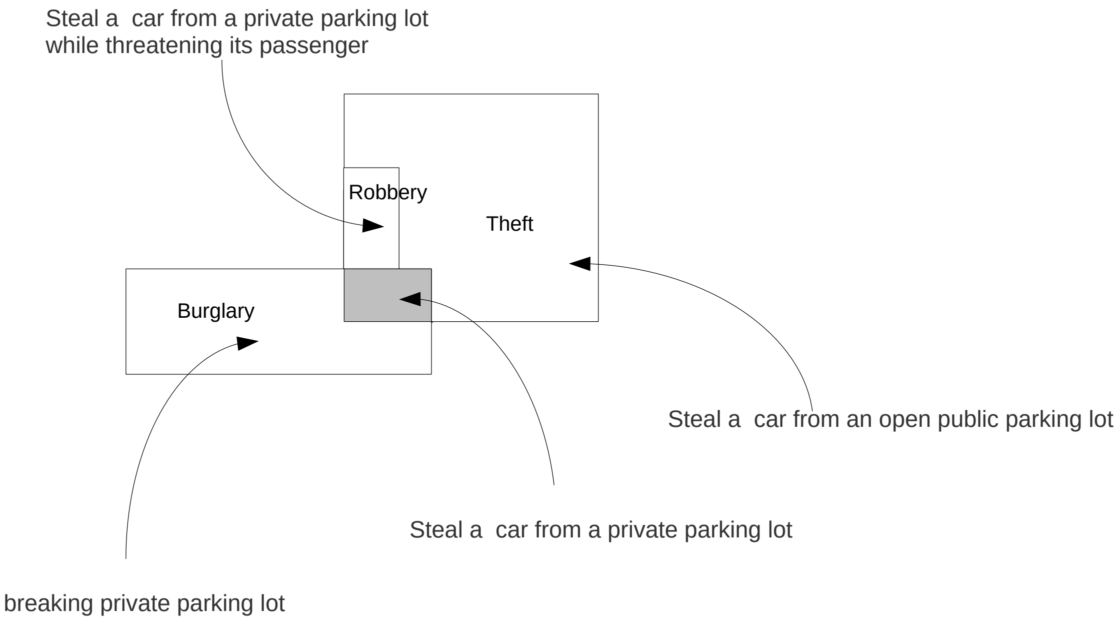
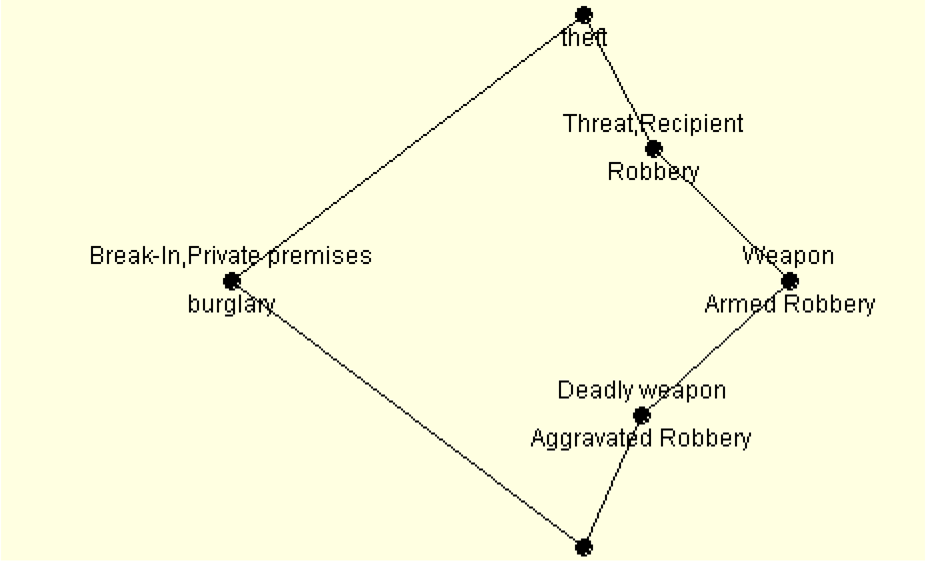

Steal a car from a private parking lot
while threatening its passenger
Robbery
Theft
Burglary
Steal a car from an open public parking lot
Steal a car from a private parking lot
breaking private parking lot

Figure 3-7 Example showing overlap between concepts

To systematically identify the overlap between the three concepts, we refer to the concepts depicted in figure (3-6) mainly {theft,burglary,robbery}. A lattice for these concepts is given in figure (3-8).



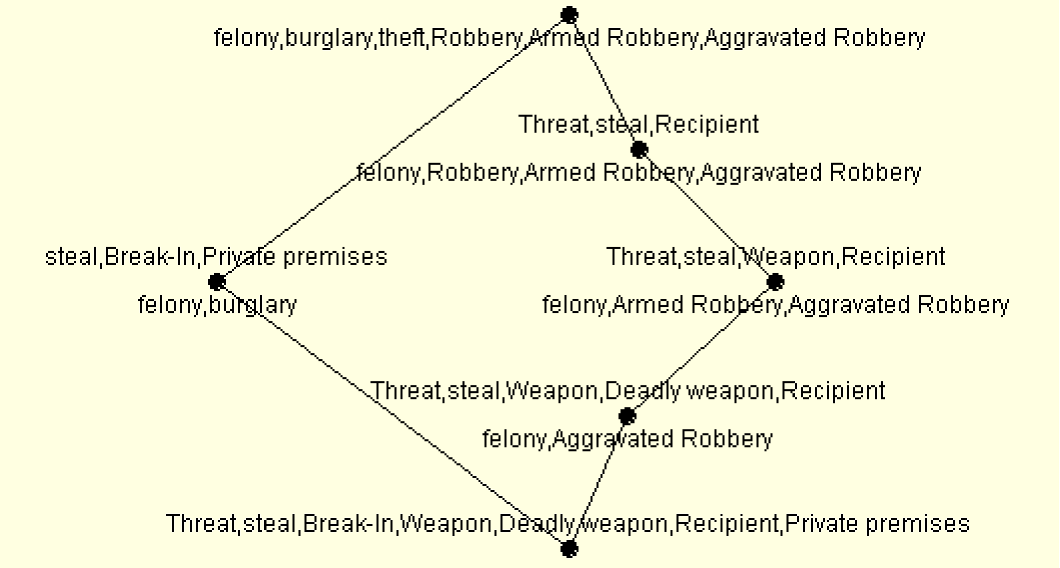

felony,burglary,theft,Robbery,Armed Robbery,Aggravated Robbery
Threat,steal,Recipient
felony,Robbery,Armed Robbery,Aggravated Robbery
steal,Break-In,Private premises
felony,burglary
Threat,steal,Weapon,Recipient
felony,Armed Robbery,Aggravated Robbery
Threat,steal,Weapon,Deadly weapon,Recipient
felony,Aggravated Robbery
Threat,steal,Break-In,Weapon,Deadly weapon,Recipient,Private premises

Figure 3-8 Concept lattice of different events from the felony domain

The lattice is built on the theory of Formal Concept Analysis [Wille, 2005]. In Formal Concept Analysis (FCA), a concept C is determined by its extent and intent.

Definition 3.1 (Extent). is the set of all objects that belong to a concept.

Definition 3.2 (Intent). is the set of all attributes shared by the objects in a concept.

Definition 3.3 (Concept Context). A concept of the context (G,M, I) is a pair C = (A,B) with A ⊆ G,B ⊆ M, such that A′ = B and B′ = A. We call A the extent and B the intent of the concept (A,B). B(G,M, I) denotes the set of all concepts of the context K = (G,M, I). We assume that $G \cap M = \phi$. [Ganter and Wille, 1999]

Concept ***(A_1, B_1)*** is more general than concept (A_2, B_2) if and only if the extent A_1 contains A_2:

$$(A_1, B_1) \geq (A_2, B_2) \Leftrightarrow A_1 \supseteq A_2$$

This is equivalent to the intent B_1 contains B_2

$$(A_1, B_1) \geq (A_2, B_2) \Leftrightarrow B_2 \supseteq B_1$$

The relation ≤ is called the hierarchical order of the concepts

A context may be depicted as a |G| × |M| a binary matrix where the concepts of G form column labels and the objects of M form row labels [Alqadah and Bhatnagar, 2011]. In order to express that an object g is in a relation I with an attribute m, we write gIm or (g, m)∈ I, and read it as “the abject g has the attribute m”.

Let mat(𝕂) denote the matrix representation of 𝕂, then we may specify the entries of the matrix as

$$mat\ (\mathbb{K}) = \begin{cases} 1 & if\ \boldsymbol{g_i I\, m_i} \\ 0 & otherwise \end{cases}$$

Example 1. An event context can be represented by a *cross table*, such as *in table (1)*. It represents a formal context **K =** (G,M,I),where G={Burglary, Theft, Robbery, Armed Robbery, Aggravated Robbery} and M ={threat,steal,break-in,weapon, Deadly weapon,Recipient},and a “1” in row g∈G and column m∈M means that the object g has the attribute m.

Table 3-2 Extent and Intent of sample of events

	Private premises m1	***Threat m2***	***Steal m3***	***Break-In m4***	***Weapon m5***	***Deadly weapon m6***	***Recipient m7***
Burglary (g1)	*1*	*0*	*1*	*1*	*0*	*0*	*0*
Theft (g2)	*0*	*0*	*1*	*0*	*0*	*0*	*0*
Robbery(g3)	*0*	*1*	*1*		*0*	*0*	*1*
Armed Robbery(g4)	*0*	*1*	*1*	*0*	*1*	*0*	*1*
Aggravated Robbery(g5)	*0*	*1*	*1*	*0*	*1*	*1*	*1*
High jacking(g6)	*0*	*1*	*1*	*0*	*0*	*0*	*1*

1	<{felony}, {Threat,steal,Break-In,Weapon,Deadly weapon,Recipient,Private premises}>
2	<{felony,burglary}, {steal,Break-In}>
3	<{felony,burglary,theft,Robbery,Armed Robbery,Aggravated Robbery}, {steal, Recipient }>
4	<{felony,Robbery,Armed Robbery,Aggravated Robbery}, {Threat,steal,Recipient}>

5	<{felony,Armed Robbery,Aggravated Robbery}, {Threat,steal,Weapon,Recipient}>
6	<{felony,Aggravated Robbery}, {Threat,steal,Weapon,Deadly weapon,Recipient}>

Furthermore, given the set of attributes of different observations, the concept lattice is used to select the best matching type from the lattice. However, the selected type must satisfy the condition of being the largest lower bound of the selected concept (event type).

Example #2 Storm Events < Natural Events >

Object feature	Cyclone	Typhoon	Hurricane	Tornado
Strong wind	1	1	1	1
Violent wind	1	1	1	1
Hail	0	0	0	1
Ice	0	0	0	0
Rain	1	1	1	1
Snowfall	0	0	0	0
Heavy rain	1	1	1	1
Thunder	1	1	1	1
Lightning	1	1	1	1
Dust	0	0	0	1
Sand	0	0	0	1
Special shape	1	1	1	1
Overland	0	0	0	1
Overseas	1	1	1	0
Short duration	0	0	0	1
Cause high waves	1	1	1	0
local	0	0	0	1
Harmful	1	1	1	1
Cause injury	1	1	1	1
Cause death	1	1	1	1
Cause property damage	1	1	1	1
Cause flood	1	1	1	0
Storm surge	1	1	1	0
Tropical	1	1	1	0
Cause sleet	0	0	0	1

Formal Concepts of "Meteorological events"	
1	<{Cyclone,Typhoon,Hurricane}, {flood,storm surge,Strong wind,Rain,Heavy rain,Thunder,Special shape,tropical,Overseas,high waves,Harmful,Cause injury,Cause death,Cause property damage}>
2	<{Tornado}, {Violent wind,Hail,sleet,Rain,Heavy rain,Thunder,Dust,Special shape,Overland,Harmful,Cause injury,Cause death,Cause property damage}>
3	<{Cyclone,Typhoon,Hurricane,Tornado}, {Rain,Heavy rain,Thunder,Special shape,Harmful,Cause injury,Cause death,Cause property damage}>
4	<{}, {flood,storm surge,Strong wind,Violent wind,Hail,sleet,Rain,Heavy rain,Thunder,Lightning,Dust,Special shape,tropical,Overland,Overseas,high waves,Harmful,Cause injury,Cause death,Cause property damage}>

3.1.4 A lattice based similarity measure

A formal concept consists of two sets; therefore we can use set-based similarity to measure the similarity between two concept $\boldsymbol{c_1}, c_2$. [Blachon et al., 2007] measured the distance between two formal concepts using the following formula

$$d_{ij} = \frac{1}{2}\frac{\left|X_i \Delta\, X_j\right|}{\left|X_i \cup X_j\right|} + \frac{1}{2}\frac{\left|T_i \Delta\, T_j\right|}{\left|T_i \cup T_j\right|}$$

Where Δ is the symmetrical set difference between $S_i \; and \; S_j = \left|\frac{S_i \cup S_j}{S_i \cap S_j}\right|$

The symmetrical difference can be calculated using :

$$\text{Jaccard index } S_{jac} = \frac{|x \cap y|}{|x \cup y|}$$

$$\text{Dice's coefficient } S_{sor} = \frac{2*|x \cap y|}{|x|+|y|}$$

$$\text{Symmetric difference } S_{\ominus} = 1 - \frac{|x \ominus y|}{|x \cup y|}$$

$$\text{where } x \ominus y = \frac{x}{y} \cup \frac{y}{x}$$

[Formica, 2007] used an information content approach to calculate concept-based similarity . For a two concepts $(\boldsymbol{A_1}, B_1) \; and \; (\boldsymbol{A_2}, B_2)$

$$Sim((\boldsymbol{A_1}, B_1), (\boldsymbol{A_2}, B_2)) = \frac{|\boldsymbol{A_1} \cap \boldsymbol{A_2}|}{r} * w + \frac{\mathrm{M}(B_1, B_2)}{m} * (1 - w)$$

Where $\mathrm{M}(\mathrm{B}_1, \mathrm{B}_2)$ is defined as maximum sum of information content ics of pairs, within all possible candidate sets of pairs such that there are no two pairs in the set sharing an element, and r,m are the greatest between the cardinalities of the sets $\mathrm{A}_1, \mathrm{A}_2$ and B_1, B_2

To calculate the similarity between two concepts we define the following general measure:

$$Sim(C_1, C_2) = w * Sim(A_1, A_2) + (1 - w) Sim(B_1, B_2)$$

Where $0 \leq w \leq 1$ and Sim is one of the similarity measures listed above. However, it is also possible to measure the similariy by using the intents of the objects alone. This a plausible approach since it only depends on measuring the distnace between the two concepts by using the difference between their attributes only. This could be done by setting w="0" and is equivalent to:

```
Input:    Concepts            $C_1, C_2$
          Context             (G,M, I)
Output:   Similarity degree   $Sim(C_1, C_2)$

begin
    $sim \leftarrow 0$
    K ← (G, M, I)
    If $C_2 = C_1$ then
        $Sim(C_1, C_2) \leftarrow 1$
    else If $C_1 < C_2$ then
        $Sim(C_1, C_2) \leftarrow \frac{|intent(A_1|)}{|intent(A_2)|}$
        else if $C_2 < C_1$ then
            $Sim(C_1, C_2) \leftarrow \frac{|intent(A_2)|}{|intent(A_1)|}$
        else
            $\boldsymbol{Sim(C_1, C_2)} \leftarrow \frac{\boldsymbol{2 * |intent(A_1)| \cap |intent(A_2)|)}}{\boldsymbol{|intent(A_1) \cup intent(A_2)|)}}$
        End else
        return $\mathrm{Sim}(\mathrm{C}_1, \mathrm{C}_2)$
end
```

Example #1 Theft and robbery

{theft }, { steal }
{Robbery }, { steal, threat }

$$\text{Sim (theft,Robbery)} = \frac{|\{steal\}|}{|\{steal, threat\}|} = \frac{1}{2} = 0.5$$

Example #2 Robbery and Armed Robbery

{Robbery}, { Steal, threat }
{Armed Robbery}, { Steal, threat, weapon}

Sim(Robbery, Armed Robbery) = 2/3 = 0.666

Pair	lattice	wup	jcn	lch	lin	res	path	lesk	hso
theft#n#1, robbery#n#1 w1>w2	0.5	0.9600	0.9885	2.9957	0.9488	9.3679	0.5000	155	4
robbery#n#1, armed_robbery#n#1 w2>w5	0.666	0.9630	0.9885	2.9957	0.9488	9.3679	0.5000	155	4

3.2 Location Similarity

Comparing the location of two events is not always a straight forward, especially when events are reported using natural language. Different qualitative spatial relations are used to express the location of an event with other spatial entities. For the orientation aspect, events are described using qualitative terms such as "north of", "in front of", "behind", etc. Many approaches and calculi have been used to express the orientation of one object on reference to another. Most approaches use points as the basic spatial entities and use different versions of jointly exhaustive and pairwise distinct (JEPD) orientation relations. Distance qualitative relations are also used when describing the location of events. For the distance aspect, terms such as "near", "far", "close to" are commonly used. As mentioned by [Renz and Nebel, 2007] combining the orientation and distance aspects is called positional information.

In this work, we use the Region Connection Calculus (RCC8) theory to describe the spatial relation between two locations. A location is defined as an inherently grounded spatial entity, a location includes geospatial entities such as countries, mountains, cities, rivers, etc. It also includes classificatory and ontological spatial terms, such as edge, corner, intersection [Pustejovsk, 2011]. The location element covers both locations and places (where a place is considered a functional category), and is assumed to be associated with a region whenever appropriate [David et al, 1992]

3.2.1 RCC8 relations

There are different aspects of space related to describing the event location on reference to another object. The location of an event could be expressed using a combination of orientation relations, distance relations and topological relations. While orientation and distance relations are important, in this thesis we focus mainly on topological relations. Topology in mathematics concerned with the most basic properties of space, such as connectedness, continuity and boundary, while in qualitative spatial reasoning, the focus is on mereotoplogy [Cohn and Renz, 2008].

In the Region Connection Calculus, regions are the basic spatial entities and relationships between spatial regions are defined in terms of the binary relation C(x; y), meaning spatial entity x connects with spatial entity y, which is true if and only if the closure of region x is connected to the closure of region y, i.e. if their closures share a common point [Renz et al, 2007]. Using the relation C, many versions of RCC could be found for instance RCC1, RCC2, RCC3, RCC5, RCC8, RCC15, and RCC23. The most common used and researched version is RCC8, which defines the following eight Jointly Exhaustive and Pairwise Disjoint (JEPD) relations: disconnected (DC), externally connected (EC), partially overlaps (PO), equal (EQ), tangential proper part (TPP), nontangential proper part (NTPP), tangential proper part inverse (TPPi) and nontangential proper part inverse (NTPPi) [8]. The intended meaning of these relations is illustrated in table (3-3).

Table 3-3 DEFINING RCC8 RELATIONS

Name	Symbol	Relation	Meaning
Equals	EQ	EQ(x,y)	X is identical with y
Disconnected	DC	DC(x,y)	X is disconnected from

Externally Connected	EC	EC(x,y)	X is externally connected to y
Partially Overlap	PO	PO(x,y)	X partially overlaps y
Tangential Proper Part	TPP	TPP(x,y)	X is tangential proper part of y
Non-Tangential Proper Part	NTPP	NTPP(x,y)	X is non-tangential proper part of y

Adapted from [David et al., 1992] [Renz, 1998]

The formal definition of the relations in table (3-3) is :

- $DC(x,y) \equiv \neg\, C(x,y)$
- $P(x,y) \equiv_{def} \forall\, z[\, C(z,x) \rightarrow C(z,y)]$
- $PP(x,y) \equiv_{def} P(x,y) \ \wedge \neg\, P(y,x)$
- $O(x,y) \equiv_{def} \exists z\ [P(z,x) \ \wedge P(z,y)]$
- $EC(x,y) \equiv_{def} C(x,y) \ \wedge \neg\, O(x,y)$
- $PO(x,y) \equiv_{def} O(x,y) \wedge \neg\, P(x,y) \ \wedge \neg\, P(y,x)$
- $TPP(x,y) \equiv_{def} PP(x,y) \ \wedge \exists z[EC(z,x) \ \wedge EC(z,y)]$
- $NTTP(x,y) \equiv_{def} PP(x,y) \ \wedge \neg\, \exists z[EC(z,x) \ \wedge EC(z,y)]$

3.2.2 Reasoning using RCC8 Relations

Since events are spatio-temporal entities, it is natural to use spatio-temporal reasoning to reason about the location of events. Studying how people report about the location of events, we notice that qualitative knowledge is used to express the event location as could be seen from the following example:

Event 1: 8 Palestinians are arrested across the West Bank

Event 2: Thursday eight Palestinians arrested from Jerusalem, Jenin and Hebron, according to local and security sources.

In these two events, the event location is expressed using different qualitative representations which are used with different levels of granularity and expressiveness. When performing reasoning about the location of the two events, we may need to know if West Bank contains Jerusalem, Jenin and Hebron. Other aspects of event locations are usually described qualitatively, such as distance, orientation and topology.

Furthermore, There are many places that share the same or similar names (“AL-Tireh”:a neighborhood in Ramallah city;“AL-Tireh”: a Village in Ramallah region and “AL-

Tireh": a village north of Jenin city). Also some places have multiple names(e.g. AL-Manarah square is also called Lions square). Some places are called after the most famous point of interest found near that place.

With RCC we can reason if two events have the same location by using the connection relations as explained in the following rules:

Disconnected: Since one event cannot take place into two separate locations, and we have two events with disconnected locations, we can deduce that these are two different events .

$$\frac{(e_1 \; in \; x) \wedge (e_2 \; in \; y) \wedge (\boldsymbol{DC}(x, y))}{e_1 \; \not\equiv \; e_2}$$

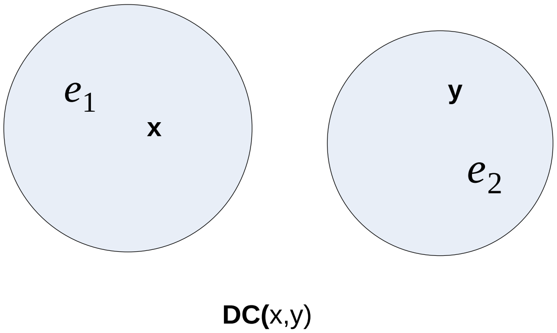
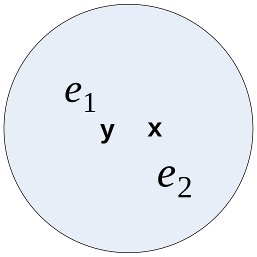


Figure 3-9 Disconnected regions

Equal: If the two regions are equal, then at least one condition is met in the matching criteria, therefore it is possible that these two events are matched.

$$\frac{(e_1 \; in \; x) \wedge (e_2 \; in \; y) \wedge \big(\boldsymbol{EQ}(x, y)\big)}{e_1 \; \cong \; e_2}$$



EQ(x,y)

Figure 3-10 Equal regions

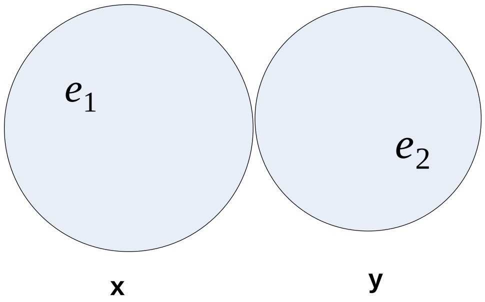
As in the following two events, if we know that Radio street and Al-Ersal street are the same street from our knowledge base then this condition is met.

Event 3: On 11 May 13, 11:14 hrs, reportedly, a car accident was reported in Radio street

Event 4: On 11 May 13, 11:18 hrs, a car accident was reported in Al-Ersal street

Externally Connected: with externally connected regions, there is a possibility that the two events are taking place at the border of these two regions, therefore it possible that these two events have equal location and therefore a possible match.

$$\frac{(e_1\ in\ x) \wedge (e_2\ in\ y) \wedge \left(\boldsymbol{EC}(x,y)\right)}{e_1\ \cong\ e_2}$$



EC(x,y)

Figure 3-11 3 Externally Connected regions

Event 1: On 31 Mar 13, 0930 hrs, approximately 40 people demonstrated at DCO Beit-EL, NE Ramallah. It ended peacefully at1440 hrs.

Event 2: On 31 Mar 13, between 0945-1200 hrs, families protested near City Inn Hotel, NE Ramallah against prisoners conditions.

Non tangential proper part: The semantic of the non tangential proper part is that region R1 is totally inside region R2 and that they are not equal and do not share any border.

$$\frac{(e_1 \; in \; x) \wedge (e_2 \; in \; y) \wedge \left(\boldsymbol{NTTP}(x, y)\right)}{e_1 \; \cong \; e_2}$$

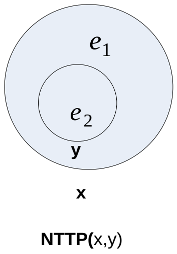


Figure 3-12 Non tangential proper part region

Event 1: A house in fire, in Jaffa street, second floor, near store AL-Manara close to AL-Families park

Event 2: A smoke is seen,near supermarket AL-Manara in Ain-Munjid area.

In these two events, Al-Families park is located in Ain-Munjid area.

Tangential proper part: in TTP relations, there might be more than two regions involved in the event. If x,y, and z are regions then y and x might be connected through a TPP, also y and z might be connected through a TPP**.**

$$\frac{(e_1 \; in \; x) \wedge (e_2 \; in \; y) \wedge \left(\boldsymbol{TTP}(x, y)\right)}{e_1 \; \cong \; e_2}$$

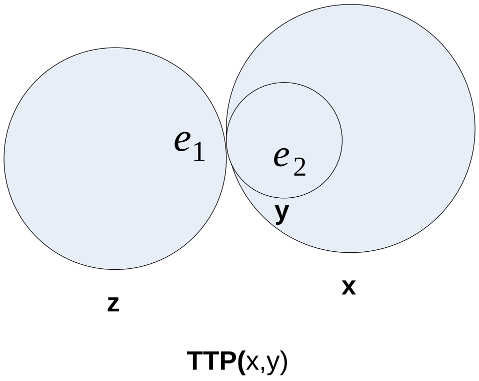


Figure 3-13 EC and TPP regions

Event 1: A teen is injured in clashes near Jerusalem

Event 2: A 17-year-old student was injured Thursday morning during clashes in the town of Abu Dis.

A main advantage for using RCC to reason about the location of events is that as examined by [Knau and Renz, 1997], RCC is structurally similar to the way people reason about space and is a model of people's conceptual knowledge of spatial relationships.

3.2.3 Space Ontology

In this thesis all the examples are taken from events located in populated places. A populated place is an area of land inhabited by people [Giunchiglia et al., 2010]. Therefore cities, villages, hamlets, towns, townships etc. are type of populated places. By definition, what mainly characterize an entity from another is its area. It is common to find the following definitions: a village is small human settlement, or a city is a large settlement and a hamlet is just a few dwellings [vocab.org]. Location and regions are more important for our work, however places are sometimes used to describe a region by its functional place like “city center”. A city center is a circle on a map that indicates the center of the city and it is only perceived by the human mind.

We have noticed that the three themes of geography (location, place and region) are used to describe where an event occurred or is happening. An observer uses relative location to describe the event when the observer is not familiar with the area. Also absolute locations are used when the observer knows the address of the event. Functional locations such as ‘city center’ or formal name such as ‘ name of the city’, or vernacular region such as ‘at the south area of the city’ are all used to describe an event location.

To model our regions, we use a region ontology where the country regions are classified into populated places and administratively declared places as shown in Figure (3-15)

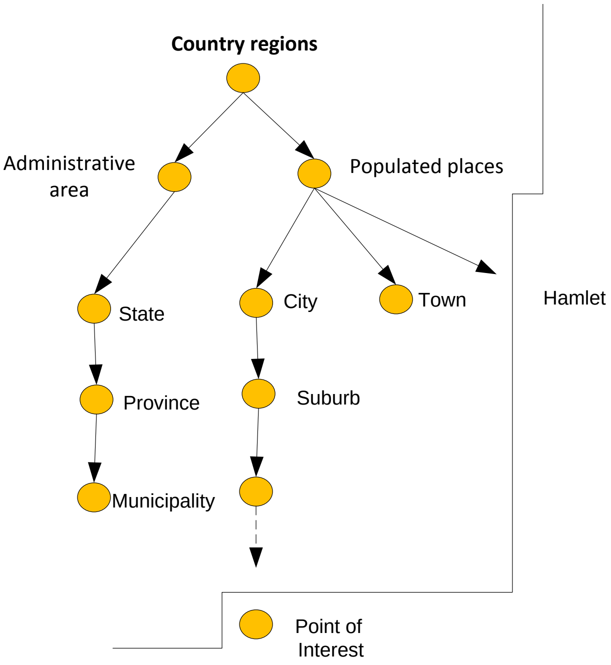


Figure 3-14 Classification of a Country Region

Populated places are classified into extended entities such as city and non-extended entities such a point-of-interest. All populated places are disjoint classes and are continuous and have no holes. Suburbs and neighborhoods are part of a larger entity and is represented in one of the following forms:

NTPP: a suburb(S) has an NTTP relation with a town (T) if a suburb lies in a town and shares no border with it. The relation is denoted by NTTP(S,T)

TPP: a suburb(S) has a TTP relation with a town (T) if a suburb lies in a town and shares borders with it. The relation is denoted by TTP(S,T)

EC and **DC**, these relations hold between suburb of a larger entity such as a city or town.

3.2.4 Building RCC Algorithm

In this section we demonstrate how RCC relations between geographical spaces could be calculated automatically. The following five topological relations between locations are built: (1) Equal (2) Externally Connected (3) Disconnected (4) Tangential Proper Part, and (5) Non-Tangential Proper Part. In this work, and by using a dataset of a country we build RCC relations between cities, towns, villages, suburbs and points of interests. For this purpose, we use an approximation technique to represent a region as circular shape. Furthermore, we represent a country map from circular tiles. The radius of the circle is calculated based on the type of the region being a city or a hamlet as an example. Other parameters are also considered if available such as the area and population of a region.

The proposed methodology for calculating RCC between geographical regions is to approximate the exact region tiles by circular tiles as shown in figure (3-17). In the case of a country regions, the frame of reference is the partition of the country into cells which share boundaries but do not overlap. RCC relationships could then be calculated by using the longitude and latitude of the region as the center of the cell and then calculating the distance between cells.

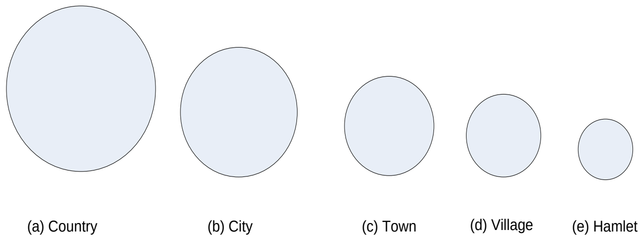


Figure 3-15 Region Classification based on approximate area size

The difference between each type is identified by a set of features specially the size of the region. By comparing the distance between center of the cells and the reference distance, we can calculate the following relations:

Disconnected (DC): if two cells, R1 and R2, share no border then the relation between them is denoted by DC(R1,R2). This is calculated using the following formula

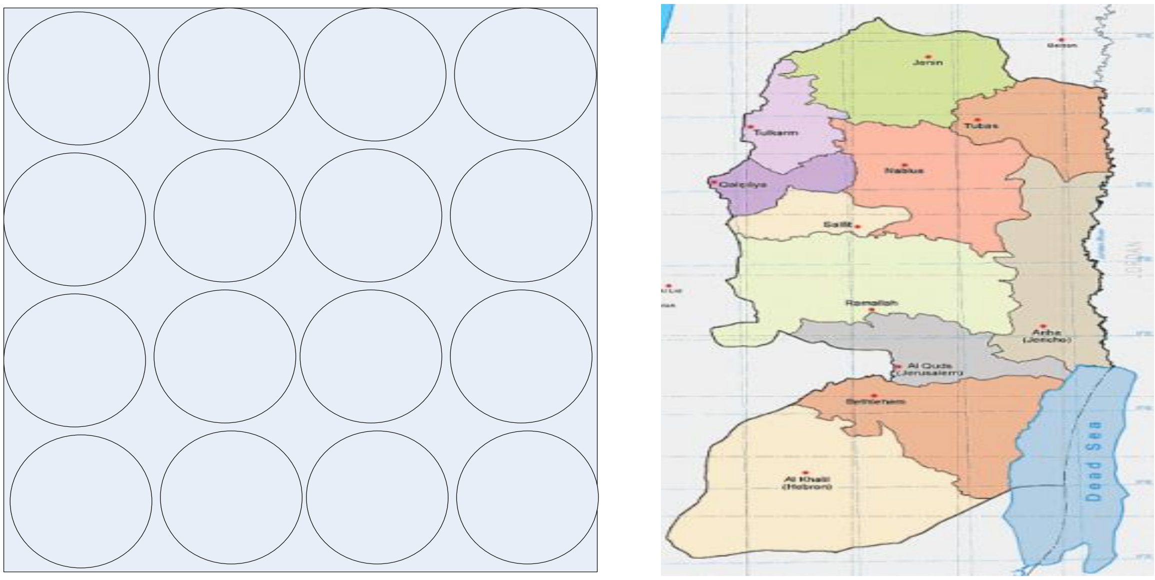
DISTANCE : (R1, R2) > (2* α + c) ; α denotes a constant that represents the maximum radius of a town and c denotes an error margin constant

Externally connected (EC): if two regions, R1 and R2, share borders then the relation between them is denoted by EC(R1,R2).
DISTANCE : (R1, R2) < (2* α + c)

Equals (EQ): the relation between each town, or any other location type, and itself is denoted by ED(R1,R2).

DISTANCE : (R1, R2) < c

Both DC and EC relations are bidirectional. The algorithm is basically divided into three main parts: (1) calculates relations between town or cities (2) calculates relations between suburbs and towns (3) calculates relations between point of interests and suburbs or town with no suburbs. Following pseudo code illustrates the order of calculating the relations based on radius value.



(a) Approximation using circular tiles **(b) Exact region tiles**

Figure 3-16 Approximate and exact tiles for a region

```
Pseducode for RCC8 Relations among towns/cities

Declare region Radius α // represents the maximum radius in meters

Declare c // denotes an error margin constant defined in meters


Input region dataset containing longitude, latitude, place name

TOWNS_SET = FIND_ALL_LOCATIONS_BY_TYPE (“TOWN”)

POIS_SET = FIND_ALL_LOCATIONS_BY_TYPE (“POIS”)

SUBERBS_SET = FIND_ALL_LOCATIONS_BY_TYPE (“SUBERB”)

BEGIN

    Build RCC8 Relations among towns
    Build RCC8 Relations among Suburbs and Towns

    Build RCC8 Relations among places of interests, towns and suburbs
    and towns

END

Output set of Relations between all regions
{EQ,DC,EC} // same type.
```

3.2.4.1 Building Town Suburb Relations

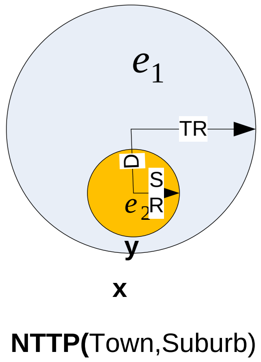


Figure 3-17 NTTP(Town, Suburb)

The second part of the algorithm is concerned with building the relations between towns and suburbs.

TownRaduis > Distance + SuburbRaduis + Constant
This part of the algorithm try to build the town or city suburbs only based on the input data which are are the lat,lon and suburb name.

3.2.4.2 Building Suburb- Suburb Relations

Building the suburb-suburb relations, follows the same approach for towns except we limit the comparison among a city or town suburbs.

Externally connected (EC): if two regions, S1 and S2, share borders then the relation between them is denoted by EC(S1,S2).
Equals (EQ): the relation between each suburb, or any other location type, and itself is denoted by EQ(S1,S2).

DISTANCE : (S2, S2) < c

3.2.4.3 Building POI Relations

A point of interest can be located either in a city or a suburb. The set of relations are all At this stage we are mainly considering the NTPP relation between a point and a region (suburb,village and town).

NTPP: a POI(S) has an NTTP relation with a town (T) if a POI lies in a town and shares no border with it. The relation is denoted by NTTP(S,T)

3.2.4.4 An illustrative example

To build the data set for this experiment, we used Palestinian regions . We collected the shape files from different municipalities like the one shown figure (3-19) and loaded the shape files into **PostGIS**/PostgreSQL database using the right coordination system for the selected region. The total spatial entities for this experiment is 5957 entity classified as shown in table (3-4).

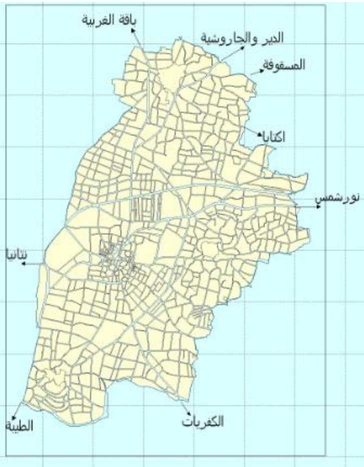


Figure 3-18 Map of a region created from a shape file

Table 3-4 Sample of spatial entities per type

Type	Count
locality	144
hamlet	23
village	323
pois	5337
suburb	39
region	7
town	81
Border Crossing	1

city	10

The challenging question at this point is how to select the best radius for each region type. Obviously the algorithm will produce wrong results if the radius is chosen too small or too large. In order to select the best radius, we created a visual map that can help the user to select the best radius. As shown in figure (3-20), choosing a radius of 800 meter will create more relations than 400 meters. Also we enhanced the algorithm by considering the area of the region. If the area of the region is found, then we can calculate the radius using the formula Area = sqrt((Area)/3.14) and thus we can get more reliable relation

3.2.5 Experiment and Validation of results

To develop our ground truth database for region relations, we had to build up the relations manually from existing maps. The ground truth data might include attribute data about the area size or population size of the region. However, not all towns or cities have these attributes filled. At this moment, we manually built the EC relationship between all towns, cities and villages. Also suburbs relations were built for two cities. Point of interests relations with their suburbs are built for nine suburbs.

- A – Number of relevant relations not retrieved
  B – Number of relevant relations retrieved

  C – Irrelevant relations retrieved

$$\text{Precision} = \frac{|B|}{|B| + |C|};$$

$$\text{Recall} = \frac{|B|}{|A| + |B|};$$

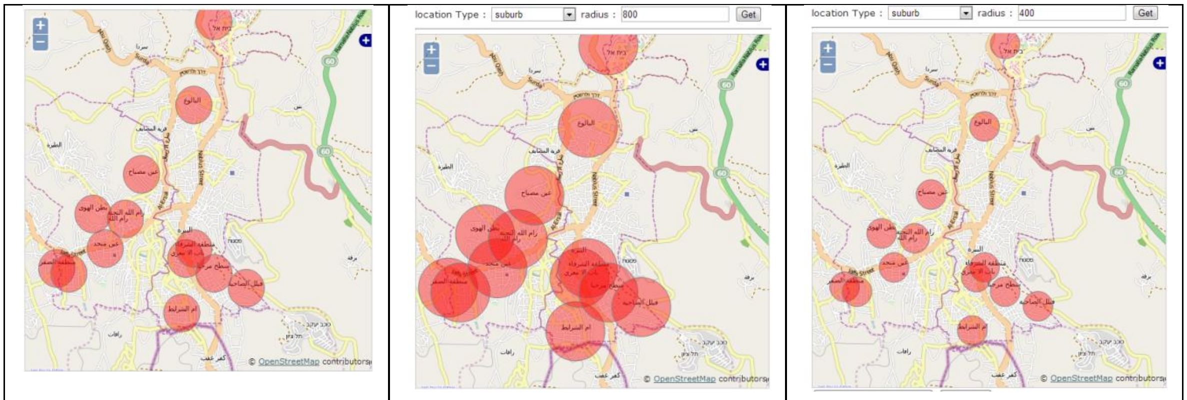
Precision = 0.82926829
Recall = 0.90265487

3.2.5.1 *Discussion of results*

Since the approach relies on approximating the area using a circle region. Selecting the radius (R) might produce wrong results as shown in the following cases. When the radius

R is much smaller than region radius (RR) (R << RR), the algorithm creates no relations between the two regions. This is equivalent to region A is disconnected from region B . This could be improved by using the area of the region to

IFFERENT REGION RADIUS BASED ON VISUAL MAP


location Type : suburb radius : 800 Get

location Type : suburb radius : 400 Get

(a) Radius 500 meter

(b) Radius 800 meter

(c) Radius 400 meter

Figure 3-19 Visual map of different region radius coverage

calculate the radius and overriding the estimated one.

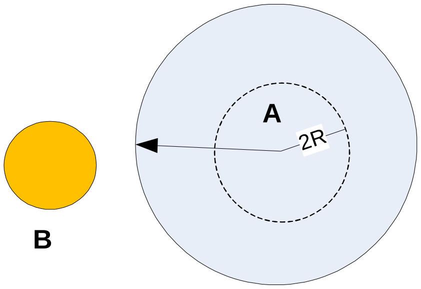


Figure 3-20 City radius larger than double of selected radius

A second case occurs when the selected radius R is much larger than region radius (R >> RR)

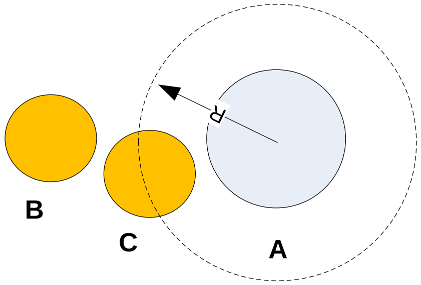


Figure 3-21 City radius less than double of selected radius

When region radius is much lesser than selected radius, it is possible to make an EC relation with a region although there is another region in between. This is equivalent to having the following relations:

(a) A Externally connect to B
(b) A Externally connect to C
(c) C Externally connect to B

Using an automated method to build RCC relations between geographic regions is challenging especially if data has only attributes related to longitude and latitude. Although locating events is best done by its address, which is the more accurate among other methods like post address or boundary, the boundary approach in many rural areas is the only option available. However, from our experiments we found encouraging results. With such results it is now possible to use the new data set to find automatically the matching relationships between a pair of events such as in the two events presented earlier:

Event 1: “8 Palestinians are arrested across the West Bank “
Event 2: “Thursday eight Palestinians arrested from Jerusalem, Jenin and Hebron, according to local and security sources”. Since Jerusalem, Jenin and Hebron has NTPP relationships with West Bank, we can infer that the location of these two events is the same.

3.2.6 Spatial Similarity Measure

There are three types of qualitative spatial relations: qualitative distances, topological relations, and directional relations. A similarity measure between location A and location B could be expressed as the weighted sum of the three relations given by the formula

$$S_{spatial}(A,B) = \boldsymbol{\alpha}\, \boldsymbol{T_{A,B}} + \boldsymbol{\beta}\, \boldsymbol{D_{A,B}} + \boldsymbol{\gamma} \boldsymbol{M_{A,B}}$$

Where, T,D,M represent the three topological, directional and distance relation respectively. Following the approach of [Egenhofer et al,1992] [Papadias et al., 1999], the Topological Relation $T_{A,B}$ is calculated based on the distance between two relations in the topological relations neighbors graph

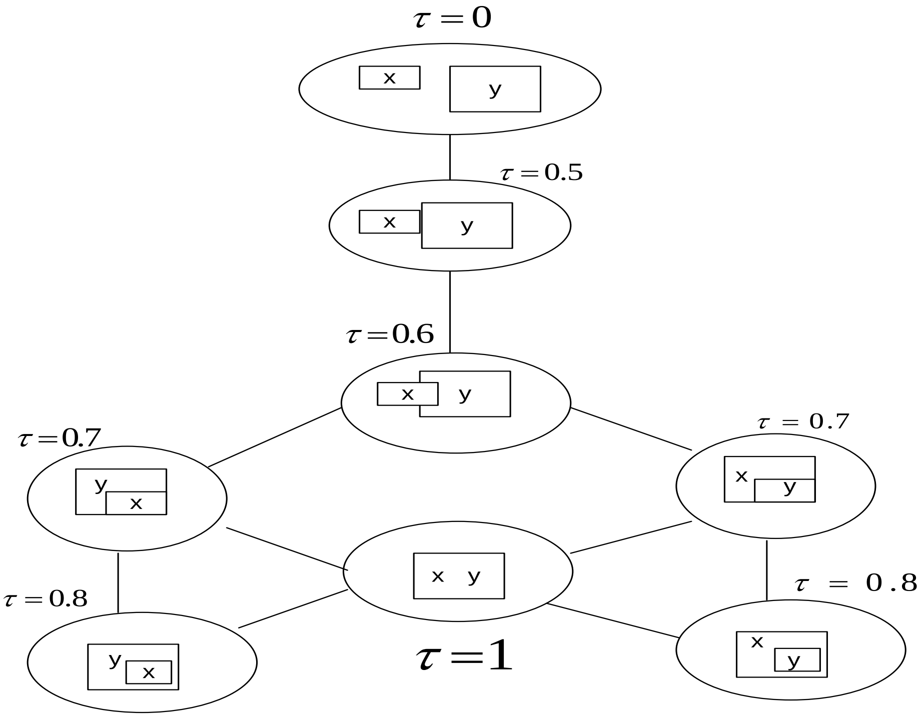


Figure 3-22 topological relations neighbors graph

and the similarity measure is given by:

$$\sigma(l_i, l_j) = \begin{cases} 1, if\ R = EQ \\ \tau\ (0 < \ \tau < 1), otherwise \\ 0, if\ R = DC \end{cases}$$

The directional and distance relations $\boldsymbol{\beta\, D_{A,B} + \gamma M_{A,B}}$ are calculated based on their fuzzy membership function (f). Both directional and distance relations could be approximated by using a trapezoidal function. The directional relation requires having a value for the angle between the centroid of the two locations, also the distance is calculated based on the distance between their centroids.

To compute the distance between the two events, we use Hausdorff distance .

$$h(A,B) = max_{a\, \in A}\{\ min_{b\, \in\, B}\{d(a,b)\ \}\}$$

```
Input:   Sets of the two locations   A, B
Output:  Similarity degree           $Sim_{spatial}(A,B)$

begin
    $h \leftarrow 0$    %% initial distance
    for every point $a_i \in A$
        $shortest \leftarrow inf$
        for every point $b_i \in B$
            $d_{ij} = d(a_i, b_j)$
            If $d_{ij} < shortest$ then
                shortest $= d_{ij}$
        End
    If shortest > h then
        $h \leftarrow shortest$
    return $h$
end
```

Distance similarity

As we presented in chapter, we will use a metric distance network of four nodes ({*equal, near, medium, far*}) as shown figure 2-4, the transformation cost is set as 1. If in one scene, the metric distance between the two objects is near, while in the other scene, the metric distance between the two objects is far, the transformation cost is 1 + 1 = 2.

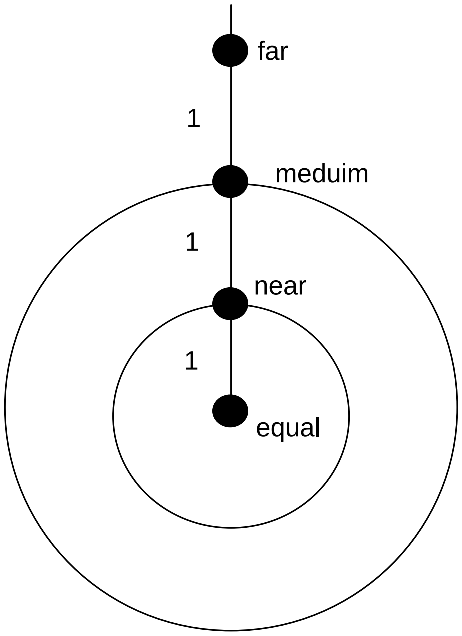


Figure 3-23 Metric distance network

We then can translate the transformation cost into a score in [0,1] using the following distance table:

Term	Distance
Equal	0
Near	0.25
Medium	0.5
far	0.75

Special considerations

- When one location is contained or is within another location, the distance between them is zero. This is similar to calculating the distance between a point and polygon when the point is inside the polygon..
- The distance between two externally connected locations is zero, this means that when two location cross or touch, the distance between them is zero.
- The distance between a point location and a polygon is calculated to the boundary of a polygon, not to the center or centroid of the polygon.

Example # 1: Let us consider the following two event, focusing only on spatial similarities.

Event 1: On 31 Mar 13, 0930 hrs, approximately 40 people demonstrated at DCO Beit-EL, NE Ramallah. It ended peacefully at1440 hrs.

Event 2: On 31 Mar 13, between 0945-1200 hrs, families protested near City Inn Hotel, North Ramallah against prisoners conditions

- **Topological relation**

	Ramallah	City Inn Hotel
Ramallah	EQ	TPP
Beit-EL	EC	EC

	Ramallah (a1)	City Inn Hotel (a2)
Ramallah (b1)	1	0.8
Beit-EL (b2)	0.5	0.5

Distance [the opposite of Topology]

Ramallah	Ramallah	equal	0
Ramallah	Beit-EL	near	1
City Inn Hotel	Beit-EL	near	1

<u>Solution</u>

The distance from A to B

D(a1,b1)= 0 ;

D(a1,b2) = 0.5

h(A,B) = 0.5

Now we take a2

D(a2,b1)= 0.2

D(a2,b2) = 0.5

By taking the max of the two values, h(A,B)= 0.5

Example # 2 .

Burj Alshaikh	Oxygen Gym

Burj Alshaikh is a commercial building ; Oxygen Gym is located on floor number seven in this building

	Ramallah (a1)	**Burj Alshaikh** (a2)
Ramallah (b1)	1	0.8
Oxygen Gym (b2)	0.8	0.8

If POIS penalize the DC and increase the weight of distance

The distance between a2 and b2 is equal which mean the distance = “0”

W1*Topology + w2*(1-Distance)

0.75*0.8 +(0.25)*1 = 0.85

Example # 3 { only point of interests }

	Taxi-AlBarq (a1)	**d(a,b)**	**shortest**	**Maximum distance**
Jaffa street (b1)	near	d(a1,b1)	0.25	
Near store Al-Manara (b2)	Medium	d(a2,b1)	0.5	0.5
Close to AL-Families park (b3)	near	d(a3,b1)	0.25	

$h(A,B) = max_{a\in A}\{\ min_{b\in B}\{d(a,b)\}\}$ = (1- h(A,B)) = 0.5

3.3 Time similarity

An Observation occurs at a particular place and time, while events unfold over time and place. Therefore we precisely may not know when the event started or finished from a particular observation. The time of an observation is expressed using either a time interval or a time instant. For example, the expression 'around 8:30 AM' denotes an interval time period, while at 8:30 AM denotes the time instant of an observation.

The definition of interval-interval relations or interval-instant relations is crucial in the context of observed events. In particular, having a clear definition of two time intervals or between a time interval and a time instant, will allow us to handle the query "How similar are these times?" for different options as listed in table:

Table 3-5. Examples of different times used in observations

Query on the pair $O = \{o_1, o_2\}$		Type of relation
On 31 Aug 14, night	On 31 Aug 14, evening	Interval-interval
30/08/2014 10:53	30/08/2014, overnight	instant -interval
On 29 Jul 14, 0600-0800 hrs	On 29 Jul 14, 0625 hrs	Interval- instant (specific duration)

To reason using different types of time intervals or instants, we will unify the processing of different time relations by using ideas from the fuzzy set [Zadeh, 1965], in which the membership degree of each element lies in the interval [0,1]. A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The input space is referred to as the universe of discourse.

For each Temporal Linguistic Terms as the ones listed in table (3-6), we will consider each linguistic term as a fuzzy number. In addition to depicting the linguistic term by its function we need to define its relation with other linguistic terms.

Table 3-6 Temporal Linguistic Terms and their membership function

id	**Terms**	MF
1	Midnight	midnight = [0,0,0,1/24]
2	Late night	latenight = [2/24,4/24,4/24,5/24];
3	Dawn	
4	Early morning	earlyMorning= [5/24,6/24,8/24,9/24];

5	Morning	morning= [5/24,6/24,11/24,12/24];
6	Noon	noon= [11/24,12/24,12/24,13/24];
7	Afternoon	afternoon= [13/24,14/24,16/24,18/24];
8	Evening	evening= [18/24,19/24,19/24,20/24];
10	Night	night= [20/24,21/24,22/24,23/24];

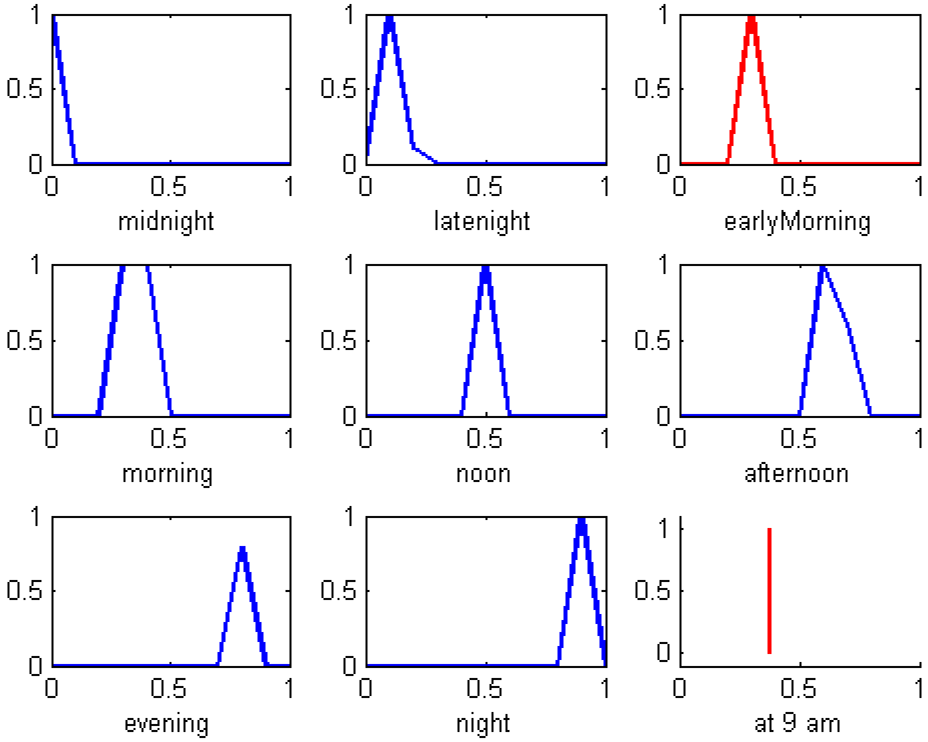


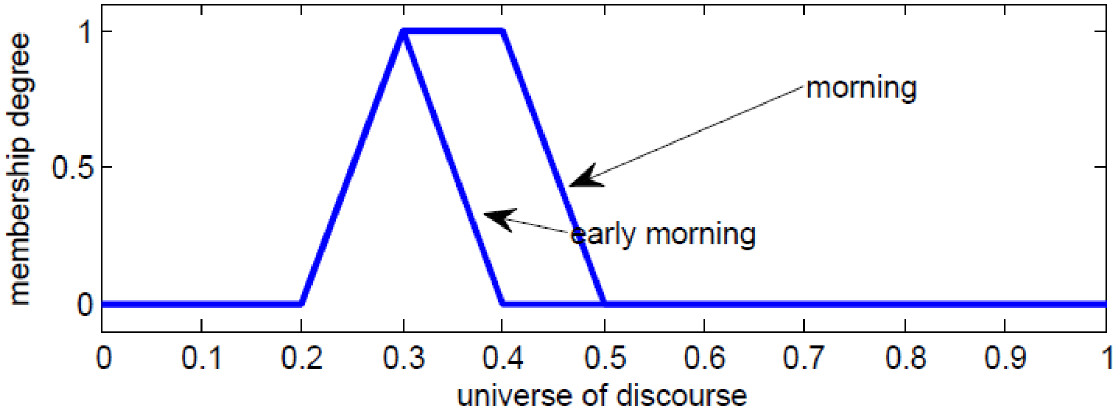
Figure 3-24 Time linguistic terms and membership functions

The relations between time fuzzy sets are defines as:

DEFINITION 2.2. (Subsumed Time Set) A fuzzy set A, on the universe of discourse X is subsumed within a fuzzy set B, on the universe of discourse X if and only if for all

$$\boldsymbol{x'} \in \boldsymbol{X}\, \boldsymbol{f_B}(\boldsymbol{x'}) \geq \boldsymbol{f_A}(\boldsymbol{x'})$$

For example, we say that the fuzzy set of the linguistic term ‘morning’ subsumes the fuzzy set of the linguistic term ‘early morning’.



DEFINITION 2.3. (Partially overlapping) fuzzy set A, on the universe of discourse X is partially overlapping another fuzzy set B, on the universe of discourse X if and only if $\exists x'$ where $\boldsymbol{f_A(x') = max(f_A)}$ **but** $\boldsymbol{f_B(x') \neq max(f_B)}$**, and** $\exists x''$ where $\boldsymbol{f_B(x'') = max(f_B)}$ **but** $\boldsymbol{f_A(x'') \neq max(f_A)}$

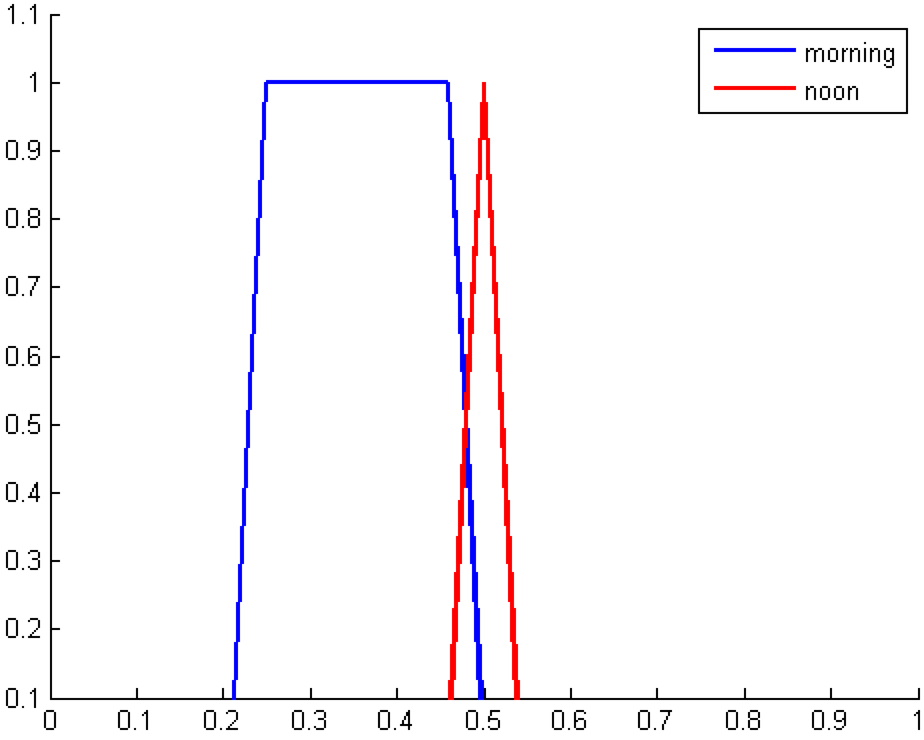


Figure 3-25 Partially overlapping time intervals

DEFINITION 2.3. (Distinct Time Set) A fuzzy set A, on the universe of discourse X is distinct from fuzzy set B, on the universe of discourse X if and only if for all $\boldsymbol{x' \in X}$ ***when*** $\boldsymbol{f_A(x') > 0}$ **then** $\boldsymbol{f_B(x') = 0}$ **and when** $\boldsymbol{f_A(x') > 0}$ **then** $\boldsymbol{f_A(x') = 0}$

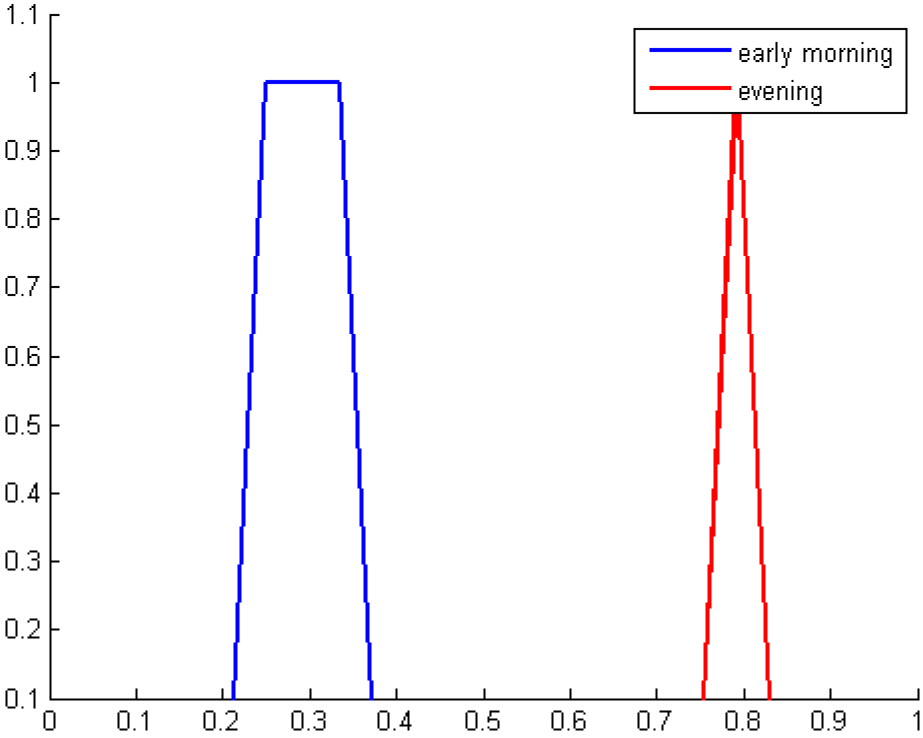


Figure 3-26 Distinct time intervals

Suppose we have two observations $\mathrm{O} = \{\mathrm{o}_1, \mathrm{o}_2\}$, Both $\mathrm{o}_1, \mathrm{o}_2$ can reflect time instants or time intervals. We define the time (instant or interval) to match if:

For $t_1 \in o_1$ and $t_2 \in o_2$, if

- t_1 is time instant, t_2 is time instant

  $$t_1 = t_2$$

- t_1 is time instant, t_2 is time interval

  $$t_{2_{start}} \leq t_1 < t_{2_{send}}$$

- t_1 *is time interval*, t_2 *is time interval*

$$\exists t : t_{1_{start}} \leq t < t_{1_{send}} \wedge t_{2_{start}} \leq t < t_{2_{send}}$$

However, several linguistic variables for day times are frequently used in describing the event time. The linguistic variables ‘Morning’, ‘Early morning’ and ‘Late morning’ are used to denote the different time periods between dawn and noon. To be able to match such time terms which are frequently used in describing events we need either to:

- Map a time instant to a time interval, or
- Map a time interval to a time interval.

Example 1. let v be a linguistic variable denoting the period of the day called ‘***Morning’***. The values of v, which are fuzzy variables, could be defined by the fuzzy set ***A = {early morning, late morning}*** and the associated base variable for early morning could span the time period from 5 AM till 9 AM and the period from 11 AM till Noon for ***‘Late morning’***.

Example. To formailize the linguistic variable ‘early morning’, we may choose a fuzzy trapzoid[a,b,c,d] which returns a fuzzy set with membership grades that linearly increase from 0 to *h* in the range *a* to *b*, are equal to *h* in the range *b* to *c*, and linearly decrease from *h* to 0 in the range *c* to *d*. Arguments *a*, *b*, *c*, and *d* must be in increasing order, and *h* must be a value between 0 and 1, inclusive.

$$f(x) = \begin{cases} 0, x < a \\ \frac{x-a}{b-a}, a \leq x \leq b \\ \frac{d-x}{d-c}, c \leq x \leq d \\ 1, b \leq x \leq c \end{cases}$$

The membership function declares which elements of U are members of A and which are not

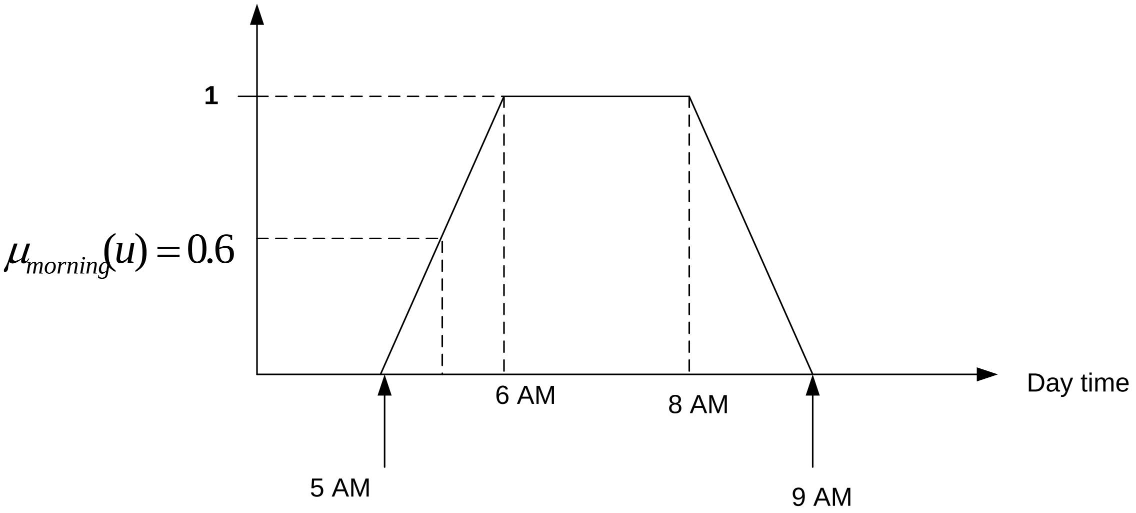
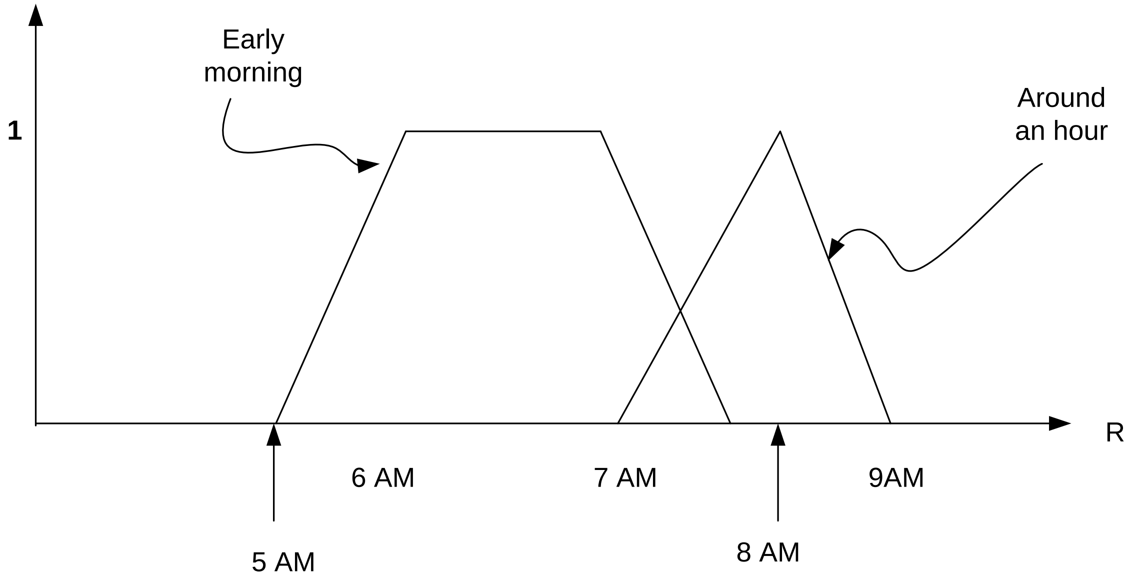


Figure 1. An example of a membership function for the fuzzy set Early Morning.

Assume that the we have two fuzzy sets A,B to formalize the linguistic variable around 8:00 AM (generalized by around hh:24) and the linguistic variable in the ‘early morning “



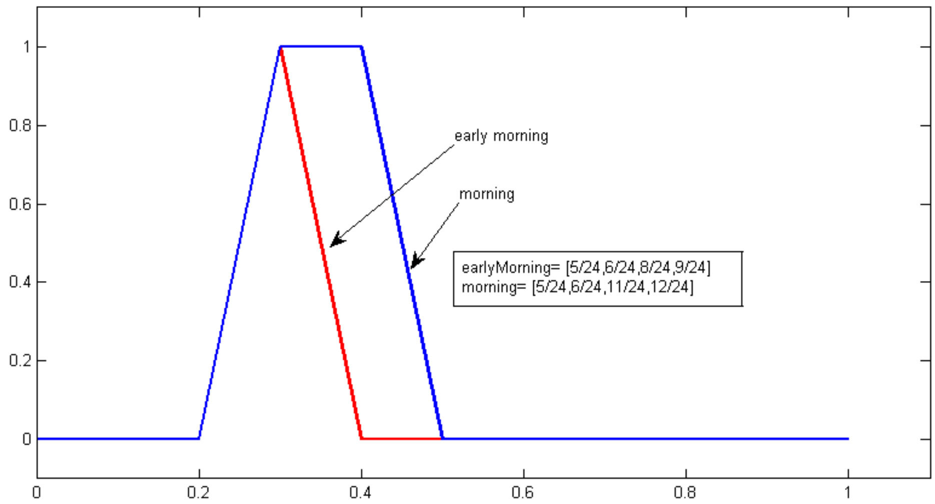
3.3.1 Time similarity measure

Since we can express any time interval or time instant using a trapezoidal function including the triangular membership function which is a particular case of the trapezoidal one [Barua et al.,2014], therefore we build our similarity measure based on the generalized fuzzy number approach. For any 2 trapezoidal fuzzy numbers $A = (a_1, a_2, a_3, a_4)$ and $B = (b_1, b_2, b_3, b_4)$, there exists different approaches to find similarity between fuzzy numbers. By examining different similarity measures for a generalized fuzzy number and for the set of time intervals defined for a day linguistic terms, we get the following results

methods = {'chen','hsieh','scgm','hamming','overlap'};

Table 3-7 Results of comparison different time functions

	'chen'	'hsieh'	'scgm'	'overlap'
Morning, earlyMorning	0.937500	0.941176	0.809519	0.823529
earlyMorning, at9	0.916667	0.923077	0.700231	0.777778
earlyMorning, at13	0.750000	0.800000	0.468750	0.538462
earlyMorning, at20	0.458333	0.648649	0.175058	0.35000



Based on the results presented in table (3-7) and comparing that with human judgment on different linguistic terms, we found that the simple center of gravity method gives the best results. The s**imple center of gravity method (SCGM) Chen and Chen** [Chen et al., 2003] is defined as :

$$S(A,B) = \left[1 - \frac{\sum_{i=1}^{4} a_i - b_i}{4}\right] * (1 - |x_A^* - x_B^*|)^{B(S_A,S_B)} * \frac{\min\,(y_A^*, y_B^*)}{\max\,(y_A^*, y_B^*)}$$

Where,

$$y_A^* = \begin{cases} \frac{w_A\left(\frac{a_3 - a_2}{a_4 - a_1} + 2\right)}{6} \; if\ a_1 \neq a_4 \\ \frac{w_A}{2} \;\; if\ a_1 = \ a_4 \end{cases}$$

$$x_A^* = \frac{y_A^*\,(a_3 + a_2) + \ (a_4 + a_1)(w_A - y_A^*)}{2w_A}$$

$$B(S_A, S_B) = \begin{cases} 1\ if\ S_A + S_B \ > 0 \\ 0\ if\ S_A + S_B \ = 0 \end{cases}$$

3.4 Participants similarity

When an instance of an observation consists of some property-value descriptions about a participant, a pair of distinct instances A and B has partial match if they refer to the same participant. Each participant is assigned a unique role. The role that a participant plays is called a thematic role and sometimes called a semantic role and is defined as:

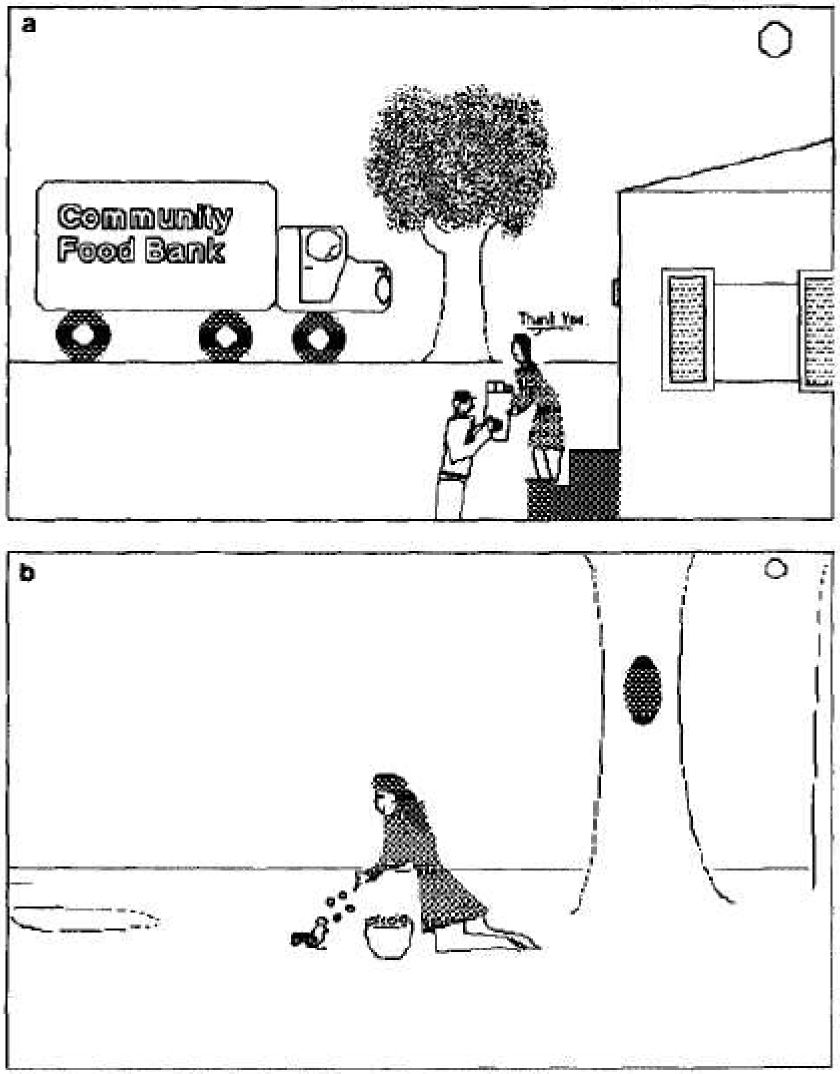
Definition. Thematic Role. Is the underlying relationship that a participant has with main verb in a clause. [Payne, 1997]

In one clause we may have more than one role defined as follows”

- **Agent:** The ‘doer’ or instigator of the action denoted by the predicate and sometimes it is called the **ACTOR**
- **Patient:** defines patient as the entity undergoing a change of state or location, or which is possessed, acquired or exchanged. The ‘undergoer’.
- **RECIPIENT:** a subtype of GOAL involved in actions describing changes of possession.

- **Experiencer:** The living entity that is moved by the action or event denoted by the predicate. Aware of event, but not in control. entity moved or located
- **Theme:** The entity that is moved by the action or event denoted by the predicate.
- **Goal**: The location or entity that in the direction of which something moves.
- **Instrument:** an inanimate thing that an agent uses to implement an event**.** Means by which event comes about
- **Manner**: how the event is carried out.

In their analysis to similarity between different scenes [Markman and Gentner,1993] as shown in figure (3-28), they differentiated between two types of mapping between the two scenes . They called the first map as perceptual mapping: which in the scene is between the two women. While the other mapping which is called matching based on relational structure (structural alignment) would align the women in the first scene to the squirrel, because in the first scene the women is receiving food, while the women in the second scene is giving food to the squirrel



3-27 Sample pair of causal scenes containing a cross-mappings. The woman in the top scene is receiving food, while the woman in the bottom scene is giving food away [Markman and Gentner,1993]

This argument by [Markman and Gentner,1993] has a close relation to our event observations. Two observations with participants are alignable only if the participant has the same role in both observations. Participants of different types are not alignable. Furthermore, if two observations have the same type and number of participants, alignment takes place at the feature level between the two participants. As also indicated by [Medin et al.,1993], what gets aligned is not fixed a priori but depends on the particular context of the comparison. For instance, if we don't have enough information about the details of the participants we conduct the comparison at the observation scene level [participant type and number for instance].

3.4.1 Thematic Role similarity measure

Table 3-8 Thematic role similarity example

Type	participant Role	Participant type::Role	instrument	Object attributes	Time	location
Car hit	Child::Patient	Driver::agent	Vehicle	Name:X Age:12	18:34	Nablus
Car hit	Child::Patient	Driver::agent	Vehicle	Name:Y Age:7	18:40	Nablus
Car hit	Child::Patient	Driver::agent	vehicle	Name:Z Age:5	18:45	Nablus

Let us consider the example given in table (3-8) and calculate the similarity based on the structural alignment approach. Thematic similarity should be calculated at two levels

3.4.2 Scene level similarity

$$Role(p_{o_1}^{(1)}) = Role(p_{o_2}^{(2)})$$

$$Number(p_{o_1}^{(1)}) = Number(p_{o_2}^{(2)})$$

Group vs. individual

$$Type(p_{o_1}^{(1)}) = Type(p_{o_2}^{(2)})$$

$$sim_{scene} = \frac{1}{1+d};$$

Where d is the difference function . A simple approach to calculate the difference is by counting the missed links

$$sim_{scene} = \frac{1}{1 + Count(missed\_links)}$$

Example

Observation#1	**Observation#2**	**#of links**						
Agent	Agent	1	1	1	1	1	0	
Patient	Patient	1	1	1	1	0	0	
Experiencer	Experiencer	1	1	1	0	0	0	
Instrument	Instrument	1	1	0	0	0	0	
Manner	Manner	1	0	0	0	0	0	
	Links	0	1	2	3	4	5	
	Sim	1	0.833	0.667	0.4	0.2	0	

```
Input:    Link counts          $L_0, L_1$ %% number of aligned and non-aligned links
Output:   Similarity degree    $Sim_{scene}(O_1, O_2)$

begin
    $sim \leftarrow 0$
    If $L_1 = zero$ then
        $sim_{scene}(O_1, O_2) \leftarrow 0$
    else If $L_0 = zero$ then
        $sim_{scene}(O_1, O_2) \leftarrow 1$
    else
        $sim_{scene}(O_1, O_2) = \frac{1}{1 + \frac{L_0}{L_1}}$
    End else
    return $Sim_{scene}(O_1, O_2)$
end
```

3.4.3 Object level similarity

As an example of observation instances, we consider the following instances p_1 *and* p_2

(p_1 ,“Unknown”)	:hasName
(p_1 ,“young ”)	:has Age
(p_1 ,Tall)	:hasHeight
(p_1 ,Male)	:hasGender
(p_1 , Black)	:hairColor
(p_1 ,2.36)	:hasSpecialMark
(p_1 , around 60 Kg)	:hasWeight

And the second instance

(p_2,“Unknown”)	:hasName
(p_2,“around 30 years”)	:has Age
(p_2,>180 cm)	:hasHeight
(p_2,Male)	:hasGender
(p_2, Black)	:hairColor
(p_2,2.36)	:hasSpecialMark
(p_2,Light)	:hasWeight

To measure the similarity between two participants, we define a weighted aggregation function that sums all the scores for all properties as follow

$$\boldsymbol{sim_{object}} = \frac{\sum_{i}^{n} \boldsymbol{w_i} * \boldsymbol{Aff}(\boldsymbol{A_i^{p_1}}, \boldsymbol{A_i^{p_2}})}{\sum_{\boldsymbol{A}\in\{\boldsymbol{A^{p1}}\}\cup\{\boldsymbol{A^{p2}}\}}^{n} \boldsymbol{w_A}}$$

The affinity function $Aff(A_i^{p_1}, A_i^{p_2})$ calculates the similarity between each pair of attributes. However, there might be an infinite number of properties and therefore an infinite number of affinity functions. However, in practice there are few attributes that are commonly associated with each type of events. An expert can select the weight and the order of attributes that should be filled first and any other attribute is not considered in the calculation. Also the affinity function itself should be defined, for instance, an affinity function to compare the height of two participants is not based on getting a precise height from the observer, rather a linguistic term is usually associated with height, therefore a fuzzy function that can compare between two heights is more common.

3-9 Example of participant properties

		Type of affinity function
(p_1 ,“Unknown”)	:hasName	function
(p_1 ,“young ”)	:has Age	Fuzzy function
(p_1 ,Tall)	:hasHeight	Fuzzy function
(p_1 ,Male)	:hasGender	Lookup
(p_1 , Black)	:hairColor	Lookup
(p_1 ,”Y”)	:hasSpecialMark	Boolean

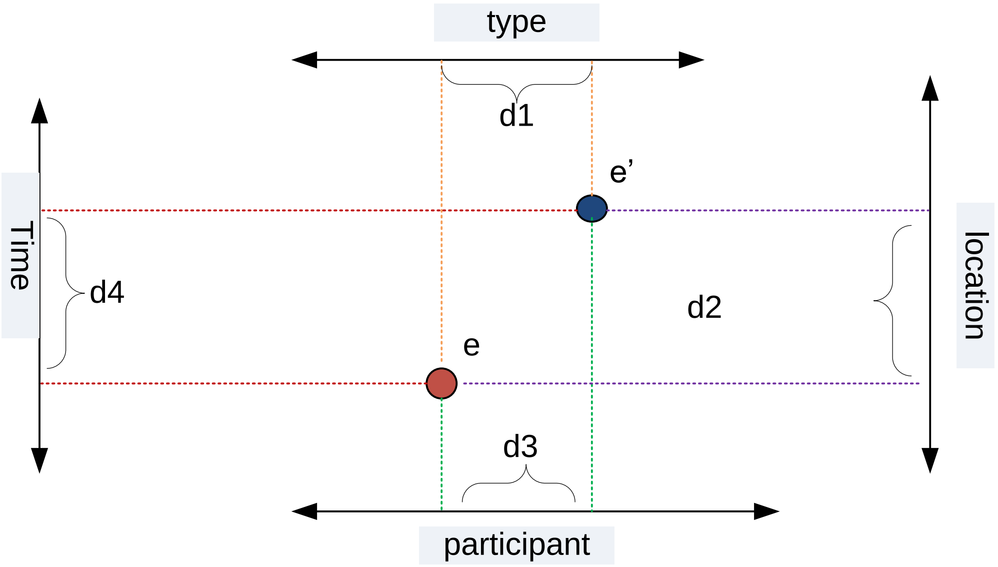
		function
(p_1 , around 60 Kg)	:hasWeight	Fuzzy function

4 Algorithmic Framework for Learning Similarity Relations

In the previous chapter, we illustrated the approaches to calculate the similarity measure between various components of any two event observations. In this chapter, we will capitalize on having these similarity measures and calculate the combined similarity for a pair of events. For this purpose, we utilize two methods: logistic regression and support vector machines. We analyze the features that may achieve maximum classification performance and justification for selecting these features.

4.1 Base and Aggregate Measures

For illustration purposes let us consider positioning the two events on 2-dimensional space. The relative distance between the two events can be uniquely determined from the distances between their attributes $\{d_1, d_2, d_3, d_4, \dots, d_n\}$



An overall similarity, can be defined using an aggregation similarity function as:

$$f^* : [0,1]^n \rightarrow [0,1]$$

The similarity function f^* accepts attribute-attribute similarities in the range of [0,1] and produces a similarity score in the same range. For a pair of two events (e,e'), f^* is equivalent to:

$sim((\mathrm{e},\mathrm{e'})) =$ $w_1 S_{type}((\mathrm{e},\mathrm{e'})) + w_2 S_{time}((\mathrm{e},\mathrm{e'})) + w_3 S_{location}((\mathrm{e},\mathrm{e'})) + w_4 S_{participant}((\mathrm{e},\mathrm{e'})) + \cdots +$ $w_n S_{attribute\_n}((\mathrm{e},\mathrm{e'}))$ …….(1), and equivalently

$$sim\left((\mathrm{S}_1, \mathrm{S}_2, \ldots, \mathrm{S}_\mathrm{n}), (\mathrm{w}_{1,}\mathrm{w}_{2,}, \ldots, \mathrm{w}_\mathrm{n})\right) = \sum_{i=1}^{n}(\mathrm{S}_\mathrm{i} * \mathrm{w}_\mathrm{i}) \tag{5-1}$$

Since not all attributes have the same impact on the overall similarity, we added a weight for each attribute. [Gower, 1971] highlighted the problems and challenges of assigning weights to individual scores.

4.2 Pairwise Classification Framework

We can consider the problem of determining whether a pair of events, (e_i, e_j) belong to the same class or not, as a pairwise classification problem. The main objective of pairwise classification is to infer the relation between two objects. As shown if figure (4-1), we can define a relation between objects of the same type (monadic) or objects of different types (dyadic).

	E_1	E_2	E_3	E_4
E_1		1	0	1
E_2			1	0
E_3				0
E_4				

a) A monadic relation between pair of events. This is a binary relation taking crisp values {0,1}. For example, this relation might measure if the pair are similar or not

	P_1	P_2	P_3
E_1		0.47	
E_2	0.5		0.3
E_3			
E_4		0.33	

b) A dyadic relation between pair of events and participants. This relation takes real values [0,1]. This relation might measure the correlation between a participant type and event type.

Figure 4-1 Monadic vs. dyadic relations

When modeling similarity, a special attention should be given to the properties of the similarity measures such as symmetric and transitivity. If we assume that similarity is symmetric then it is true that $Sim(e, e') = Sim(e', e)$, by definition this property is defined as

Definition x.1 . A binary relation $R: E^2 \rightarrow [0,1]$ is symmetric relation if for all $(e, e') \in E$ it holds that $R(e, e') = R((e', e)$

We also will assume that our similarity measures satisfies the triangle inequality. The transitivity property is more important in ranking similarity and preferences. The intuition for transitivity property is that:

If event_1 is similar to event_2 and event_2 is similar to event_3
Then event_1 is similar to event_3 .

With this intuition we have a problem if the similarity degree is required, but for clustering purpose a binary relation should be sufficient.

To learn the similarity relation between a pair of events, we need a classification function g that can be learned from a set of training examples where for each pair $(\mathrm{e_i}, \mathrm{e_j})$ of the example, we know if the pair belongs to the same class $(\mathrm{y_{ij}} \coloneqq 1)$ or not $(\mathrm{y_{ij}} \coloneqq -1)$.

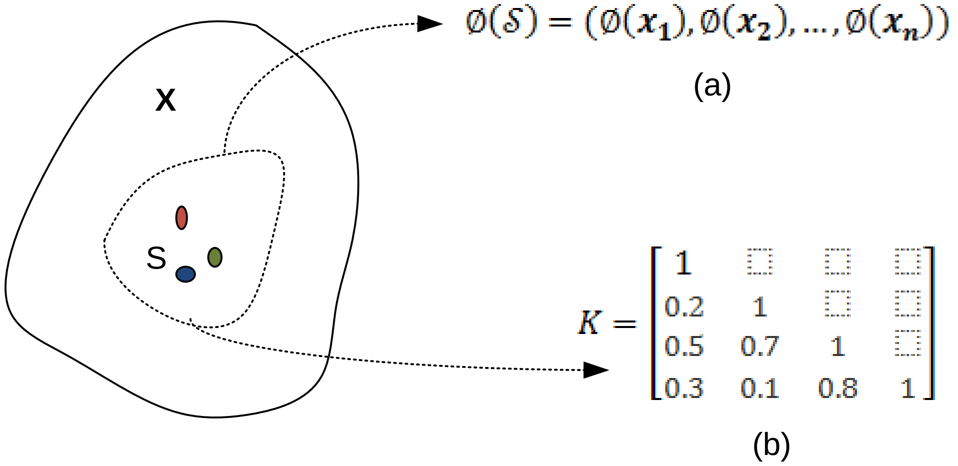
$$g(e', e) = \begin{cases} 1\ if\ e', e\ belong\ the\ the\ same\ class \\ \qquad -1\ (otherwise) \end{cases}$$

Definition 4.1 A training set of m examples can be denoted as,
$\{(y_1, (e_{q1}, e'_1), (y_2, (e_{q1}, e'_2), \dots, (y_m, (e_{qn}, e'_n)\}$, where y_k is the label and $((e_{qi}, e'_j)$ is the pair of event to compare.

Definition 4.2 Learning problem. [**Balcan, 2008]** A learning problem specified as follows. We are given access to labeled examples (x, ℓ) drawn from some distribution P over X × {−1, 1}, where X is an abstract instance space. The objective of a learning algorithm is to produce a classification function g: $X \rightarrow \{-1,1\}$ whose error rate $Pr_{(x,\ell)} \sim p[g(x) \neq \ell]$ is low. We will be considering learning algorithms whose only access to their data is via a pairwise similarity function K(x, x') that given two examples outputs a number in the range [−1, 1].

4.2.1 Representation approaches for learning

For learning purposes, events (or their observations) could be represented in two different ways. As illustrated by [Vert et al., 2004], we can represent the set of objects in S dataset by a feature representation of the objects in that dataset as shown in part (a)- figure 4-2, or by using a matrix of pairwise similarity (kernel representation) as shown in part (b) of figure (4-1).


$\emptyset(\mathcal{S}) = (\emptyset(\boldsymbol{x_1}), \emptyset(\boldsymbol{x_2}), \dots, \emptyset(\boldsymbol{x_n}))$
(a)
X
S
$K = \begin{bmatrix} 1 & ⬚ & ⬚ & ⬚ \\ 0.2 & 1 & ⬚ & ⬚ \\ 0.5 & 0.7 & 1 & ⬚ \\ 0.3 & 0.1 & 0.8 & 1 \end{bmatrix}$
(b)

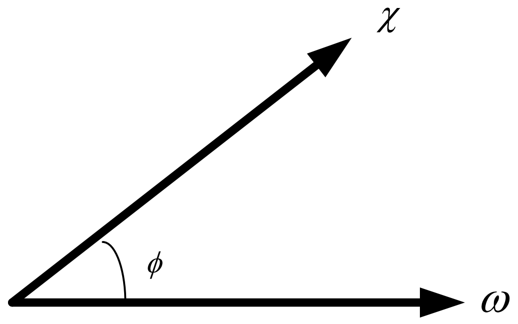
4-2 Two representation of S dataset (a) feature representation (b) Kernel representation

In the feature representation approach, the data set of n objects is represented as the set of individual object representation:

$$\emptyset(\mathcal{S}) = (\emptyset(\boldsymbol{x_1}), \emptyset(\boldsymbol{x_2}), \dots, \emptyset(\boldsymbol{x_n})) \tag{5-2}$$

In equation 5-2, the feature vector represents features from only one object, where each example is represented by a feature vector X. . The algorithm uses the inner product between X (the features vector) and a parameter vector ω ∈ $\mathbb{R}^d$, to find the values of the vector ω

$$\omega^T \chi \tag{5-2}$$



4-3 dot-product between features vector and parameter vector

The problem of finding similarity between a pair of objects, is similar to record linkage or record deduplication problem in the database domain. [Sarawagi and Bhamidipaty, 2002] used active learning to design a learning-based deduplication system to detect and eliminate duplicate records in a database. They used a set of Similarity functions each of which computes a similarity match between two records r1; r2 based on any subset of d attributes. Examples of such functions are edit-distance, soundex, abbreviation match on text fields, and absolute difference for integer fields. They use a mapper module which takes as input a pair of records r1; r2, computes similarity using the set of similarity functions nf and returns the result as a new record with nf attributes. For each duplicate pair they assign a class-label of “1" and for all the other pairs assign a class label of “0".

[Bilenko and Mooney, 2003] to detect duplicate records they first calculate the similarity measure at the field level and used the similarity outputs as a new feature to construct a new vector to represent the pair of records. At the field level, they represent each instance of the string pair by a feature vector $\mathrm{x} = (\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{n})$ and $\mathrm{x}' = (\mathrm{x}'_1, \mathrm{x}'_2, \dots, \mathrm{x}'_\mathrm{n})$ where each feature corresponds to whether a word appears in the string ; n is the number of words in the vocabulary. The new pair instance $\mathrm{x}_{\mathrm{pair}} = (\mathrm{x}_1\mathrm{x}'_1, \mathrm{x}_2\mathrm{x}'_2, \dots, \mathrm{x}_\mathrm{n}\mathrm{x}'_\mathrm{n})$ is then classified by the trained SVM, and the final similarity value is obtained from the SVM prediction. The output of this step is used to construct a new pair at the record level to classify whether the pair is duplicate or not.

In order to construct the set of features and be able to construct a pair vector from the instances of each event or observation context, we need to identify and extract the set features $\emptyset(\mathcal{S}) = (\emptyset(\mathrm{x}_1), \emptyset(\mathrm{x}_2), \dots, \emptyset(\mathrm{x}_\mathrm{n}))$.

4.2.2 Features Extraction

The goal of the feature selection task is to select the minimum set of predictors or features that achieves maximum classification performance [precision and recall]. The challenge of finding meaningful features is to keep these features applicable to wide spectrum of event types.

Event-type based features. These features are concerned with semantic similarity and relatedness similarity measures. As discussed in chapter 3.1, different similarity measures provide different scores to similarity between two concepts. This is mainly due to different approaches followed to compute similarity: node-based or edge-based or simply path or content information approaches. A node-based approach depends on the shortest path or on the average of all paths, when more than one exists. A node-based approaches rely on the properties of the terms and their ancestors or descendants. Information content is the common approach in edge based similarity and quantifies how much two concepts share information. Usually, information content is quantified using either the most informative common ancestor (MICA technique), in which only the common ancestor with the highest IC is considered; and the disjoint common ancestors (DCA technique), in which all disjoint common ancestors (the common ancestors that do not subsume any other common ancestor) are considered [Pesquita et al., 2009].

Using the following example, we illustrate different strategies for selecting pairwise similarity in a taxonomy structure. If we consider the two synsets in WordNet for the two concepts snowstorm#n#1, tornado#n#1, we get the following results

Table 4-1 Min-Max score of different similarity measures

Measure	Min_score	Max_score	Score
Path	0	1	0.2000
Wup	0	1	0.8182
Lin	0	1	0.7888
Resnik	0	infinity	9.2809
jcn	0	Infinity	0.20
LCH	0	infinity	2.07
Lesk	0	infinity	15
HSO	0	16	3

After doing normalization for given measures using different samples and following the procedure in [Sinha and Mihalcea, 2007], where they normalized the score based on the ranges of each individual measure. For the lesk measure, they observed that the edge weights were in a range from 0 up to an arbitrary large number. Consequently, values greater than 240 were set to 1, and the rest were mapped onto the interval [0,1]. Similarly, the jcn values were found to range from 0.04 to 0.2, and thus the normalization was done with respect to this range. Finally, since the lch values ranged from 0.34 to 3.33, they were normalized and mapped to the [0,1] scale using this interval.

The following strategies were examined, but didn’t yield acceptable results [compared to human judgment]:

1. Using the average of all pairwise similarities as the combination strategy
2. Using the Maximum of the pairwise similarities
3. Using the minimum of the pairwise similarities

Therefore, we selected a combination of different measures strategy which are by default normalized and represents methods of node and edge based similarity. Mainly, we select the following similarity measures to fuse them as features for event- type matching.

1. **Path length similarity:** in this category we selected the path similarity measure and Wup.

2. **Information content similarity:** in this category we selected lin similarity measure, which quantifies the informativeness of concepts.

Location based features: A single similarity measure is selected for this group, which encapsulates all the complexity of using proximity features, orientation features and distance features. As we illustrated in chapter 3, the location similarity also differentiate between place types whether they are cities, suburbs or point of interests. Encapsulating the location feature into a single measure, keeps the modularity of the learning algorithm and creates a buffer between the similarity measure and the procedures of computing it. Thus giving the end users the ability to change or modify this function.

Time based features : For the temporal features, the simple center of gravity method (SCGM) is selected . As discussed in chapter 3, a similarity based on the generalized fuzzy numbers covers the following features in a single score which are :

1. Time intervals and instants
2. Time linguistic terms
3. Begin time
4. End time

Cause and Effect features: In actionable knowledge, the direct and indirect effects of events are very important and plays a crucial role in the decision making process. There are some common features that are usually gathered regardless of the event type. We found that many departments distinguish between event types only by their effects. For example, the following event types are used to describe different vehicle accident:

- Vehicle collision –self
- Vehicle collision – more than vehicle – property damage only
- Vehicle collision – more than vehicle – pedestrian injury or death and property damage
- Vehicle collision – vehicle damage . Other combinations from the above such as collides with another vehicle, collides with pedestrian, collides with fixed object

Meteorological events also report the following effects with almost every event type:

- Deaths Direct/Indirect

- Injuries Direct/Indirect
- Property Damage
- Crop Damage

Participant based features: participant are distinguished by their thematic role in events. At the object level a function that uses a binary relation is used to compare the participants in two observations. The first feature that we have tested is the Agent and Patient similarity. A function that calculate similarity based on the following two attributes is used:

sameAgent

1. Age
2. Gender

sameRecipient

1. Age
2. Gender

However, in most cases the information about the Agent cannot be provided, causing a missed data that is missed not at random as we will discuss later in chapter 5, when handling missed data. Therefore, we only used the information about the Patient or Recipient in our feature list.

Summary of selected features

4-2 Summary of selected features

Event-Type	Location	Time	Participant	effect
• SimWup • SimLin • SimPath	• Topological and distance Similarity	• Temporal Similarity	samePatient • Gender • Age	• Deaths Direct/Indirect • Injuries Direct/Indirect • Property Damage or Crop Damage

4.2.3 Gradient Descent Approach

Logistic regression is well suited for studying the relation between a categorical or qualitative outcome variable and one or more predictor variables. For example, the similarity vector (X) also expressed as the predictors is used to predict the dichotomous outcome variable y (similar, not similar). In logistic regression the dependent variable is always binary (with two categories). Therefore the logistic regression is mainly used to for prediction and also calculating the probability of success. Logistic regression uses the gradient descent approach to maximize the likelyhood of the parameters which is explained in section (2.2)

Hypothesis

To learn the similarity relation between a pair of events, we need a classification function g that can be learned from a set of training examples where for each pair $\left(\mathrm{e_i}, \mathrm{e_j}\right)$ of the example, we know if the pair belongs to the same class ($\mathrm{y_{ij}} \coloneqq 1$) or not ($\mathrm{y_{ij}} \coloneqq -1$).

$$h_\theta(x) = \theta^T X$$

For our classification task, we need our model to give the probability of y=”1” given X and θ or $h_\theta(x) = p(y = 1|X; \theta)$ or y=”0” which is $1 - p(y = 1|X; \theta)$. Since the model values are now

$$0 \leq h_\theta(x) \leq 1$$

We represent our hypothesis with a logistic function or sigmoid function. As explained in section 2.xx, a sigmoid function confines the values between “0” and “1”.

$$h_\theta(x) = \frac{1}{1 + e^{-(\theta^T X)}}$$

And from the probability of the logistic function we can predict the value of y based on the following threshold

If $h_\theta(x) \geq 0.5$; then y=”1”

If $h_\theta(x) < 0.5$; then y=”0”

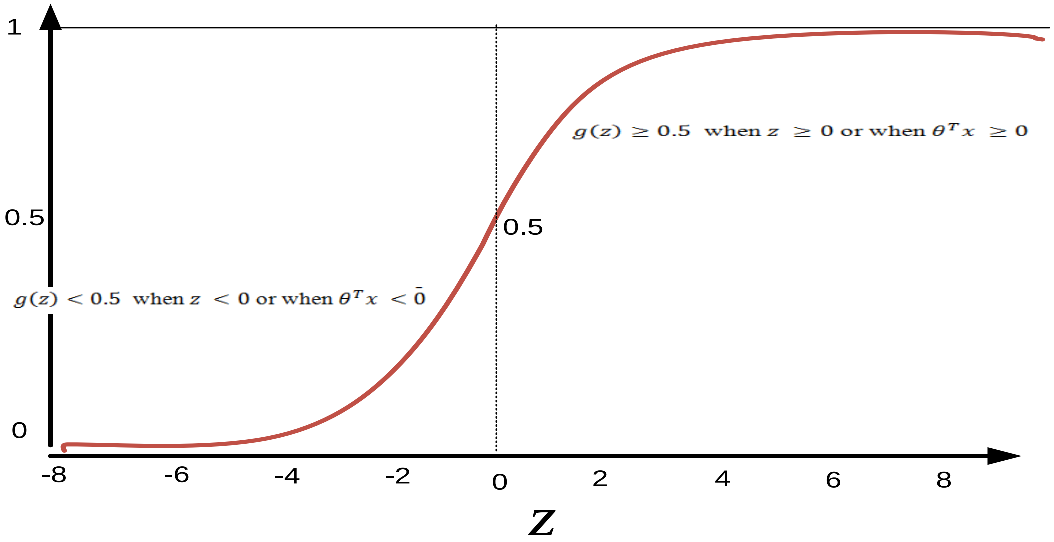
For notation purposes, if we let

$$h_\theta(x) = g(\theta^T x) = g(z) \text{ ; then}$$

$$g(z) = \frac{1}{1 + e^{-(z)}}$$

If we recall the s-curve we then realize that $g(z) \geq 0.5$ when $z \ \geq 0$ or when $\theta^T x \ \geq 0$ or $g(z) < 0.5$ when $z \ < 0$ or when $\theta^T x \ < 0$

$$y = \begin{cases} 1 \ \ when \ \theta^T x \ \geq 0 \\ 0 \ \ when \ \theta^T x \ < 0 \end{cases}$$



4-4 Decision boundary for a sigmoid function

Cost function

$$j(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost(h_\theta(x^{(i)}), y^{(i)})$$

$$j(\theta) = \frac{1}{m}\sum_{i=1}^{m} -y^{(i)} log\left(h_\theta(x^{(i)})\right) - (1 - y^{(i)})\log\left(1 - h_\theta(x^{(i)})\right)$$

Use gradient descent to minimize the cost function using the weight update rule

$$\theta_{new} := \ \theta_{old} - \ \alpha\Delta(\theta)$$

$$\theta_{new} := \ \theta_{old} - \ \alpha\frac{\partial}{\partial\theta_j} j(\theta)$$

$$\frac{\partial}{\partial\theta_j} j(\theta) = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x^{(i)}{}_j$$

$$\theta_{new} := \theta_{old} - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) x_j^{(i)}$$

Note that the likelihood function is concave, thus Gradient ascent will find the global optimal solution. To minimize the logistic regression cost function we use gradient descent method to find a local minimum of a function.

- The Logistic Regression model will be constructed by an iterative maximum likelihood procedure.

- The gradient descent works as follow:
  1. starts with arbitrary values of the regression coefficients and constructs an initial model for predicting the observed data.
  2. then evaluates errors in such prediction and changes the regression coefficients so as make the likelihood of the observed data greater under the new model.
  3. repeats using a learning rate until the model converges, meaning the differences between the newest model and the previous model are trivial.

The idea is that we find the parameters that are most likely to have produced the data.

```
Input:      Training samples   X, y
                       theta   Theta ← zeros
                       alpha   alpha = 0.01
                   num_iters   iterations = 500

Output:                theta   Optimized parameters

 begin
                           1   n = number of features
                           2   m = number of examples
                           4   for iter = 1:num_iters
                               % Perform a single gradient step on the parameter vector
                               theta
                                  4.1   Compute new theta for all feature (j=number of
                                        features) ; err = X * theta - y
```

```
                $\forall_j \, \theta_{new} := \theta_{old} - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) x_j^{(i)}$
            4.2  Normalize theta
                $\forall_j \, \theta_{new} := \frac{\theta_{old}}{\sum_{j=1}^{j} \theta_{new}}$
            4.3  Choose new value for alpha
                $\alpha := \alpha - reduction\_rate$
        end
    5   return theta
end
```

4.2.4 Kernel Method Approach

Kernel methods are widely used in interaction prediction. For example, [Ben-Hur and Noble, 2005] used kernel methods to study the relationship between pairs of protein sequences: whether two pairs of sequences are interacting or not. [Oyama and Manning, 2004] used pairwise classifiers on author matching problem. In kernel methods, an interaction between two classes is first mapped to a kernel feature space using a kernel function, and then a linear classifier is obtained in the kernel feature space. The advantage of the kernel classifier such as SVM is that it is robust against the overfitting problem and we do not need to explicitly compute the feature mapping, and rather compute the inner product of samples in the kernel feature space, which is computationally efficient for high dimensional data.

With kernel methods data is represented through a set of pairwise comparions, where a real-valued comparison function $\mathrm{k} : \mathrm{X} \times \mathrm{X} \rightarrow \mathbb{R}$ is used and the data set of n objects $\mathcal{S}$ is represented by $\mathrm{n} \times \mathrm{n}$ matrix of pairwise comparisons $\mathrm{k_{ij}} = \mathrm{k}(\mathrm{x_i}, \mathrm{x_j})$ [Vert et al.,2004]. An example for a sample similarity matrix is given in figure 5-4.

$$K = \begin{bmatrix} 1 & & & \\ 0.2 & 1 & & \\ 0.5 & 0.7 & 1 & \\ 0.3 & 0.1 & 0.8 & 1 \end{bmatrix}$$

4-5 Kernel similarity matrix

In pairwise classification the aim is to decide whether the examples of a pair $(x_i, x_j) \in \mathbb{R}^n \times \mathbb{R}^n$ belong to the same class or not. Following [Brunner et al., 2012], given a training data $((x_i, x_j), y_{ij})$, where $y_{ij} = +1$ if the examples of the pair (x_i, x_j) belong to the same class and $y_{ij} = -1$, if the examples of the pair (x_i, x_j) do not belong to the same class. A sample of a training example is shown in table 5-2.

Training dataset

4-3 An example of pairs in the training dataset

Event pair	**Class**
e_1, e_2	1
e_1, e_3	1
e_1, e_4	-1
e_2, e_6	1
e_2, e_5	-1
….	…

$$\overrightarrow{x_1}$$

4-4 The pairwise kernel matrix with sample similarity measure

K	e_1, e_2	e_1, e_3	e_1, e_4	e_2, e_6	….
e_1, e_2	1				
e_1, e_3	0.5	1			
e_1, e_4	0.4	0.2	1		
e_2, e_6	0.3	0.6	0.4	1	….
e_2, e_5	0.6	0.5	0.7	0.1	
….					

Pairwise decision function

The pairwise decision functions $f: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, predicts whether a new pair (x_1, x_2) Belong to the same class ($f(x_1, x_2) > 0$) or not ($f(x_1, x_2) < 0$).

The pairwise decision functions f should be symmetric

$$f(x_1, x_2) = f(x_2, x_1) \text{ for all } x_1, x_2 \in \mathbb{R}^n$$

For a given m training examples $x_i \in \mathbb{R}^n \, with \, i \in M :== \{1, \dots, m\}$ and assume that I ⊆ M × M, Frequently, a pairwise decision function f is given by

$$f(x_1, x_2) := \sum_{(i,j)\in I} \gamma_{ij} K((x_i, x_j), (x_1, x_2)) + b$$

where $b \in \mathbb{R} \, and \, \gamma_{ij} \in \mathbb{R} \, for \, all \, (i,j) \in I.$
And K is pairwise kernel function, which could be any of the following [Brunner et al., 2012]:

- symmetric direct sum pairwise kernel
$$K_{SD}\big((a,b),(c,d)\big) := \frac{1}{2}(\langle a,c\rangle + \langle a,d\rangle + \langle b,c\rangle + \langle b,d\rangle$$
- metric learning pairwise kernel
$$K_{ML}\big((a,b),(c,d)\big) := \frac{1}{4}(\langle a,c\rangle - \langle a,d\rangle - \langle b,c\rangle + \langle b,d\rangle)^2$$
- Tensor learning pairwise kernel
$$K_{TL}\big((a,b),(c,d)\big) := \frac{1}{2}(\langle a,c\rangle\langle b,d\rangle + \langle a,d\rangle\langle b,c\rangle$$
- asymmetric tensor pairwise kernel
$$K_{AT}\big((a,b),(c,d)\big) := \frac{1}{4}(\langle a,c\rangle\langle b,d\rangle - \langle a,d\rangle\langle b,c\rangle)^2$$

If we consider the Tensor Product Pairwise Kernel

$$K_{TP}\big((x_1,x_2),(x_1',x_2')\big) := K'(x_1,x_1')K'(x_2,x_2') + K'(x_1,x_2')K'(x_2,x_1')$$

Then we only need to define the kernel K' to be able to use this approach. Unfortunately, the only reasonable kernel function to find similarity between a pair of events is to revert back to the approach described in the gradient descent approach and define the kernel based on the events features. Therefore, we will test the feature model using one of the existing kernels such as the RBF kernel, linear kernel or a polynomial kernel.

$$K' = K_{rbf}(\boldsymbol{x}, \boldsymbol{x'}) = \; exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x'}\|^2}{2\sigma^2}\right) = exp\left(-\;\frac{\sum_{j=1}^{n}(x_j - x_j')^2}{2\sigma^2}\right)$$

4.3 System Architecture

From a user perspective the main use case of the System is to enable the detection of duplicate events in the recent past without the need to handle them manually one by one.

The detection of duplicate events is handled by clustering similar events together, thus making a group of cluster, where the end user can smoothly follow before taking any decision. We now define our clustering system as a tuple:

EC = [S, KB, M, *Sim*(e), C]

where:

- $S = \{e_1, e_2, e_3, \ldots, e_n, \ldots\}$ time ordered series of event records e_i has the form $e_i \; = \{\, e_{RDF}^{type}, e_{RDF}^{Time}, e_{RDF}^{location}, e_{RDF}^{participant}, e_{RDF}^{description}, e_{RDF}^{state}, e_{RDF}^{cause}\}$.
- KB = $\{\, \mathrm{O}_1, \mathrm{O}_2, \ldots, \mathrm{O}_\mathrm{n} \,\}$ a set of ontologies; in current version event Ontology and region ontology is implemented.
- M = is the event model used to represent the transformed event description from the observation format to RDF format.
- *Sim*(e) is the similarity function that measures the similarity between a new event $e \in E$ and existing events and returns the probability value of being similar or not similar.
- C = set of n clusters .

Definition (Cluster). Let S be a stream on E. A cluster C is a set of events in S ; $\mathrm{C} \sqsubseteq \mathrm{set}(S)$. A clustering function is a function f mapping S onto a set of clusters, f(S) = $\{\, sim(e) |\; e \;\in S \,\}$.

The high level architecture of the system is shown in the following diagram followed by a brief description of each component:

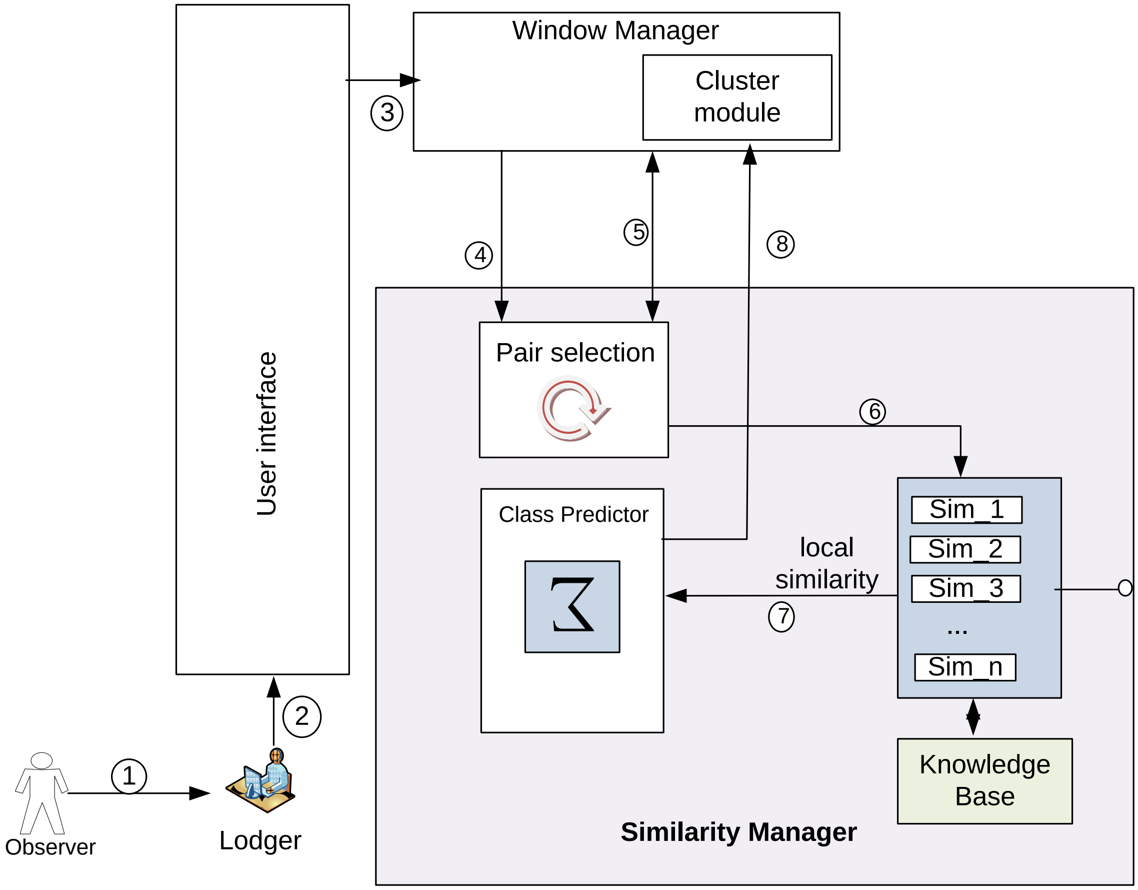


Figure 4-6 Event Matching Architecture

The system consists of the following components:

1. Event Lodging module (GUI)
2. Window Manager
3. Similarity Manager
4. Integration Manager

Event Lodging Module: This component is responsible for entering the observation data into the system using a graphical user interface utility (GUI) which is designed based on the structure of the event model. It supports manual acquisition of observation data as well as automated observation from different online feeds. The translation from text format to structured format is done manually at this stage, since many of the observations are received by phone calls or through non-structured feeders.

Event Model: The event model (*M)* and its user interface is used to transform the description of the events from natural language to RDF stream. Each event is represented as an RDF graph built using the event base model (*M*).

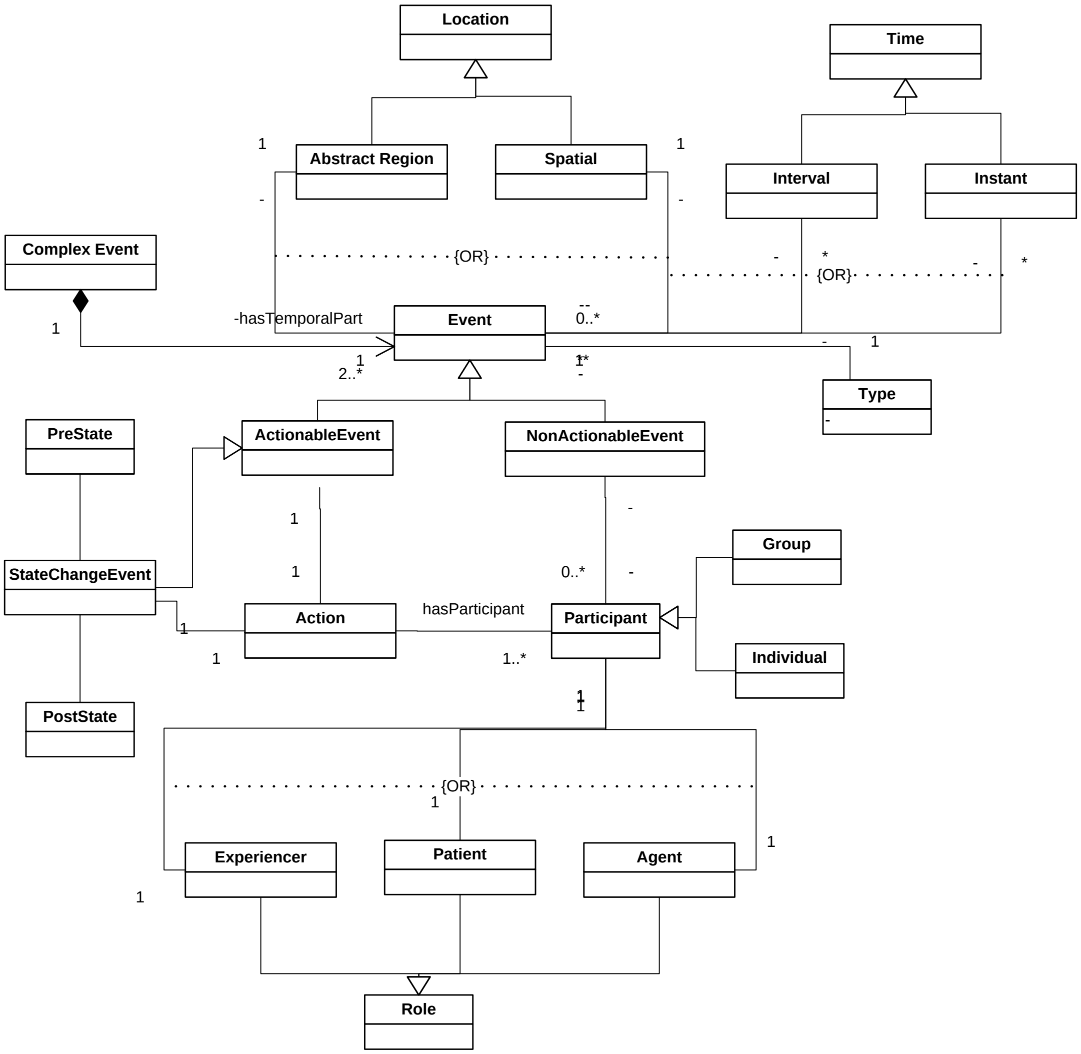


Figure 4-7 UML Event Model

The notation of Complex Events: An event could be atomic or complex. Complex events could be composed from atomic or complex events. When removing the complex event (the whole event), the part event is also removed. The lifetime of the part events is dependent on the lifetime of the complex (whole) event. Complex events are composed from at least two atomic or complex events.

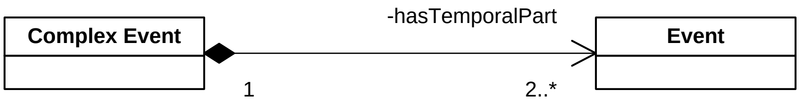


Figure 7. Complex Event

The event actions: Each atomic event has one and only one action <verb>

[WordNet, 2010], [Zacks and Tversky, 2001], OpenCyc [Jaegwon, 1973] and DUL[2] [Gangemi et al., 2002] consider action as a particular type of event . Actions in the ABC [12] ontology are not special types of events, therefore one event may have more than one action . In ABC it is easy to consider two events as one because of the action problem as in the birth event which has more than one action and is mixed with the delivery event

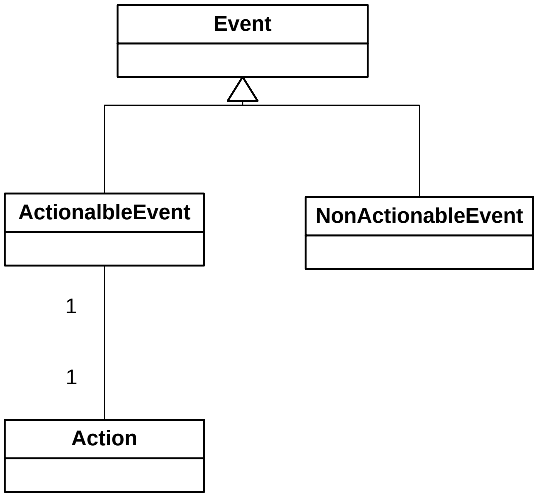


Figure 8. Actionable Event

Pre and Post states: If events mark changes in the state, then there should be at least one object involved in the event. Objects could be created as a result of an event or destroyed or have changes in some properties. In Give birth event the pre-situation is a world with total number of live objects; post-situation a new object (person) exists therefore the type of change here is creating a new object. In kill event, the post-situation is a change on the state of the live person (Dead). The change event could be extended to account for oobject creation, object destruction and object change

[2] http://www.loa.istc.cnr.it/ ontologies/DUL.owl

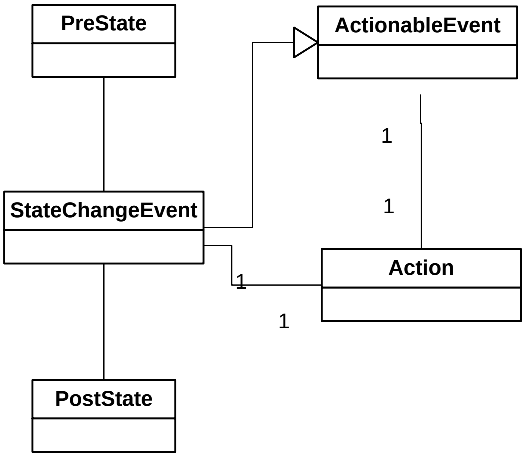


Figure 9. State Change Event

Event Participants: The model ensures that each participant is assigned a unique role (with respect to the other participant of that same event. The {OR} notation ensures that no two distinct roles are ever assigned to the same participant and the multiplicity ensures that each participant is assigned some role.

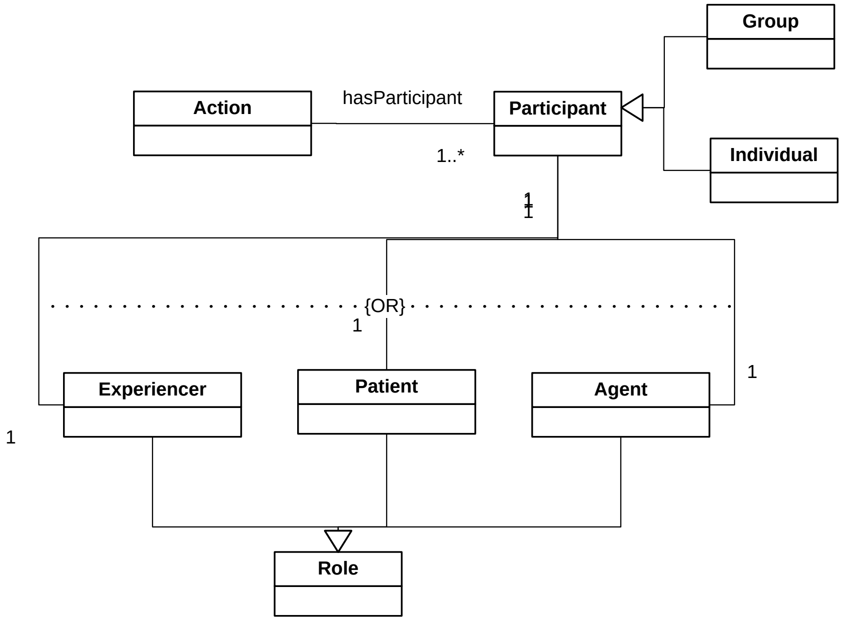


Figure 10. Participants and Roles

The participant may be an individual or a group for group actions such as a patrol.

Event-Time: The model support defining two time types. A time object is either a definite time period (accomplishment) or a definite time instant (achievement).

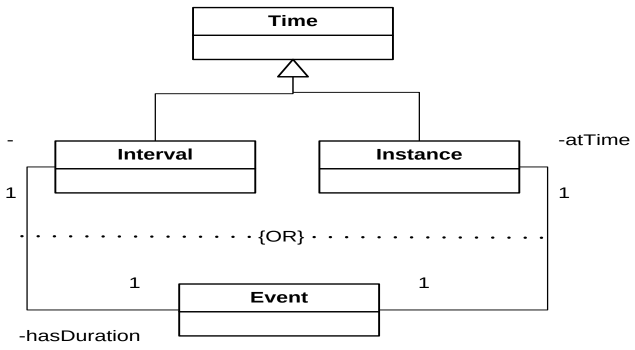


Figure 11. Event Time structure

Event-Location: The model supports defining an abstract location and spatial locations. Abstract locations are used to represent mythical or virtual location. The model allows one region for each atomic events.

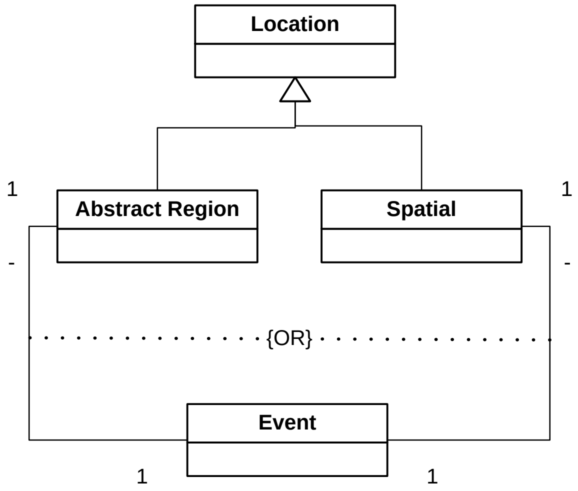


Figure 12. Event Location Structure

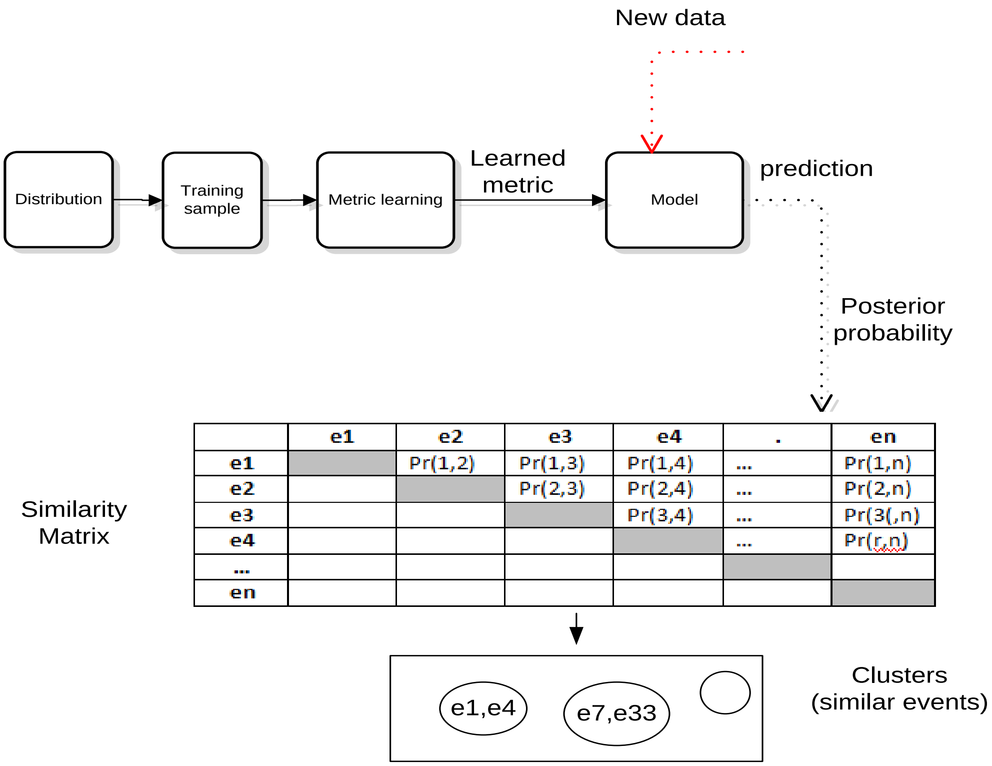
In the model location is assigned directly to event and not to the participant location. This is different from The F—A Model [Scherp et al,. 2009] which links the location of the event with the location of the object. As in the example of the wave, molecules of water change continuously making it difficult to use objects location as the event location.

Window Manager: The window manager use a delta-based time sliding window model to manage the list of events available for comparison with the new incoming event. Window (W) is defined as

W = { $\mathrm{TS}_{\mathrm{i-b+1}}, \ldots, \mathrm{TS}_{\mathrm{i-1}}, \mathrm{TS}_{\mathrm{i}}$}, where TS_{i} is the latest time slot and $\mathrm{TS}_{\mathrm{i-b+1}}$ is the first time slot in the window and the first to be evicted when the time shifts by b to the new slot TS_i . The eviction policy is controlled by the variable (b). The window expels the oldest tuples of events to maintain the window size when new tuples are coming in.

Similarity Manager: It allows to manage the similarity measures and the learned similarity predictor. On the one hand, similarity measures are not tied to each other or to the similarity predictor and can be changed or replaced as long as they keep the same interface with the similarity predictor. Each local similarity measure could be implemented locally or could use an external source to calculate the similarity. Similarity manager handles the output and input to each similarity manager and liase with other components such as the predictor and the window manager.

Learning Module: The Learning module realizes the actual learning functionality. It includes single modules capturing the functionality of particular learning algorithms and a module for computing different forms of retrieval errors required by the learning algorithms.



	e1	e2	e3	e4	.	en
e1	▨	Pr(1,2)	Pr(1,3)	Pr(1,4)	...	Pr(1,n)
e2		▨	Pr(2,3)	Pr(2,4)	...	Pr(2,n)
e3			▨	Pr(3,4)	...	Pr(3(,n)
e4				▨	...	Pr(r,n)
...					▨	
en						▨

4-8 Supervised Learning process

Knowledge Base: To improve the system recall and be able to reason about event types and their spatial and time components. We use right now a very simple ontology for crimes which is derived from WordNet for testing purposes. A complete event knowledge base on events (ABOX and TBOX) is planned to be done. Also we use a region knowledge base for populated places with their region connection relations from RCC8 .

5 Implementation and Evaluation Setup

The main objective of the evaluation is to measure the performance of the classifiers with respect to the given data from the event domain. The evaluation setup considers the measures and metrics used to evaluate the performance of the classifiers as well as data sampling, re-sampling, model selection and assessment.

In the context of detecting past event in a continuous monitoring system, there are multiple criteria for assessing the performance of the classifiers:

1. Does the classifier produce stable results?
2. Is it sensible to variations of new data?
3. How robust is the classifier to noisy data?
4. What is the tradeoff between speed and precision or recall ?
5. Is the classifier efficient in terms of speed?

In this chapter we first discuss evaluation measures (5.1). Then, we present the data and ground truth setup in (5.2).

5.1 Evaluation measures

The first and probably the most important measure we need to consider is the generalization performance of selected model and its capability to predict using independent test data. The quality of a good estimator is measured using the production error test which ideally should be measured on multiple independent data sets. The prediction error following the approach in [Hastie, et al. 2009] for the dependent variable Y and a vector of the covariates X and a prediction model g(x) that has been estimated from a training T and loss function

$$\mathrm{L(Y, g(x)\ ) = (\ Y - g(x)\ )^2}$$

The error

$$\mathrm{Err(x) = E[(\ Y - g(x)^2]}$$

This error may then be decomposed into bias and variance components:

$$Err(x) = (E[g(x)] - \ f(x))^2 + \ E\ [g(x) - E(g(x)]^2 + \ \sigma^2$$

$$Err(x) = Bias^2 + Variance + Irreducible\ Error$$

The third term, irreducible error, is the noise term in the true relationship that cannot fundamentally be reduced by any model. The ideal case is to reduce the bias and variance to zero, however this is not possible with any model, therefore there is a tradeoff between minimizing the bias and minimizing the variance.

Definition 5.1 Bias. From the above equation we can define the bias as the difference between the expected prediction of our model and the correct value which we are trying to predict.

$$bias = E[g(x)] - f(x)$$

Definition 5.2.Variance: The variance is how much the predictions for a given point vary between different realizations of the model. The variance is the measure of how much a single estimator deviates from the average estimator over multiple datasets. Usually and due to lack of multiple datasets to run the model and get the average, resampling methods are used to run the model multiple times. The general idea of resampling is to partition the data set into separate partitions and to use the training set to fit model parameters, and use the testing set to assess the prediction accuracy .

$$variance\ = E\ [g(x) - E(g(x)]^2$$

In practice, when the model complexity increases, the variance gradually increases. Additionally, as model complexity increases, the squared bias decreases. Thus there is a tradeoff between bias and variance that comes with model complexity: models that are too complex will have high variance and low bias; models that are too simple will have high bias and low variance. The best model will have both low bias and low variance.

5.1.1 Resampling methods

Resampling methods are used to handle different data set problems in machine learning problems. [Estabrooks et al., 2004] used multiple resampling methods to handle class imbalance problems where the dataset contains many more examples of one class than the other. [Breiman, 1996] used Bagging (**b**ootstrap **agg**regat**ing)** resampling technique to create different training data subsets which are randomly drawn with replacement from the original base dataset. The obtained training data subsets (bags) are used then to train different models.

Another way to do resampling is to use cross-validation. Cross-validation has different variants:

- **k-fold Cross-Validation**[3]: This variation is useful when the class-distribution of the data is skewed. It ensures that the distribution is respected in the training and testing sets created at every fold. This would not necessarily be the case if a pure random process were use.

- **Leave-One-Out In k-fold Cross-Validation**: each fold contains m/k data points where m is the overall size of the data set. In Leave-one-out, k =m and therefore, each fold contains a single data point

In this thesis, we used 5-fold cross validation. Cross validation was selected because we found that the distribution of the event data is right skewed, therefore for resampling we used k-fold cross validation.

Measuring skewness

There are several methods to test the distribution of the data such as visual inspection of data plots, skew, kurtosis, and P-P plots give researchers information about normality, and Kolmogorov-Smirnov tests provide inferential statistics on normality. Outliers can be identified either through visual inspection of histograms or frequency distributions, or by converting data to z-scores.

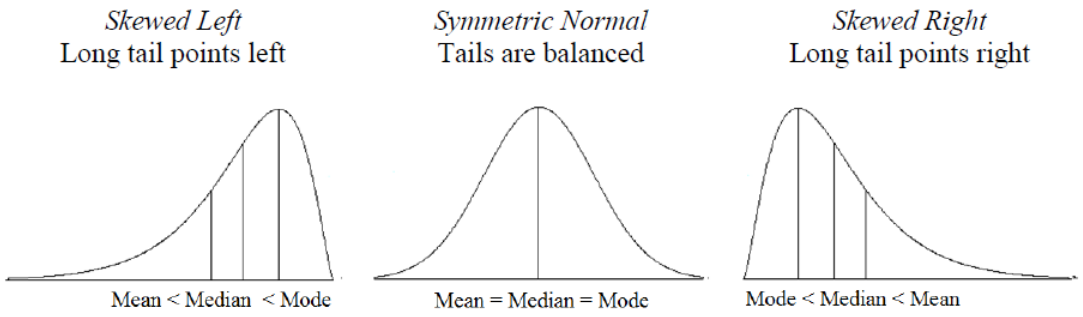


Figure 1. Sketches showing general position of mean, median, and mode in a population.

Figure 5-1 Right and left data Skewness

Adapted from [Doane et al., 2011]

[3] stratified cross-validation, where the class (category) representation in each block is same (or close) to that in your 'full' data set.

5.1.2 K-Fold Cross-validation

In a regularized logistic regression, lambda (λ) is a free parameter that needs to be tuned empirically to penalize models with extreme parameter values in order to prevent high variance (overfitting). Lambda (λ) is tuned usually by cross-validation. We describe, here the procedure we used to tune lambda (λ):

The basic idea in k-fold cross-validation is to start by sorting the dataset randomly and then to split the data into k folds. A common value of k is 5 or 10, so in the case of 5, we would divide the data into five partitions, and run 'k' rounds of cross-validation. In each round, we use one of the folds for validation, and the remaining folds for training. After training the classifier, we measure its accuracy on the validation data. We Average the accuracy over the k rounds to get a final cross-validation accuracy.

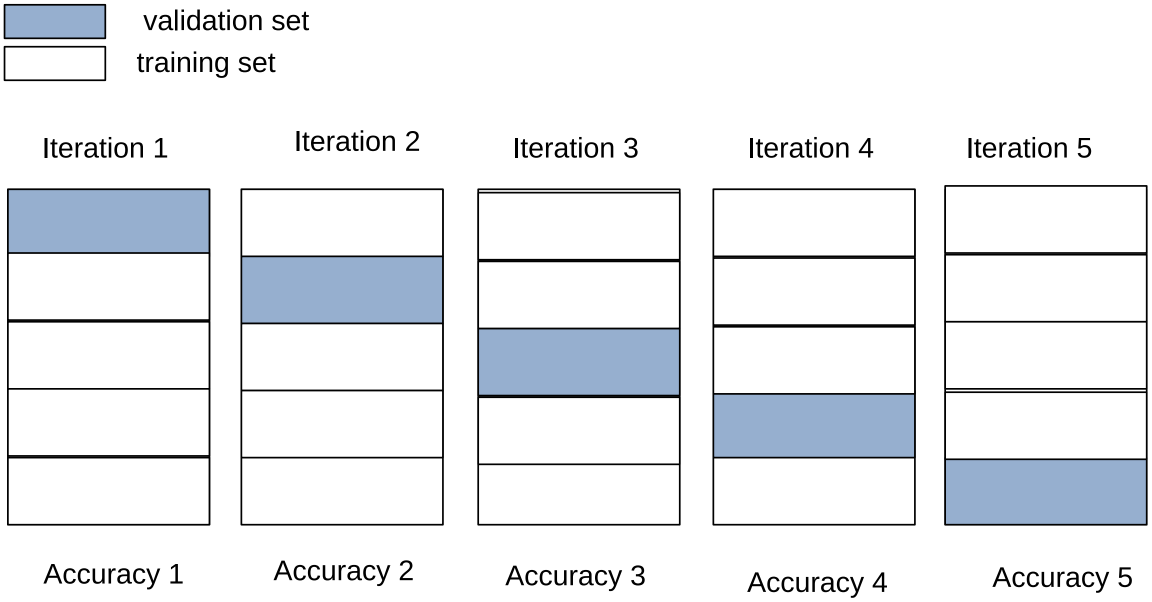


Figure: 5-fold cross-validation. The data set is divided into 5 folds. The validation accuracy is computed for each of the 5 validation sets, and averaged to get a final cross-validation accuracy.

```
1.  Add random column $\mathrm{Col}_{\mathrm{rand}}$ to the data set X
2.  Sort X using $\mathrm{Col}_{\mathrm{rand}}$
3.  Select number of folds K
4.  Divide sorted data (X) into equal subsets or folds $X = \{F_{1+}F_{1+} \ldots + F_k\}$

            numberOfRowsPerFold = dataRowNumber / crossValidationFolds;
5.  For each fold assign testRows and trainRows
6.  Select empirical values for lambda $\lambda \in \{ \lambda_1, \lambda_2 + \cdots + \lambda_n\}$
7.  For each tuning parameter $\lambda$
    8.  Calculate $g_\lambda^F(x)$ ; $F \in trainRows$
    9.  Test the model using testRows ; and calculate the error for current value of $\lambda$

                    $Err_k(\lambda) = \sum_{i \in} (y_i - g_\lambda^F(x))^2$
10. For each value of λ
    11.  Calculate the average error over the k-folds

                    $CV(\lambda) = \frac{1}{n} \sum_{k=1}^{K} Err_k(\lambda)$
12. Choose λ that that give minimum CV(λ)
```

We used MATLAB in our analysis. The result of the k-fold cross-validation with k= 5 to select the best of value of the regularization parameter is shown in figure 7-1.

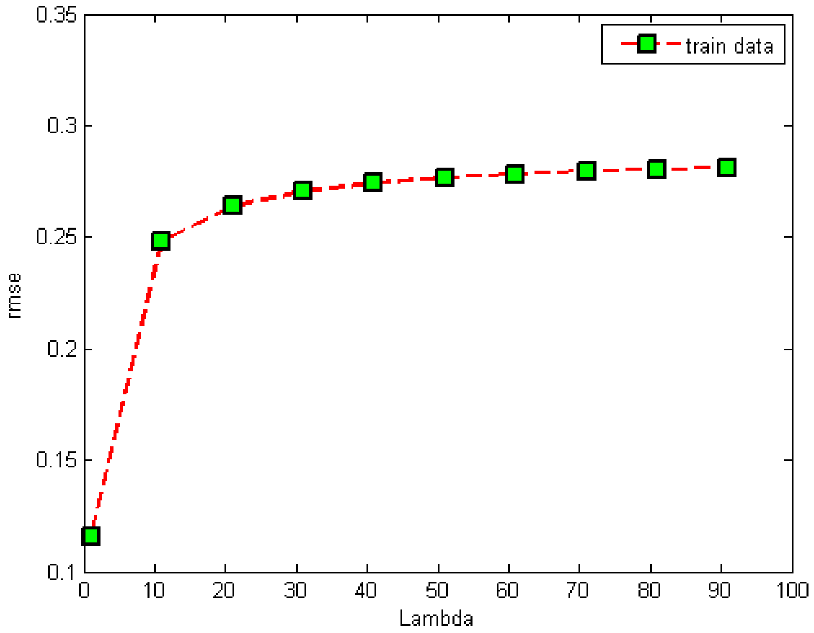


Figure 5-2 Lambda best value

5.1.3 Learning Curves

Learning curves are visual method to recognize when a model has high bias or high variance. A learning curve is a plot of the training and cross-validation error as a function of the number of training example. These plots can give a quantitative view into how beneficial it will be to add training samples.

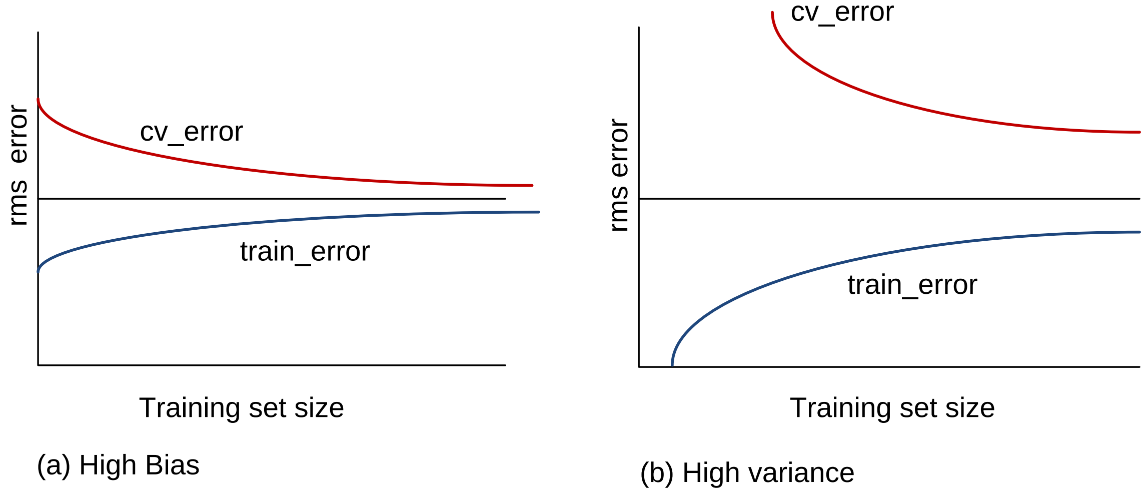


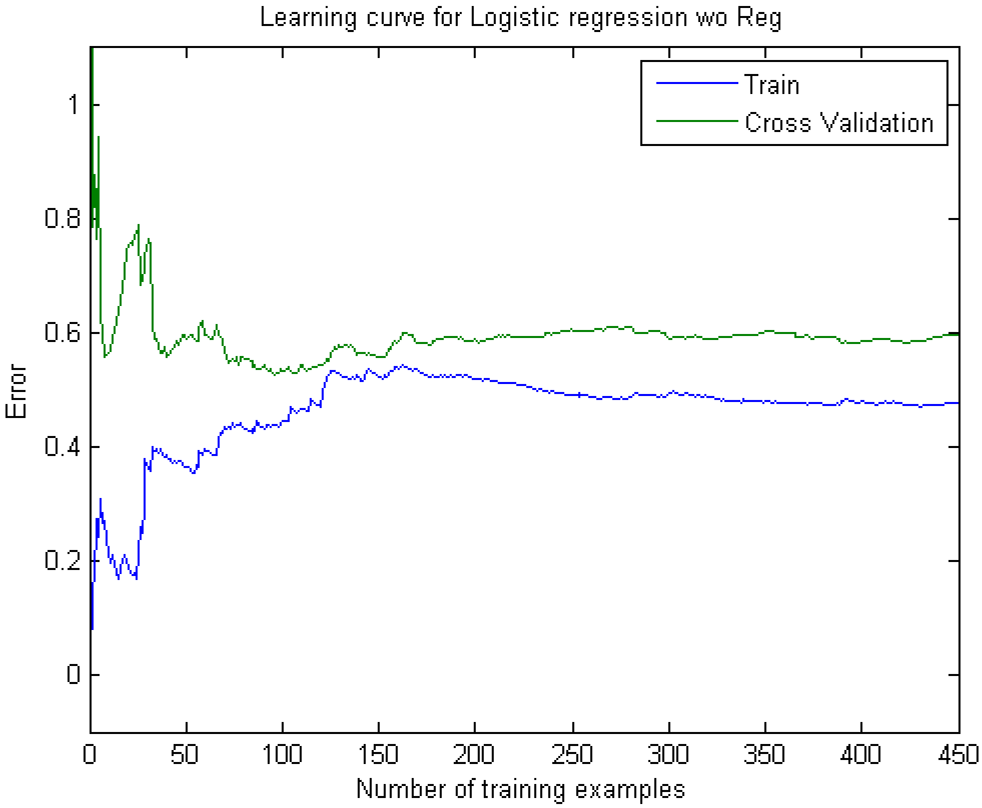
Figure 5-3 Learning curves for high bias and high variance

In Figure 5-2 (a), the curve indicates a high-bias where the estimator under-fits the data. This is indicated by the fact that both the training and cross-validation errors are very high. As we add more training example, both curves have converged to a relatively high error.

In Figure 5-2 (b), As we add more samples to this training set, the training error will continue to climb, while the cross-validation error will continue to decrease, until they meet in the middle. adding more data will allow the estimator to very closely match the best possible cross-validation error.

Accuracy: Accuracy is the overall correctness of the model and is calculated as the sum of correct classifications divided by the total number of classifications.

Train Accuracy:	84.554455
Test Accuracy:	83.809524



.

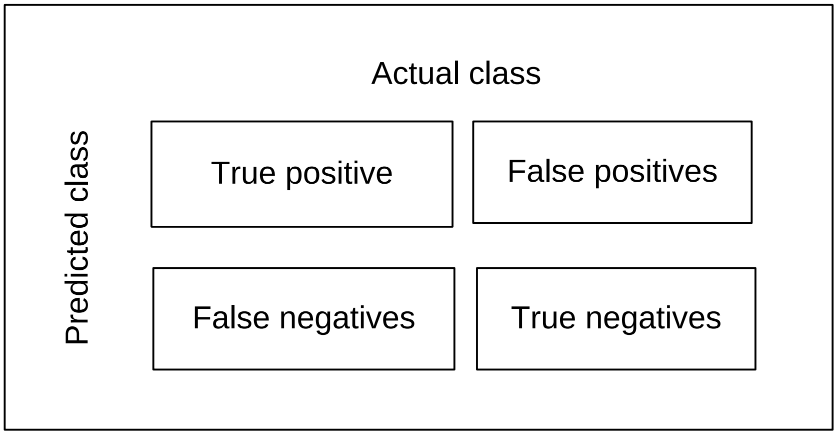
5.1.4 ROC Curve

The performance measures used to evaluate the performance of the classifier are precision, recall, and ROC curves. Definitions of the performance measures used are summarized below. The same performance measures are used to evaluate the results of the baseline experiments

The ROC curves are useful to visualize and compare the performance of classifier methods .The receiver operating characteristic (ROC) curve is a two dimensional graph in which the false positive rate is plotted on the X axis and the true positive rate is plotted on the Y axis.. There are four possible outcomes from a binary classifier:

- **true positive (TP)**: predicted to be positive and the actual value is also positive
- **false positive (FP)**: predicted to be positive but the actual value is negative
- **true negative (TN)**: predicted to be negative and the actual value is also negative
- **false negative (FN)**: predicted to be negative but the actual value is positive

These four outcomes are usually arranged in a confusion matrix as shown in figure 5-3



	Actual class	
Predicted class	True positive	False positives
	False negatives	True negatives

Figure 5-4 The confusion matrix

From the confusion matrix we can calculate the following

- **Recall:** Precision is a measure of the accuracy provided that a specific class has been predicted. It is defined by:

$$Recall = \frac{tp}{tp + fn}$$

- **Precision**: Recall is a measure of the ability of a prediction model to select instances of a certain class from a data set. It is commonly also called sensitivity, and corresponds to the true positive rate. It is defined by the formula:

$$Precision = \frac{tp}{tp + fp}$$

- **true positive rate**

$$TPR = \frac{positives\ correctly\ classified}{total\ positives}$$

- **false positive rate:**

$$TPR = \frac{negatives\ incorrectly\ classified}{total\ negatives}$$

- **false negative rate**

$$FNR = \frac{negatives\ incorrectly\ classified}{total\ negatives}$$

- **Specificity:** Recall/sensitivity is related to specificity, which is a measure that is commonly used in two class problems where one is more interested in a particular class. Specificity corresponds to the true-negative rate.

$$specificity = \frac{True\ negatives}{false\ positives + true\ negatives}$$

$$= 1 - FPR$$

As explained by [Fawcett, 2006] one point in ROC space is better than another if it is to the northwest (tp rate is higher, fp rate is lower, or both) of the first. Classifiers appearing on the left-hand side of an ROC graph, near the X axis, make positive classifications only with strong evidence so they make few false positive errors, but they often have low true positive rates as well. Classifiers on the upper right-hand side of an ROC graph may be thought of as ‘‘liberal’’: they make positive classifications with weak evidence so they classify nearly all positives correctly, but they often have high false positive rates. In 6-1, A is more conservative than B. Many real world domains are dominated by large numbers of negative instances, so performance in the far left-hand side of the ROC graph becomes more interesting. The diagonal line y = x represents the strategy of randomly guessing a class

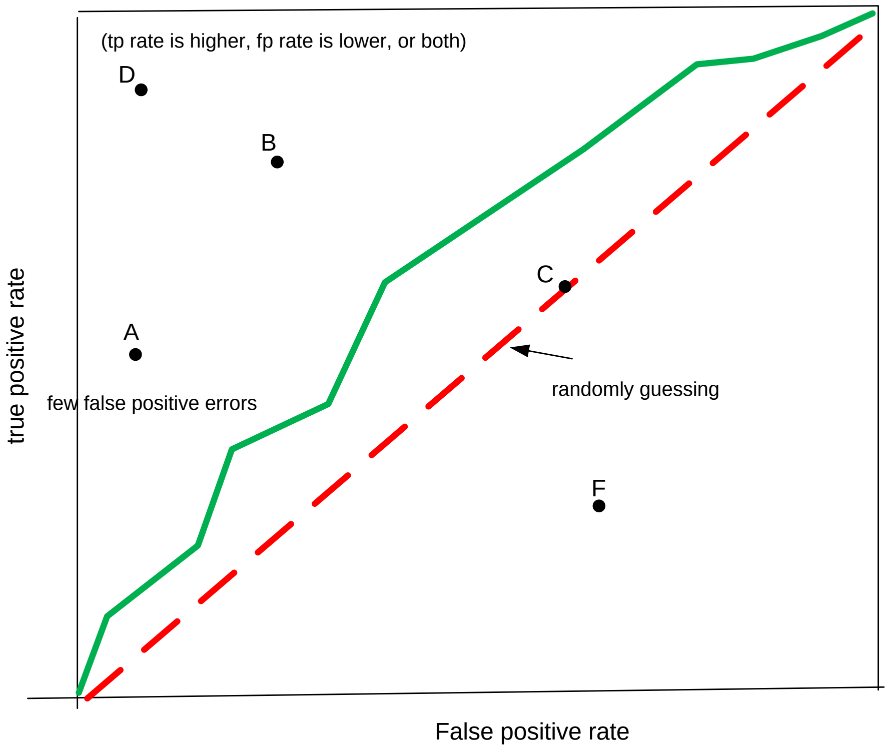


Figure 5-5 ROC graph with discrete classifiers

5.1.5 Control Parameters

The efficiency of gradient descent and support vector machines algorithms depends on different control parameters including

- Initial weight vector theta
- Learning rate alpha
- Reduction rate
- Regularisation parameter C

Initialisation of Feature Weights: In the gradient descent algorithm, it is not guaranteed that a global minimum of the error function will be found. Actually, the algorithm may converge towards a global minimum or to a local one. The convergence depends on the initial values of the weight parameters. Choosing different values for the initial weights may lead to different outcomes of the gradient descent search. Basically, the options for choosing the initial values depends on our knowledge of the shape of the error function or the intelligence of a domain expert. In this thesis, we opt to choose a uniform weight-vector for the weights, rather than depending on a domain expert.

$$\forall i \; \theta_i = 0$$

Learning rate alpha: In order for Gradient Descent to work we must set the α (learning rate) to an appropriate value. This parameter determines how fast or slow we will move towards the optimal weights. If α is very large we will skip the optimal solution. If it is too small we will need too many iterations to converge to the best values. There are many strategies to assign the value of α depending on the approach of the gradient descent: batch (BGD) or stochastic (SGD) . For BGD, the strategies to choose α are summarized by the following approaches:

- **Fixed Learning Rate:** The learning rate is chosen by trial and error. It can be kept constant across all epochs, or it can be decreased gradually as a function of the epoch number. Choosing α by experience might be problematic and tedious work.

- **Adaptive Learning Rate:** At each iteration of the gradient descent, start from the learning rate alpha = 0 and gradually increase alpha by the fixed step delta Alpha = 0.01, for example or try increasing the learning rate by 5%. Recalculate parameters theta and evaluate the cost function. Since the cost function is convex, by increasing alpha (that is, by moving in the direction of negative gradient) cost function will first start decreasing and then (at some moment) increasing. If the error rate is increasing (meaning that it skipped the optimal point), we should reset the values of theta to the values of the previous iteration and decrease the learning rate by 50%. This technique is called Bold Driver[4]. In case that the cost function never starts increasing, stop at alpha = 1.

- **Line search method**: Line search is a method which chooses an optimal learning rate for gradient descent at every iteration, which is better than using fixed learning rate throughout the whole optimization process. Optimal value for learning rate alpha is one which locally (from current theta in the direction of the negative gradient) minimizes cost function.

The Stop-Predicate: Since there is no guarantee that the gradient descent will terminate, there are different approaches to select a stop-predicate [Wilke and Bergmann, 1996]:

Maximal Number of Optimisation Iterations: The algorithm stops when it complete an N number of iterations implemented using a for loop. N being a threshold defined manually.

Number of failed Optimisation Iterations: Another criterion that can be used to break off the optimisation process is a repeated failure of optimisation steps. If the algorithm runs repeatedly without any improvement for N times, then we assume the algorithm reaches its best optimization and no further iterations are needed..

Minimal Improvement of Similarity Error: This also based on a threshold value of minimum error achieved . The algorithm stops when the error is small and does not reduce further as the number of iterations increases.

SVM Control parameters

SVM classification as presented in chapter 2 uses the variable to C to enforce that all slack variables are as close to zero as possible. If we recall the optimization objective, the goal is to find w and b that minimize:

[4] Lecture notes at http://www.willamette.edu/~gorr/classes/cs449/intro.html

$$\tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i{}^2$$

$$\text{subject to: } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i$$

C is essentially a regularisation parameter, which controls the trade-off between achieving a low error on the training data and minimising the norm of the weights **w.**

Finding an appropriate value for C is a vital step in the use of SVMs. The parameter C enforces an upper bound on the norm of the weights, which means that there is a nested set of hypothesis classes indexed by C. As we increase C, we increase the complexity of the hypothesis class, therefore C plays a role in limiting the complexity of the hypothesis class. A common practice is to use cross-validation to select best value of C.

In our experiment, we followed the same approach of k-fold cross validation for choosing lambda to select the value of C.

5.2 Data and Ground Truth Setup

All the experiments in this thesis are conducted on real-data collected and gathered from the field. The data was reviewed by experts and annotated based on human experience and interpretation. In this section, we describe the process of collecting the data and the process of building the ground truth for our experiments.

5.2.1 Annotating Events

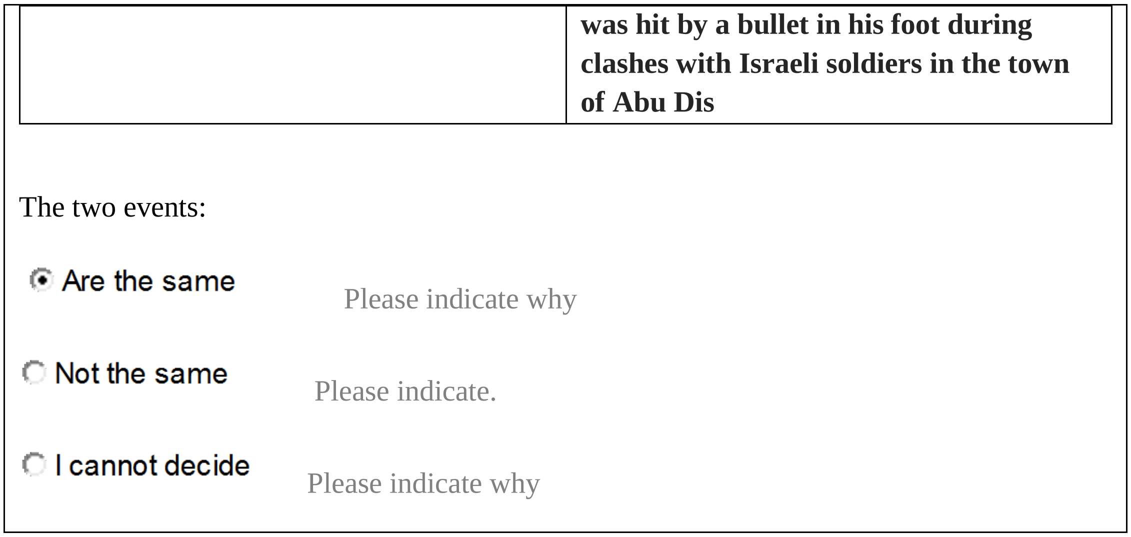
From a sample of 6,000 records, we asked a group of 9 users to look at sub-sets of the this sample and try to identify duplicate events. Deliberately, some of the records were distributed across all sub-sets to be able to measure how different people cognitively think of what is a duplicate event. Users were also given the chance to say 'I cannot judge' and if possible to give the reason for that. We designed a simple GUI to collect the feedback of the annotators as shown in figure (5-5)

Experiment [5]#1

Please for each pair of events answer the set of questions listed below:

On 03 Apr 13, 0930 hrs **Israeli Soldiers Injure Teen in Clashes near Jerusalem**	On 03 Apr 13, 0900 hrs **A 17-year-old student was injured Thursday when he**

[5] The original GUI is in Arabic and this is the translation of the original one.



	was hit by a bullet in his foot during clashes with Israeli soldiers in the town of Abu Dis

The two events:

● Are the same — Please indicate why

○ Not the same — Please indicate.

○ I cannot decide — Please indicate why

Figure 5-6 The basic interface with which our workers label each query

We repeated this experiment 15 times [10 pairs per experiment], making the total set of annotated pairs equals to 150 pair. Then we analyzed the results before completing the set to 1000 pair. The aim of doing this into two phases is to evaluate the quality of annotation and to evaluate the agreement between annotators. Because there might be a need to enhance the user interface or give more information to annotators.

Experiment Setup

- **Selection of pairs:** it is unusual that a person uses interchangeably too far concepts when describing an event. For example, it is rare to mix between fire event and arrest event, however a user easily mix between an arrest and detain events. Therefore, we selected the pairs that are mostly used interchangeably from our dataset of events. For one experiment the result is shown in the table 5-1

Table 5-1 Sample of Annotated events by end users

First Event Id	Second Event Id	Match Count (frequency)	No Match (frequency)	Cannot Decide (frequency)
217	218	5	2	2
219	220	1	4	4
221	222	9	0	0
223	224	4	4	1
225	226	0	9	0
227	228	4	1	4
229	230	4	3	2
231	232	6	0	3
233	234	0	4	5
235	236	5	2	2

…				

- **Analysis of the results:** We can examine whether two different annotaters agree among themselves by using the *Cohen's Kappa statistic* (or simply *kappa*) which is intended to measure agreement between two raters. A common practice [Landis et al, 1977] is to state that Kappa values over 0.61 indicate substantial levels of agreement, while values over 0.81 represent almost perfect agreement. However in our experiment we are more interested in evaluating the narrative description of which each rater expresses the reason behind her judgment.

Reasons for differences between annotators

- **Scope of similarity .** What qualifies as a "match" in the context of events ?

- **lack of information about places .** Since the annotators are from different cities, lack of knowledge about common names for point of interests used in the event description, may not help them to reason about the similarity of events.

E1	E2
On 01 Apr 13, 0830 hrs, protesters forced their way through the main gates of UNRWA Jabalia Relief Office and staged a sit-in tent at the yard protesting against the cash aid cuts. The protest ended at 1200 hrs, but the tent remained in place	On 01 Apr 13, 0900 hrs, approximately 50 people from the SHC families organized a sit-in at the yard of UNRWA Beach Relief Office, west of Gaza City, protesting against the financial aid cuts. At 1030 hrs another group of SHC families staged a sit-in inside the Beach Distribution Center and forced the employees out. The sit-in ended at 1640 hrs
Palestinian refugees on Monday rallied outside an UNRWA ration office south of Gaza in protests at reducing their financial support from the agency and suspending the emergency and unemployment programs	On 31 Mar 13, 0930 hrs, approximately 40 people from the SHC families organized a sit-in at UNRWA Jabalia Relief Office protesting against the cut of financial aid. The protestors forced the employees out of their offices. The protest ended peacefully at 1130 hrs

Not using Full reasoning. Some annotators use part of the available knowledge to decide if the events are similar or not. If the type of the event and location are the same, then they judge that this pair is 'the same' without taking time into the reasoning process.

Coverage of time and location. Some differences between annotators were due to disagreement about location coverge . When an event contains a term like ‘near place A’ and the second event uses a term like ‘in place B’ . Some annotators do some kind of linkage between nearby places and some do not. The same is valid for time coverage such ‘in the early morning’ and ‘at 6 AM’.

Missed information. Some annotators consider the two events are not the same, and some say ‘cannot decide’. Those who say cannot decide attribute their choice to missed information.

Discriminative criteria. When two events occurs into two far locations such as two non-neighbor cities, we didn’t find any disagreement between annotators. The same reason is also valid for time.

The final version of the annotated dataset was subject to quality assurance and reviewed by one senior expert

5.2.2 Handling Missing Data

Almost any experiment suffers from incomplete or missing data. Majority of software tools do not accept missing data or have options to generate the missing data automatically based on user selection. Most approaches to complete missed data are summarized by the following:

1. Replace missing values with column averages (i.e. replace missing values in feature 1 with the average for feature 1).
2. Replace missing values with column medians.
3. Impute missing values using the other features.
4. Remove records that are missing features.
5. Use a machine learning technique that uses classification trees, e.g. random forests, boosted trees, bagged trees, etc.

However, one should be careful before selecting one of the above options to complete the missing data, since in many scenarios completing the data has no meaning and may mislead the computation. Missing data is classified into three categories [Graham, 2009] [Little and Rubin, 1987] [Howell,2012]:

- Missing completely at random (MCAR)
- Missing at random (MAR
- Missing Not at Random (MNAR)

The main consequence of MCAR missingness is loss of statistical power. The good thing about MCAR is that analyses yield unbiased parameter estimates (i.e., estimates that are close to population values). MAR missingness (i.e., when the cause of missingness is taken into account) also yields unbiased parameter estimates. The reason MNAR missingness is considered a problem is that it yields biased parameter estimates[Graham, 2009]

The definition of each type is summarized from [Howell, 2012]

Missing completely at random (MCAR): MCAR is perhaps the easiest to understand. If the cases for which the data are missing can be thought of as a random sample of all the cases, then the missingness is MCAR. This means that everything one might want to know about the data set as a whole can be estimated from any of the missing data patterns, including the pattern in which data exist for all variables, that is, for complete cases. When we say that data are missing completely at random, we mean that the probability that an observation (X_i) is missing is unrelated to the value of X_i or to the value of any other variables

Missing at random (MAR): The data can be considered as missing at random if the data meet the requirement that missingness does not depend on the value of X_i *after controlling for another variable*. For example, people who are depressed might be less inclined to report their income, and thus reported income will be related to depression. Another way of saying this is to say that to the extent that missingness is correlated with other variables that are included in the analysis, the data are MAR.

The randomness in MAR missingness means that once one has conditioned on (e.g., controlled for) all the data one has, any remaining missingness is completely random (i.e., it does not depend on some unobserved variable).

Missing not at random (MNAR). If data are not MCAR or MAR then they are classed as **Missing Not at Random (MNAR).** For example, if we are studying mental health and people who have been diagnosed as depressed are less likely than others to report their mental status, the data are not missing at random. Clearly the mean mental status score for the available data will not be an unbiased estimate of the mean that we would have obtained with complete data. The same thing happens when people with low income are less likely to report their income on a data collection form. When we have data that are MNAR we have a problem. The only way to obtain an unbiased estimate of parameters is to model missingness Although statisticians prefer not to

Handling missing data at Random

- The simplest approach--listwise deletion.

By far the most common approach to missing data is to simply omit those cases with missing data and to run our analyses on what remains. Thus if 5 subjects in group one don't show up to be tested, that group is 5 observations short. Or if 5 individuals have missing scores on one or more variables, we simply omit those individuals from the analysis. This approach is usually called listwise deletion, but it is also known as complete case analysis.

Although listwise deletion often results in a substantial decrease in the sample size available for the analysis, it does have important advantages. In particular, under the assumption that data are missing completely at random, it leads to unbiased parameter estimates. Unfortunately, even when the data are MCAR there is a loss in power using this approach, especially if we have to rule out a large number of subjects. And when the data are not MCAR, bias results. (For example when low income individuals are less likely to report their income level, the resulting mean is biased in favor of higher incomes.)

- Imputation

Hot deck replaces a missing value by imputing it from a randomly selected similar record. One form of hot-deck imputation is called "last observation carried forward" which depends on creating a ordered dataset, then finds the first missing value and uses the cell value immediately prior to the data that are missing to impute the missing value.Another form is Cold-deck imputation, where the value could be selected from a diiferent data set. Imputation also include other techniques such as regression and stochastic regression techniques.

- Mean substitution

The idea of substituting a mean for the missing data has a couple of problems. In the first place it adds no new information. The overall mean, with or without replacing missing data, will be the same

- Regression substitution

If we don't like mean substitution, why not try using linear regression to predict what the missing score should be on the basis of other variables that are present? We use existing variables to make a prediction, and then substitute that predicted value as if it were an actual obtained value. This approach has been around for a long time and has at least one advantage over mean substitution. At least the imputed value is in some way conditional on other information we have about the person

Handling missed data in the event dataset

While gathering event data, we faced the problem of having a (MNAR) records. Mainly missed data was related to the event agent [the doer of the event]. It is quite frequent that the information about the agent is not available and if available the number of records are small compared to the rest of the data. Unfortunately, due to its nature we cannot model the data and we cannot delete the records having this information missed. Therefore, the model was built by ignoring this feature.

If the records miss some important information like the value of the similarity measures which is due to some computation error, we deleted these cases. The percentage of deleted cases are not more than 2% of the total sample.

6 Evaluation results

This chapter provides evaluation results for the two approaches introduced in chapter 5 which are logistic regression and support vector machines. The results are measured based on the evaluation measures introduced in chapter 6.

First we present the results for logistic regression, then we provide a comparison between the predictors based on the two approaches.

6.1 Evaluation of the similarity model

A logistic regression model was fit to the event data to explain the predicted odds of similarity . The model main selected predictors are

$$\begin{aligned}\textbf{Predicted logit}(\mathbf{y}=\mathbf{1})\\ &= \beta_0 + \beta_1 * Sim_{lin} + \beta_2 * Sim_{path} + \beta_3 * Sim_{wup} + \beta_4 * samePatient\\ &+ \beta_5 * casusePropertyDamage + \beta_6 * casueInjury + \beta_7 \ * fatal\\ &+ \beta_8 * Sim_{time} + \beta_9 * Sim_{location}\end{aligned}$$

The three variables 'lin', 'path','wup' represent event-type similarity, where 'Human','Agent','Gender','Effect' represent the thematic role similarity. When performing regression analyses we would like to characterize how the value of some dependent variable changes as some independent variable $x$is varied.

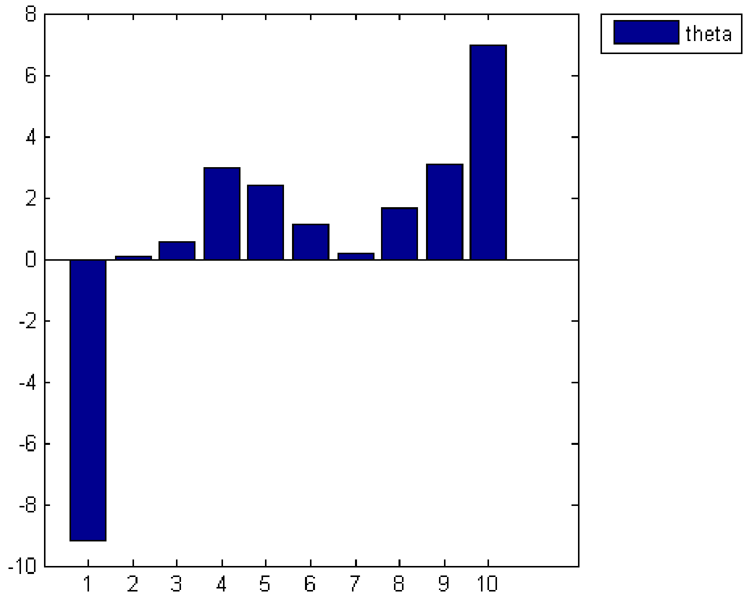


Figure 6-1 Coefficients using Logistic Regression Model

The coefficients obtained for our 9 features are presented in Table (6-1) along with other statistical values such as odds ratio.

Table 6-1 Features and their Coefficients

						Odds
Predictor	**theta**	**SE**	**Wald**	**df**	**p-value**	**ration**
constant	-9.6529	0.7888	149.7628 1	0	0.0000	N/A
Type similarity						
lin	0.5362	0.7234	0.5493	1	0.4586	1.7094
path	3.1842	0.7286	19.0994	1	0.0000	24.1484
wup	2.5193	0.303	69.1534	1	0.0000	12.4201
Participant						
samePatient	2.5193	0.303	69.1534	1	0.0000	12.4201
Cause & Effect						
propertyDamage	1.1407	0.3479	10.7485	1	0.0000	3.129
causeInjury	0.1711	0.4252	0.1619	1	0.6874	1.1866
causeFatal	1.7152	0.4326	15.7236	1	0.0001	5.558
Time feature						
Time similarity	3.2448	0.4947	43.0266	1	0.0000	25.6569
Location features						

Location similarity	7.349	0.6419	131.0894	1	0.0000	1554.699

Our primary interest is to test our hypotheses regarding the coefficients β . *The change in* the coefficient has the following meaning

$$\beta = 0 \;\Rightarrow P(presence) is\ the\ same\ at\ each\ level\ of\ x$$

$$\beta\ > 0 \;\Rightarrow P(presence)\ increases\ as\ x\ increases$$

$$\beta < \ 0 \;\Rightarrow P(presence)\ decreases\ as\ x\ increases$$

6.1.1 Overall Model Evaluation

A logistic model is said to provide a better fit to the data if it demonstrates improvement over the intercept-only model (no predictors).Such an improvement could be examined using some inferential statistics which may include: the likelihood ratio, Score, and Wald tests [Peng et al.,2002]. We examined the overall performance of the model using the Wald test.

Recall that if $\beta_i = 0$, then the predictor has no bearing on the probability of success or failure. The Wald Test is used to test the hypotheses test on coefficient β_i, with

$$H_0 : \beta_i = 0$$

$$H_1 : \beta_i \neq 0$$

the t-statistic is:

$$t = \frac{\beta_i}{Se(\beta_i)}$$

where $Se(\beta_i)$ is the standard error of the estimated coefficient β_i,which is found using the theory of maximum likelihood (which involves taking second partial derivatives of the log-likelihood function). The Wald test for all coefficients is presented in table 7-2.

Table 6-2 Coefficients Wald Test

Predictor	**theta**	**Wald**
constant	-9.6529	149.7628 1
Type similarity		
lin	0.5362	0.5493
path	3.1842	19.0994
wup	2.5193	69.1534

Participant		
samePatient	2.5193	69.1534
Cause & Effect		
propertyDamage	1.1407	10.7485
causeInjury	0.1711	0.1619
causeFatal	1.7152	15.7236
Time feature		
Time similarity	3.2448	43.0266
Location features		
Location similarity	7.349	131.0894

based on these results, we can rejects the null hypothesis.

6.1.2 Evaluating individual predictors

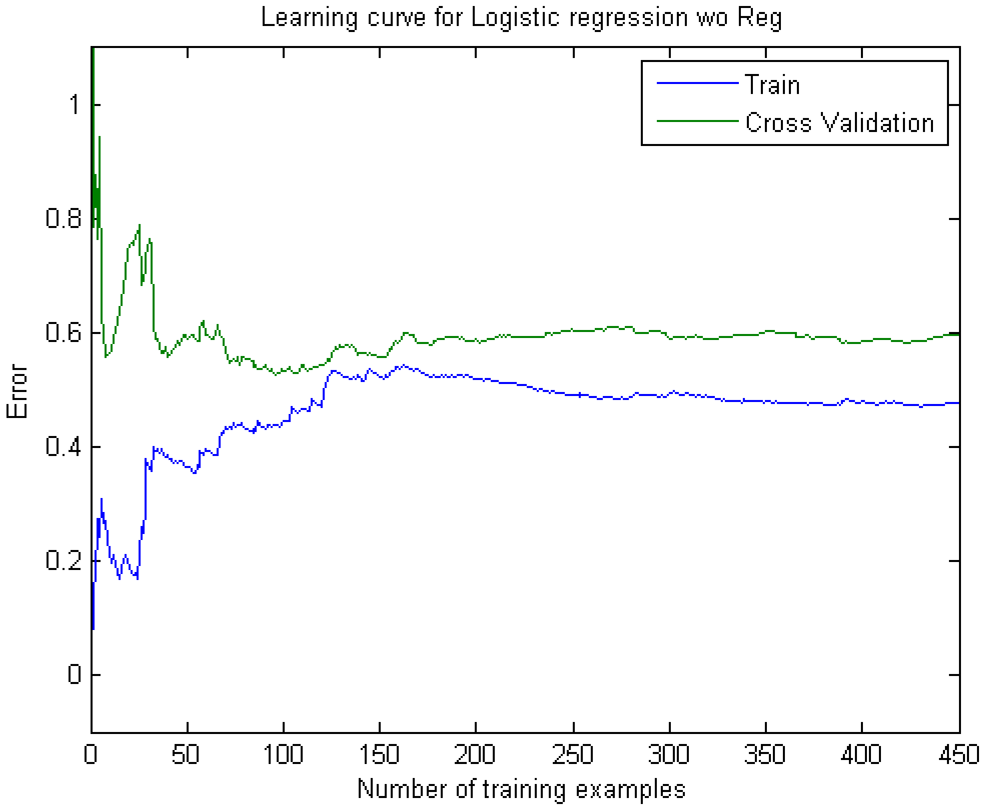
We noticed that the Wald test for significance of the coefficients for lin similarity and causeInjury the p-values as shown in table below are not significant.

Predictor	**theta**	**SE**	**Wald**	**df**	**p-value**	**Odds ration**
Type similarity						
lin	0.5362	0.7234	0.5493	1	0.4586	1.7094
Cause & Effect						
causeInjury	0.1711	0.4252	0.1619	1	0.6874	1.1866

The general approach in this case is to remove these two variables from the full model and use a compact model instead.

6.1.3 Bias and Variance evaluation

Learning Curvers



6.1.4 Evaluating model performance

To measure the degree to which predictions agree with the data, we can show this degree using a receiver operating characteristic (ROC) curve or an overlay plot of sensitivity and specificity versus predicted probabilities.

The ROC curve is a plot of sensitivity versus (1 – specificity). Sensitivity is defined as the proportion of observations correctly classified as an event (true positive fraction). Specificity is defined as the proportion of observations correctly classified as nonevent. Therefore (1-specificity) means the proportion of observations misclassified as an event or the false positive fraction.

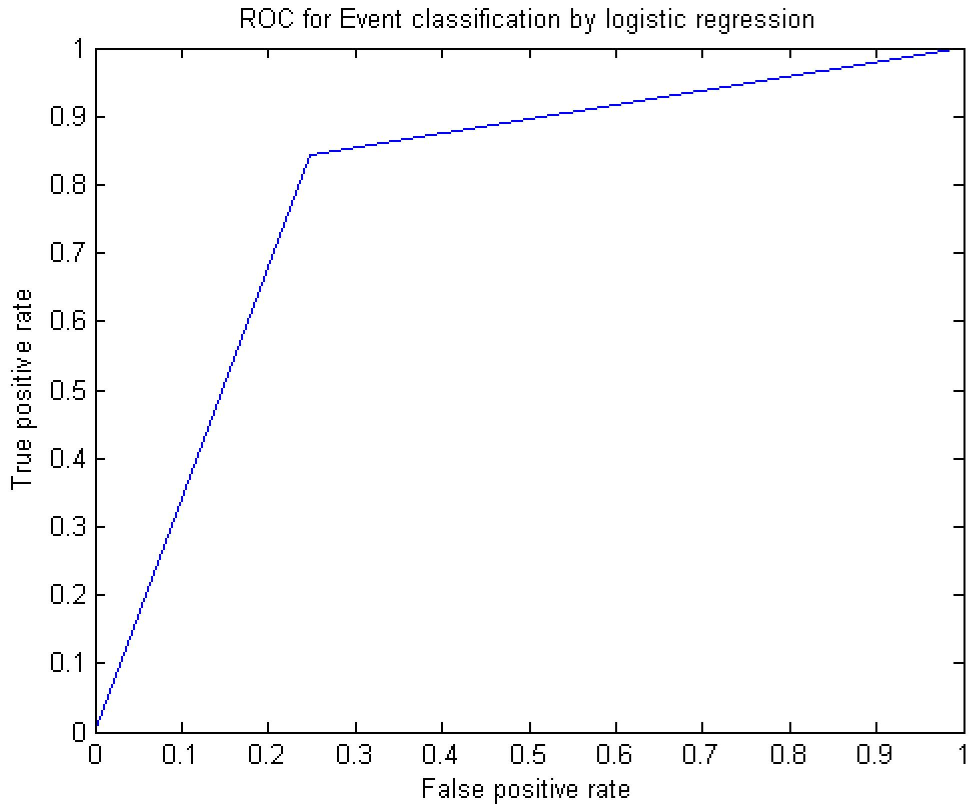


Figure 6-2 Receiver operating characteristic curve . Area under ROC curve =0.7984

Usually the ROC curve demonstrates several things:

1. It shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity).
2. The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test.
3. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate is the classification.
4. The area under the curve is a measure of accuracy.

We found that the area under the curve equals to 0.7984 which indicates the ability of the model to correctly classify similar and non-similar pairs. Ideally, we would like to reach the top left corner, since there sensitivity is 100% and specificity is 100%. The closer we approach this point, the better the model.

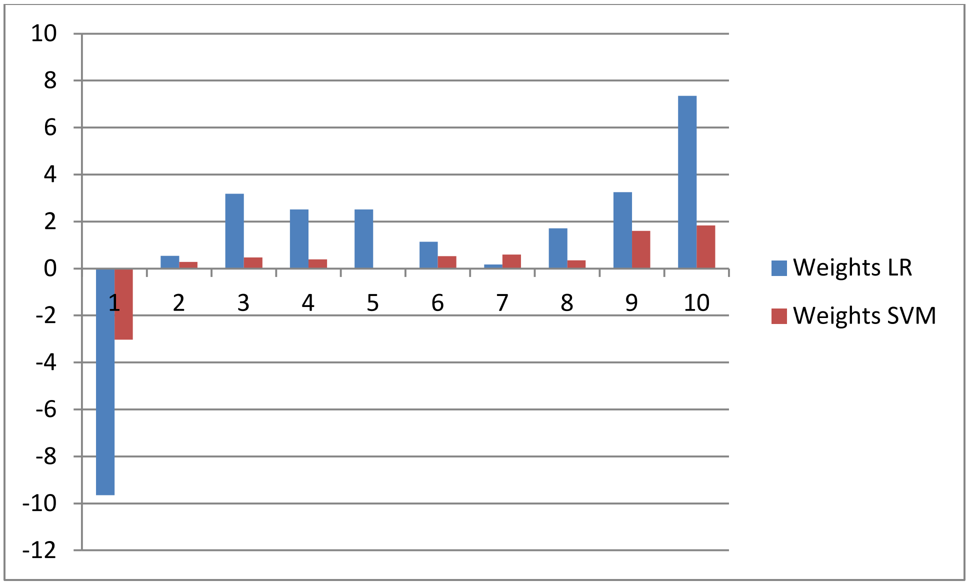
Another method to look at the performance of the model is to calculate the precision and recall

6.1.5 Summary of results

<table>
<tr><th></th><th>Logistic Regression</th><th>Support Vector Machines</th></tr>
<tr><td>The confusion matrix</td><td><table><tr><td>90</td><td>31</td></tr><tr><td>19</td><td>97</td></tr></table></td><td><table><tr><td>103</td><td>19</td></tr><tr><td>35</td><td>81</td></tr></table></td></tr>
<tr><td>Recall</td><td>0.8256</td><td>0.74</td></tr>
<tr><td>precision</td><td>0.7438</td><td>0.84</td></tr>
<tr><td>f-score</td><td>0.7923</td><td>0.79</td></tr>
<tr><td>AUC</td><td>0.7900</td><td>0.77</td></tr>
</table>

Coefficient s

Predictor	**LR**	**SVM**
constant	-9.6529	-3.03326
Type similarity		
lin	0.5362	0.281181
path	3.1842	.473282
wup	2.5193	0.388652
Participant		
samePatient	2.5193	-0.02145
Cause & Effect		
propertyDamage	1.1407	0.53005
causeInjury	0.1711	0.592395
causeFatal	1.7152	0.343762
Time feature		
Time similarity	3.2448	1.607563
Location features		
Location similarity	7.349	1.834504



6.1.6 Analysis of Results

- **False positive when the location is “far “ or “disconnected”**

We are interested to find out how many cases are classified as similar when the location is far or disconnected. In the Logistic regression model, there are 31 false positive cases among these, we found “3” cases that are classified false positive when the location is “far” or “disconnected”. However, In the SVM model, the number of cases are “5”.

The system gives false positive with probability = 0.65663, when the location is far or disconnected when the following minimum conditions are met :

- Type similarity is high
- Time similarity is high
- sameAgent is true

The probability increases to “0.973475” when all other conditions have their maximum values :

- Type similarity =1 ;
- sameAgent is true
- causePropertyDamage is true
- causeInjury is true
- causeFatal is true
- Time similarity equals “1”

We also found the the system gives false positive when the time location similarity is around 0.5 given that the following two conditions are met:

- Time similarity is very high ~ 1
- Type similarity is very high ~1

False negative when location and time are similar (threshold > 0.5)

The System gives false negative when the following conditions are met :

Case # 1 < time and location are around 0.5 >

- Type similarity = ‘1’
- Time = 0.5
- Location is less than 0.57
- and given that all other conditions are not similar, however if one of the other similarity measures are true, the model gives a correct prediction.

Case # 2.
When

- time similarity is 1
- location similarity is 1
- and all other measures are false or zero

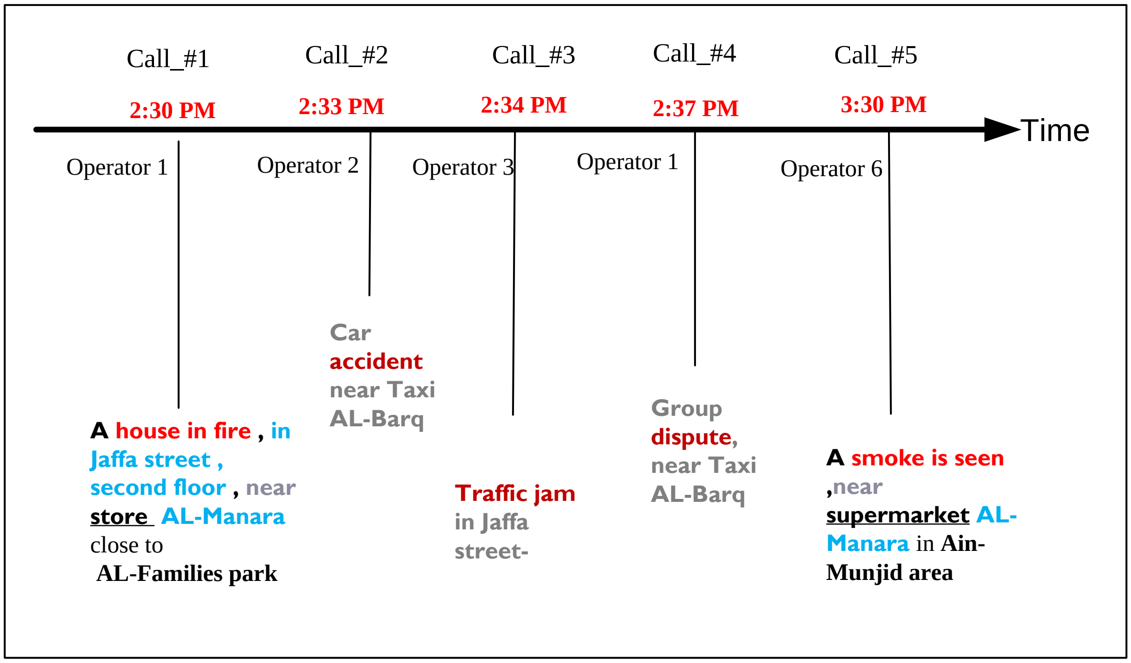
the system classify the two events as similar with probability of “0.705731”, this indicates that for two events to be classified as similar while all other features are not similar the time and location similarity should be very high.

To classify two events as similar, given the event type is not similar while the time and location are similar then one of the other features at least is required to be similar such as samePatient.

The following values gave similar result

- Type =”0”
- Time=”0.6”
- Location =”0.7”
- sameAgent is true.

Based on the above results, let is consider evaluating how the system handles the events given on the introductory example in section 1.1, which is shown again in figure (6-2).



Call_#1
2:30 PM
Call_#2
2:33 PM
Call_#3
2:34 PM
Call_#4
2:37 PM
Call_#5
3:30 PM
Time
Operator 1
Operator 2
Operator 3
Operator 1
Operator 6
A house in fire , in Jaffa street , second floor , near store AL-Manara close to AL-Families park
Car accident near Taxi AL-Barq
Traffic jam in Jaffa street-
Group dispute, near Taxi AL-Barq
A smoke is seen ,near supermarket AL-Manara in Ain-Munjid area

Figure 6-3 A test case for evaluating the similarity model

In this scenario we have the following event types:

Event type	Event type description
Fire	the event of something burning (often destructive))
Car accident	(an unfortunate mishap; especially one causing damage or injury
Traffic jam	the aggregation of things (pedestrians or vehicles) coming and going
Dispute	(the speech act of disagreeing or arguing or disputing)
Smoke	a natural phenomenon

For simplicity, we consider a time window of one hour and assign the minimum value given by the temporal similarity calculation using the simple center of gravity method 6 to all events. The result is shown in table (6-3).

[6] Code is given in Appendix-I

6-3 Similarity prediction results for the sample scenario

Fire	accident	0.4914	0.166667	0.705882	0	0	0	0	0.9	0.5	0	0	0.337999
Fire	Traffic	0.092284	0.166667	0.461538	0	0	0	0	0.9	0.5	0	0	0.191021
Fire	Dispute	0.288793	0.142857	0.625	0	0	0	0	0.9	0.5	0	0	0.279261
Fire	Smoke	0	0.166667	0.285714	0	0	0	0	0.9	1	1	1	0.819834
Traffic	accident	0.087439	0.090909	0.375	0	0	0	0	0.9	0.75	1	1	0.500392
Traffic	Dispute	0.092598	0.1	0.4	0	0	0	0	0.9	1	1	1	0.861609
Traffic	Smoke	0	0.2	0.333333	0	0	0	0	0.9	0.5	0	0	0.139766
accident	Smoke	0	0.058824	0.2	0	0	0	0	0.9	0.5	0	0	0.091515
accident	Dispute	0.272455	0.1	0.526316	0	0	0	0	0.9	1	1	1	0.902443
Dispute	Smoke	0	0.083333	0.210526	0	0	0	0	0.9	0.5	0	0	0.095349

As shown, the model is able to predict all the cases correctly in this scenario. Results are shown in the last three column as follow

Human label	Model prediction	Probability of y=1 or similar
0	0	0.337999
0	0	0.191021
0	0	0.279261
1	1	0.819834
1	1	0.500392
1	1	0.861609
0	0	0.139766
0	0	0.091515
1	1	0.902443
0	0	0.095349

7 Related Work

We describe relevant related work in two areas: event detection and tracking in scial streams and event similarity metric learning.

7.1 Related Work

In the literature, the event detection task is classified into new event detection (NED) and retrospective event detection (RED) techniques [Yang and Pierce, 1998]. The retrospective event detection is defined as the discovery of previously unidentified events, while the on-line detection or new event detection considers the discovery of new events from live feeds in real-time. Both approaches consider content similarity and temporal or spatial proximity.

[Balazinska et al., 2007] propose a system that automatically compares events on streams and identifies past events similar to newly detected events. The system is called Moirae and uses three measures to calculate the similarity. The similarity is based on the notion of context which is any additional information obtained through a set of queries associated with the event itself. For example, for an overloaded server event, the set of processes and resources running at the time of the event is called the context. The three similarities are
1. Entity similarity. If the aspects of two different event contexts contain the same entities (i.e., tuples with the same keys), the contexts are similar and the distance between these contexts should be small.
2. Value similarity. If the aspects of two different event contexts contain entities with similar attribute-values, the contexts are similar and the distance between these contexts should be small.
3. Prioritizing entities with abnormal values. When comparing event contexts, entities with abnormal values should be prioritized over other entities

The technique used in Moirae is to treat each context as a document, where the tuple attribute values correspond to terms, then they measure the similarity between contexts by measuring their cosine similarity. With the cosine similarity metric, two contexts are similar to each other if they contain a larger number of the same "terms".

To monitor real-life events such as disasters over social media [Dittrich and Lucas, 2014] use a system to classify Twitter events based on a hierarchical tree structure (taxonomy) of natural disaster types. In this structure, the leaves represent disaster types, where each of these leaves is assigned a bag-of-words (BoW) that is virtually unambiguous for the specific event type. The union of all BoWs of the child nodes represent the BoW for the respective parent node, e.g. the BoW for the type Hydrological is the union of the BoWs of Flood and Tsunami. In total they collected 133 general disaster terms derived from investigations of tweets from past events. The system starts at the topmost level of the taxonomy and calculates for each node the classification score, i.e. the ratio of the number of identified terms that belong to the BoW of the node, and the total number of tweets in the respective cell and minute. A threshold of at least 0.3 is set to assure the relevance of the identified keywords in the current Twitter content.

Only the child node with the highest value is further analysed in an analogous manner. In case of two or more equal values as well as if all child nodes fail to reach the threshold, the parent node is set as type of the event. For example, if the system decided for Hydrological in the preceding level, but cannot distinguish between Flood and Tsunami based on the identified keywords, it will classify the event as Hydrological. To identify the location of events, a pre-defined areas were selected based on potential types of events. Each area is assigned a geographical extent as a grid.

To identify breaking news events in near real-time from Twitter data [Meyer et al., 2011] developed a system to identify breaking news topics from users' tweets. The topics are pre-defined and selected. The topics were grouped into three categories: natural events, man-made events, or other uncategorized events. For example, a natural event includes "tornado", "earthquake", and "hurricane". A man-made event includes "riots", "protest", and "arson". Uncategorized events include other relevant topics (such as "terrorism"). Then, for each topic, applicable synonyms are manually identified. For instance, the topic "tornado" utilizes the synonyms "twister" and "funnel".

To identify the topic they used document frequency (DF) weighting to calculate the number of occurrences of a given term within the entire batch of tweet posts. If a topic's DF weight is high enough, it is stored as a geospatial, real-time breaking news topic occurrence. The DF weight is a combined sum of all occurrences of the topic term and its synonyms within the batch of data. The threshold for the weight was s determined through empirical studies. The geospatial and temporal information for each tweet is collected through the Twitter APIs.

The approach followed by [Meyer et al., 2011] relies on getting the location of the event from the tweet API, which relies on whether the user is willing to share the location or not. To handle the sparsity of geo-enabled features in these services and enable new location-based personalized information services [Cheng et al., 2010] propose a probabilistic framework for estimating a Twitter user's city-level location based purely on the content of the user's tweets, even in the absence of any other geospatial cues. Their approach depends on utilizing the location cues provided by the user in the content of the tweet such as specific place names or certain words or phrases more likely to be associated with certain locations than others. The approach depends on using per-city word distributions and based on maximum likelihood estimation, the probabilistic distribution over cities for word (w) was used to decide if that tweet is issued by a user located in a city. This approach depends on finding local words for each city. Local words could be basket ball team or special words only used by people of a city. The granularity of this approach is a grid of cities.

To detect events at a finer granularity (street or building level [Xie et al., 2013] consider detecting hyper-local events using social media content alone. They trained a Gaussian Process Regression (GPR) time-series prediction models for each geo-region G for the time span T. The output of GPR models for each time t and location G represents the predicted mean value for volume and associated standard deviation for volume. The prediction for a certain region serve as volumes of data they expect to observe given no event is happening for that region at a certain time. If the number of posting by unique users for a geolocation G at time t exceeds the prediction by some threshold, the alert engine would mark that time period t as a candidate event for that location G. Since a deviation does not necessarily mean a true event they also trained a classifier to review the features of a candidate event and mark it as true or false**.** For the spatial features, their assumptions is that the latitude and longitude information of all items is given and based on that they compute three geographic distribution features for each candidate event.

To tackle the problem of grouping content available in social media applications such as Flickr and Youtube into clusters of documents describing the same event [Reuter et al., 2011] consider this problem as a record linkage task. To compute the similarity between a pair of documents or images they used the same set features used by [Becker et al., 2010] . Mainly to calculate the time similarity between a pair of documents they used the following similarity measures

$$sim_{time}(d_1, d_2) = 1 - \frac{|t_1 - t_2|}{y}$$

Where t_1 and t_2 are date/time and y is the number of minutes of one year. For the specific . If t_1 and t_2 are more than year the time similarity is set to zero. To compute the similarity between two locations they used Haversine distance using the latitude-longitude pairs:

$$sim_{geo}(d_1, d_2) = 1 - \ H(L_1, L_2)$$

Where H is the Haversine distance.

Geographical co-ordinates similarity measures

Working on Event-based Classification of Social Media Streams [Becker et al., 2010] [Reuter et al., 2011] use latitude-longitude pairs to compute the similarity between two locations using the Haversine distance

$$sim_{geo}(d_1, d_2) = 1 - H(L_1, L_2)$$

Where H is the Haversine distance.

The approach to convert latitude/longitude coordinates into actual places, is to divide the place of interest whether it is a city or state or a continent to a grid [Meyer et al., 2011]. However since the data of latitude-longitude is not always available, a similarity of 0 is assumed [Reuter et al., 2011]

[Makkonen et al., 2004] used class-wise comparison to compare two event documents. They assigned semantic classes to the terms in each document based on the four basic questions in news article: who (NAMES), what (TERMS), when (TEMPORALS), where (LOCATIONS). For each class they composed a sub-vector and do the comparison between sub-vectors before combining the results using a weighed sum of the similarity measures for the results of the four sub-vectors. To compare the sub-vectors of NAMES and TERMS they used term-frequency inverted document frequency and calculated the similarity for each pair using the cosine between the two sub-vectors for each class. Temporal similarity is based on comparison of intervals of each document. For each pair of intervals from TEMPORAL vectors X and Y they determined the maximum value. Then similarity is the average of all these maxima. For location comparison, they split the locations into a five-level hierarchy: Continent, region, country, administrative region, and city. The administrative region can be replaced by mountain, seas,lakes, or river and they represent the location using a tree.

Similarity between two locations, x and y is based on the length of the common path:

$$\mu_s(x, y) = \frac{\lambda(x \cap y)}{\lambda(x) + \lambda(y)}$$

Where $\lambda(x)$ is the length of the path from the root to the element x.

A remarkable notice stated by [Makkonen et al., 2004] is that semantic augmentation degraded performance, especially in topic tracking because in their opinion this is partially due to inadequate spatial and temporal similarity functions.

[Becker et al., 2010] used classification-based and ensemble-based techniques to address the problem of identifying events that are reflected in set of social media documents which are associated with events and to correctly assign the documents that correspond to each event. They deal with unknown events (un-typed events) and cast the problem of identifying events and their associated social media documents as a both a clustering problem and then as classification problem, where each cluster should correspond to one event and consist of all of the social media documents associated with that event.

In the clustering approach, they have created a separate cluster for each feature such as title, description, tags, location, and time and then used a ensemble clustering approach to combine the individual partitions in a single cluster.

7.2 Summary

We can summarize the approaches represented in this chapter by the following:

Event Representation: mainly events streamed from social media are represented as documents. Similarity between two events is carried between two documents using the cosine similarity.

Typed-events: some approaches use pre-defined terms for event types [Dittrich and Lucas, 2014]

Pre-defined locations: As in [Cheng et al., 2010] each city is allocated a set of local words to distinguish the city from another or as followed by [Dittrich and Lucas, 2014], where they assigned a set of event types to a certain areas.

Latitude-Longitude based location: Some approached rely completely on the assumption that the lat-lon is given in the event content. However, missing geo-tagged data is common because usually this data is provided by the end users and not all users are willing to share their current locations. Furthermore, this approach suffers from noise due to the fact that many users tweet about an event while they are not in the lactation of that event.

Granularity: Previous efforts mostly focused on detection of global events detected in global or country level. Few attempts were found to handle hyper-local events.

8 Conclusions

This last chapter discusses the results presented in this thesis and points out some open questions that could not be answered in the scope of this work. These open questions can be seen as interesting issues for future research

8.1 Objectives and Achieved Results

The main objective of this work is the development of a framework to detect hyper-local duplicate events in the near past. The matching framework employs different similarity measures which are learned using machine learning techniques. The approach we adopted is generic and can be employed in different scenarios and applications. However, our focus was on the requirements of multi-tier actionable agencies, where we continuously monitor streamed events and evaluate them before taking any decision or action.

Employing learning techniques leads to several major advantages:

- Flexibility of using local similarity measures. Local similarity measures are interfaced by the system can could be replaced or enhanced without affecting the performance of the system.

- Extraction of similarity is automated and results could be cached for future usage.

The main contribution of this thesis is represented in chapter 3 and chapter 4. Mainly the work on identifying suitable and adequate similarity measures for each element of the observed event. In essence, this thesis includes the following important contributions:

- Provides adequate type, spatial, temporal and thematic role similarity functions. The design of these similarity functions considers similarity knowledge combined from cognitive point of a view as well as functional point of view. Similarity measures in addition to semantic similarity and relatedness considers:
    - Location relations (topology, orientation and direction)
    - Temporal relations (linguistic terms and fuzzy intervals)
    - Causes and effects
    - Agent
    - Patient
    - Functions
    - Participant features

    - Instruments
    - manner

- A thorough study to different existing similarity measures: semantic and relational similarity between words or pairs of words (event types). We analyzed the adequacy of existing similarity measure for the task of learning the weights of event types . For other aspects or facets of the event, we discussed the concept of similarity from numerous view-points and their computational approach, in particular, the alignmenet model, transformational model and relational model, of similarity.

- A computation framework to calculate similarity is presented using supervised approach to learn the weights of local similarity functions. Mainly similarity between pairs of events are learned using logistic regression and support vector machines.

- The core principle of our proposed approach is based on decoupling the agreements between the parties involved whether they are observers, lodgers, consumers of events from different feeds or re-producer of the events to other receiving channels. Decoupling is designed to take place for

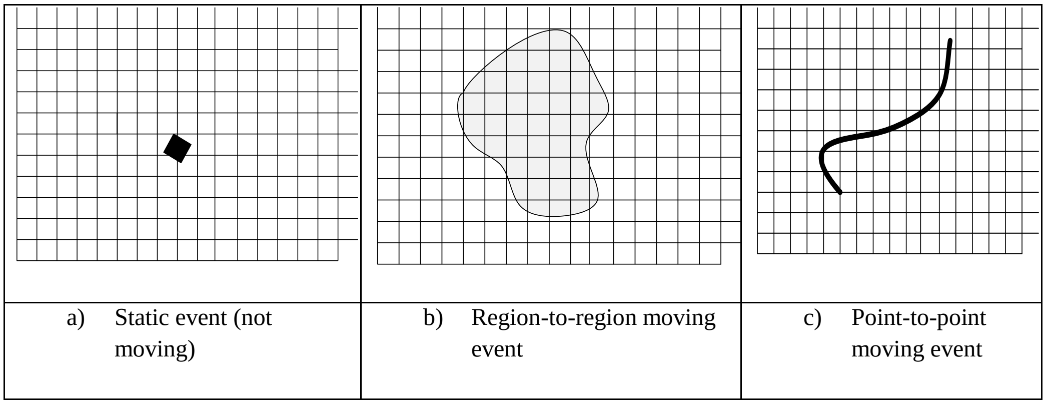
8.2 Open Questions and Future Research Directions

As already mentioned, the work presented in this thesis represents a general approach to learning event similarity measures. It introduced a basic framework and methodology, and two concrete learning algorithms. However, there are still several open aspects that represent interesting research issues for future work.

- **Learning similarity between sequence of events**

The model proposed in this thesis is designed to learn the similarity between single events (atomic). However, in real-life we also have scenarios where complex events (a chain of sub-events) that occur in a certain order and we need to know the similarity between two sequences of events, or more generally we might need to know the similarity between sub-sequences of events. The proposed model, could be extended to combine similarity across sequences. For example, temporal similarity needs to be extended to accommodate for new temporal terms such immediately-before, before and after. The spatial similarity needs to reflect the evolving areas as well. Spatial relations need to be extended to utilize the role of the location. Additional specifications that should be supported:

- Locating events: Events in terms of their location are classified as:
  - Static events: events occur in a place and remain in the same place e.g. car accident
  - Moving events: where moving events could be (a) point-to-point movement (b) region-to-region movement e.g. demonstration and patrol movement



a) Static event (not moving)

b) Region-to-region moving event

c) Point-to-point moving event

8-1 Static vs moving events

Note: Events with Moving regions . The RCC relations are not static ; C(x,y) doesn't hold all the time. It is possible that a forest is destroyed or its area shrink ; a new bridge might partially overlap an existing region

Role of location: Also the model should support assigning a role to a location. As there are some type of events where the start, end location is needed. Other roles as passing through should be also supported.

Event direction: For a moving event, the direction of the event might be specified.

- **Partial similarity search and query**

Adding additional capability to find similarity between sequences and sub-sequences, will also enable decision makers to query on partial similarity. One possible solution after portioning the sub-sequences is to cluster the sub-sequences and query on the clusters.

- **Modeling Vague and Vernacular Places**

Vague and Vernacular Places such as “city center”, South of the country”, “north region” and many others are not usually part of a gazetteer resources or geo-spatial Ontology. Encoding knowledge of vague and vernacular places is as important as official names and their boundaries in the hyper-local event domain. People refer and communicate about a place based on their experience. It is common that an event observer use a name of public organization as the name of the street. For example, we found that many people refer to the radio street as ‘Finance street’ because the Ministry of finance is located in that street. Therefore a better matching algorithm for event location is needed whenever vague and vernacular places are used by end users.

Bibliography

[Alqadah and Bhatnagar, 2011] F. Alqadah and R. Bhatnagar. (2011).Similarity measures in formal concept analysis. Ann. Math. Artif. Intell., 61(3):245{256

[Barua et al., 2014] Aditi Barua, Lalitha Snigdha Mudunuri, and Olga Kosheleva, Journal of Uncertain Systems Vol.8, No.x, pp.xx-xx, 2014 Online at: www.jus.org.uk

[Balazinska et al., 2007] M. Balazinska, Y. Kwon, N. Kuchta, and D. Lee.(2007). Moirae: History-enhanced monitoring. In Proc. of the Third CIDR Conf., Jan.

[Balcan, 2008] M.-F. Balcan, A. Blum, and N. Srebo, "A theory of learning with similarity functions," Machine Learning, vol. 72, no. 1, pp. 89–112, 2008.

[Banerjee and Pedersen, 2002] Satanjeev Banerjee and Ted Pedersen.(2002). An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet, Lecture Notes In Computer Science; Vol. 2276, Pages: 136 - 145, 2002. ISBN 3-540-43219-1

[Banu and Chandrasekar, 2012] Banu A F, and Chandrasekar C (2012). A survey on deduplication methods, International Journal of Computer Trends and Technology, vol 3(3), 364–368

[Becker et al., 2010] H. Becker, M. Naaman, and L. Gravano. (2010). Learning similarity metrics for event identification in social media. In Proceedings of the 13rd ACM International Conference on Web Search and Data Mining

[Ben-Hur and Noble, 2005] Ben-Hur A, Noble WS.(2005). Kernel methods for predicting protein-protein interactions. Bioinformatics. (Suppl 1):i38–i46.

[Bilenko and Mooney, 2003] M. Bilenko and R.J. Mooney.(2003). "Adaptive Duplicate Detection Using Learnable String Similarity Measures," Proc. ACM SIGKDD, pp. 39-48

[Blachon et al.,2007] Sylvain Blachon, RuggeroG. Pensa, J. B. C. R. J.-F. B., and Gandrillon, O. (2007). Clustering formal concepts to discover biologically relevant knowledge from gene expression data. In Silico Biology 7:467–483.

[Breiman, 1996] Breiman, L. (1996). Bagging Predictors. Machine Learning 24(2), 123--140

[Brunner et al., 2012] Brunner C, Fischer A, Luig K, Thies T. (2012). Pairwise support vector machines and their application to large scale problems. J Mach Learn Res 2012, 13:2279-2292

[Budanitsky and Hirst, 2006] Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of semantic distance. Computational Linguistics 32(1), 13–47 (2006)

[Chen and Chen, 2003] Si Jay Chen. Shyi Meng Chen.(2003). Fuzzy risk analysis based on similarity measures of generalized fuzzy numbers, IEEE Trans. Fuzzy Systems 11, pp. 45-56

[Cheng et al., 2010] Cheng, Z.; Caverlee, J.; and Lee, K. (2010). You are where you tweet: A content-based approach to geo-locating twitter users. In CIKM ’10.

[Chen, 1996] Shyi Meng Chen. (1996). New methods for subjective mental workload assessment and fuzzy risk analysis, Cyber. Syst.: Int.J. vol.27, no.5, pp.449-472

[Cohn and Renz, 2008] G. Cohn and J. Renz. (2008). Qualitative Spatial Representation and Reasoning. Handbook of Knowledge Representation”, pages 551-596

[David et al, 1992] Z. C. David. A. Randell, A. G. Cohn. (1992). A spatial logic based on regions and connection” in 3rd International Conference on knowledge representation and reasonning, vol. 1, pp. 165–176

[Davidson, 1985] Davidson,D. (1985) The individuation of events, in Davidson, p 179

[Davidson, 2001] Davidson,.(2001). Essays on Actions and Events,p 150

[Dittrich and Lucas, 2014] André Dittrich, Christian Lucas. (2014). Is this Twitter Event a Disaster? Huerta, Schade, Granell (Eds): Connecting a Digital Europe through Location and Place. Proceedings of the AGILE'2014 International Conference on Geographic Information Science, Castellón, June, 3-6, 2014. ISBN: 978-90-816960-4-3

[Doane et al., 2011] David P. Doane and Lori E. (2011). Measuring Skewness: A Forgotten Statistic? Journal of Statistics Education Volume 19, Number 2, www.amstat.org/publications/jse/v19n2/doane.pdf

[Egenhofer and Al-Taha, 1992] Egenhofer, M.J., Al-Taha, K.(1992). Reasoning about Gradual Changes of Topological Relations, International Conference GIS- From Space to Territory. Springer Verlag LNCS

[Elmagarmid et al., 2007] Elmagarmid, A., Ipeirotis, P., Verykios, V. (2007). Duplicate Record Detection: A survey. IEEE Transactions on Knowledge and Data Engineering 19(1):1–16

[Endsley and Garland, 2000] Endsley, M.R. and Garland, D.J.(2000). Situation Awareness Analysis and Measurement, Lawrence Erlbaum Associates, Mahawah, New Jersey,USA.

[Estabrooks et al., 2004] Estabrooks, A., Jo, T. and Japkowicz, N. (2004). A Multiple Resampling Method for Learning from Imbalanced Data Sets. Computational Intelligence, 20: 18–36. doi: 10.1111/j.0824-7935.2004.t01-1-00228.x

[Fawcett, 2006] Fawcett, Tom .(2006). An introduction to ROC analysis, Pattern Recognition Letters, 27, 861–874

[Fellegi and Sunter, 1969] Fellegi, I. P., and A. B. Sunter (1969), "A Theory for Record Linkage," Journal of the American Statistical Association, 64, pp. 1183-1210.

[Freksa, 1992] Freksa, C. (1992a) Temporal Reasoning Based on Semi-Intervals. Artificial Intelligence 54: pp. 199-227.

[Freksa, 1992b] Freksa, C. (1992). Using Orientation Inforamtion for Qualitative Spatial Reasoning. in:Franzosa, R., Campari, I., and Formentini, U., (Eds.), Theories and Methods of SpatialTemporal Reasoning in Geographic Space. Springer-Verlag, New York, pp. 162-178.

[Formica, 2007] Formica, A.(2007). Concept similarity in formal concept analysis: An information content approach. Knowledge- Based Systems 21:80–87.

[Garson, 2009]Garson, G. D. (2009). “Logistic Regression” from Statnotes: Topics in Multivariate Analysis. Retrieved 6/5/2009 from ttp://faculty.chass.ncsu.edu/garson/pa765/statnote.htm.

[Ganter and Wille, 1999] B. Ganter, R. Wille.(1999). Formal Concept Analysis: mathematical foundations. Springer, Heidelberg 1999.

[Gangemi et al.,2002] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening Ontologies with DOLCE. In EKAW. Springer, 2002.

[Ghidini and Giunchiglia, 2001] Ghidini, C., and Giunchiglia, F.(2001). Local models semantics, or contextual reasoning = Locality + compatibility. Artificial Intelligence 127, 2, 221–259.

[Graham, 2009] Graham, J. W. (2009). Missing data analysis: Making it work in the real world. Annual Review of Psychology, 60, 549–576.

[Giunchiglia, 1993] Giunchiglia, F. Contextual reasoning. Epistemologica - Special Issue on I Linguaggi e le Macchine 16 (1993), 345–364. Also IRST-Technical Report 9211-20, IRST, Trento, Italy.

[Giunchiglia et al., 2010] F. Giunchiglia, V. Maltese, F. Farazi, and B. Dutta. Geowordnet: a resource for geo-spatial applications. In ESWC, Heraklion, Greece, 2010

[Goldstone, 2004] Goldstone, R. L. (2004) Similarity. in: R.A.Wilson and F.C.Keil, (Eds.), Mit Encyclopedia of the Cognitive Sciences. MIT Press,

[Goodman, 1972] N. Goodman. Seven strictures on similarity, pages 437–450. Bobbs Merrill, Indianapolis, 1972.

[GO, 2000] The Gene Ontology Consortium. Gene ontology: tool for the unification of biology. Nat Genet. May 2000;25(1):25-9. Online at Nature Genetics.

[Gower, 1971] Gower J. C.(1971). A General Coefficient of Similarity and Some of Its Properties. In: Biometrics, 27(4). 857 - 871.

[Hahn and Chater, 1998] Hahn, U., & Chater, N. (1998). Understanding similarity: A joint project for psychology, case-based reasoning and law. Artificial Intelligence Review, 12, 393-427.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., Friedman J. (2001). The Elements of Statistical Learning. Springer-Verlag

[Hofmann et al., 2008] Hofmann, T., Schölkopf, B., and Smola, A. Kernel methods in machine learning. The annals of statistics 36, 3 (2008), 1171–1220.

[Howell, 2012] David C. Howell, online at
https://www.uvm.edu/~dhowell/StatPages/More_Stuff/Missing_Data/Missing.html

[Hsieh and Chen, 1999] C.H. Hsieh and S.H.Chen.(1999). Similarity of generalized fuzzy numbers with graded mean integration representation, in Proc. 8th Int. Fuzzy Systems Association World Congr., vol.2, Taipei, Taiwan, Republic of China,pp.551-555

[Jaegwon, 1973] K.Jaegwon, The Journal of Philosophy Vol. 70, No. 8 (Apr. 26, 1973), pp. 217-236 Published by: Journal of Philosophy, Inc.

[Jiang and Conrath, 1997] J. J. Jiang and D. W. Conrath.(1997). "Semantic similarity based on corpus statistics and lexical taxonomy," in Int. Conf. Research on Computational Linguistics

[Knau and Renz, 1997] M. Knau, R. Rauh, J.Renz.(1997). A cognitive assessment of topological spatial relations: Results from an empirical investigation. In Proceedings of the 3rd International Conference on Spatial Information Theory (COSIT'97),volume 1329 of Lecture Notes in Computer Science, pages 193{206, doi: 10.1007/3-540-63623-4_51

[Lambert, 1999] Lambert, D. A.(1999).Assessing Situations, Proceedings of Information, Decision and Control, pp. 503 – 508. IEEE.

[Landis et al, 1977] J. R. Landis and G. G. Koch.(1977).The measurement of observer agreement for categorical data. Biometrics, 33(1):159–174.

[Larkey and Markman, 2005] Larkey, L. B. and Markman, A. B. (2005), Processes of Similarity Judgment. Cognitive Science, 29: 1061–1076. doi: 10.1207/s15516709cog0000_30

[Leacock and Chodorow, 1998] C. Leacock and M. Chodorow. (1998) Combining Local Context and Wordnet Similarity for Word Sense Identification, pp. 265–283, 1998.

[Lee et al., 2008] Lee WN, Shah N, Sundlass K, Musen M. Comparison of ontology-based semantic-similarity measures. Proceedings of the American Medical Informatics Association Annual Symposium Proceedings; 2008; American Medical Informatics Association; pp. 384–388. [PMC free article] [PubMed]

[Lesk, 1986] M. Lesk, (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine code from an ice cream cone," in the 5th Annu. Int. Conf. on Systems Documentation, 1986, pp. 24–26.

[Li and Fonseca,2006] Li, B. and F. T. Fonseca. (2006). "TDD - A Comprehensive Model for Qualitative Spatial Similarity Assessment." Spatial Cognition and Computation 6(1): 31-62. pre print version

[Lin, 1997] D. Lin, "Using syntactic dependency as local context to resolve word sense ambiguity," in Annu. Meeting of the Association for Computational Linguistics (ACL), 1997, pp. 64–71.

[Lin, 1998] Lin, D.(1998). An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufmann 296–304

[Little and Rubin, 1987] Little, R.J.A. & Rubin, D.B. (1987) Statistical analysis with missing data. New York, Wiley

[Lord et al., 2003] Lord P, Stevens R, Brass A, Goble C. (2003). Semantic similarity measures as tools for exploring the gene ontology. Proc. of the 8th Pacific Symposium on Biocomputing. pp. 601–612.

[Makkonen et al., 2004] J. Makkonen, H. Ahonen-Myka, and M. Salmenkivi.(2004).Simple semantics in topic detection and tracking. Information Retrieval, 7(3{4):347{368, 2004.

[Markman and Gentner, 1993] A.B. Markman and D. Gentner. (1993). Structural alignment during similarity compar- isons. Cognitive Psychology, 25(3):431–467,

[Menard, 2001] Menard, S. W. (2001). Applied logistic regression analysis (quantitative applications in the social sciences) (2nd ed.). Thousand Oaks, CA: Sage Publications

[Medin et al., 1993] D. Medin, R. Goldstone, and D. Gentner. (1993). Respects for similarity. Psychological Review, 100(2):254–278

[Meyer et al., 2011] Brett Meyer, Kevin Bryan, Yamara Santos, and Beomjin Kim (2011). TwitterReporter: Breaking News Detection and Visualization through the Geo-

Tagged Twitter Network. Presented at Proceedings of the 26th International Conference on Computers and Their Applications (CATA-2011), New Orleans, LA.

[Mitchell, 2007] Mitchell, H.B. (2007). Multi-sensor data fusion: an introduction. New York: Springer-Verlag,.

[Mourelatos, 1978] Alexander P. D. Mourelatos.(1978). Events, Processes, and States. Linguistics and Philosophy 2 (3):415 - 434

[Ng, 2005] Ng A., "Feature Selection, L1 vs. L2 regularization, and rotational invariance," In the proceeding of International Conference on Machine Learning. Alberta, Canada, 2005.

Ng, A. Y. (1998). On feature selection: Learning with exponentially many irrelevant features as training examples. Proceedings of the Fifteenth International Conference on Machine Learning (pp. 404-412). Morgan Kaufmann

[Oyama and Manning, 2004] Oyama, Satoshi, Manning, D C. (2004).Using feature conjunctions across examples for learning pairwise classifiers. 15th European Conference on Machine Learning (ECML2004) 2004.

[Papadias, D. and Dellis, 1997] Papadias, D. and Dellis, V. (1997) Relation-Based Similarity. in: Fifth ACM Workshop on Advances in Geographic Information Systems, Las Vegas, NV.

[Papadias et al., 1999] Papadias, D., Karacapilidis, N., and Arkoumanis, D.(1999). Processing fuzzy spatial queries: a configuration similarity approach. International Journal of Geographical Information Science 13 (2): 93-118

[Payne, 1997] Payne, Thomas E. (1997). Describing morphosyntax: A guide for field linguists. Cambridge; New York: Cambridge University Press. 413 pages

[Peng et al., 2002] Peng, C. Y., & So, T. S. H. (2002). Logistic regression analysis and reporting: A primer. Understanding Statistics, 1(1), 31-70.

[Peng et al., 2002b] Chao-Ying Joanne Peng, Kuk Lida Lee, Gary M. Ingersoll.(2002).An Introduction to Logistic Regression Analysis and Reporting The Journal of Educational Research Vol. 96, Iss. 1, 2002

[Pesquita et al., 2009] Pesquita C, Faria D, Falcão AO, Lord P, Couto FM.(2009). Semantic Similarity in Biomedical Ontologies. PLoS Comput Biol 2009, 5(7):e1000443

[Pustejovsk, 2011] J.Pustejovsk. (2011). ISO-Space: The Annotation of Spatial Information in Language", Proceedings of the Sixth Joint ISO - ACL SIGSEMWorkshop on Interoperable Semantic Annotation isa-6

[Quine, 1985] Quine, W. V. (1985). Events and reification. In R.Casati & A.C.Varzi (Eds). Events (pp.107-116).

[Rada et al.,1989] Rada, Roy; Mili, Hafedh; Bicknell, Ellen, Blettner, Maria.(1989). Development and Application of a Metric on Semantic Nets, IEEE Transactions on Systems, Man, and Cybernetics, Volume 19

[Raghunathan, 2004] Raghunathan TE. What do we do with missing data? some options for analysis of incomplete data. Annual Review of Public Health. 2004;25:99–117

[Reuter et al., 2011] T. Reuter, P. Cimiano, L. Drumond, K. Buza, and L. Schmidt-Thieme.(2011). Scalable event-based clustering of social media via record linkage techniques. In Fifth International AAAI Conference on Weblogs and Social Media (ICWSM), 2011.

[Resnik, 1995] Resnik, Philip. (1995). Using information content to evaluate semantic similarity. In Proceedings of the 14th International Joint Conference on Artificial Intelligence, pages 448–453, Montreal, Canada, August.

[Richter, 2003] Richter, M. (2003). Learning Similarities for Informally Defined Objects. In K¨uhn,R., Menzel, R., Menzel, W., Ratsch, U., Richter, M., and Stamatescu, I.-O., editors, Adaptivity and Learning. Springer.

[Rodríguez and Egenhofer, 2003] Rodríguez MA, Egenhofer MJ. (2003). Determining semantic similarity among entity classes from different ontologies. IEEE Transactions on Knowledge and Data Engineering. 15(2):442–456.

[Rodríguez et al.,1999] Rodríguez MA, Egenhofer M, Rugg R.(1999). Interoperating Geographic Information Systems. Assessing semantic similarities among geospatial feature class definitions; pp. 189–202.

[Rodríguez and Egenhofer, 2004] Rodríguez MA, Egenhofer MJ.(2004). Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure. International Journal of Geographical Information Science.18(3):229–256.

[Renz and Nebel, 2007] J. Renz and B. Nebel.(2007). Qualitative spatial reasoning using constraint calculi. In Handbook of Spatial Logics, pages 161{215. Springer,. ISBN 978-1-4020-5586-7. doi:10.1007/978-1-4020-5587-4_4

[Renz, 1998] J. Renz. (1998). A canonical model of the Region Connection Calculus, in: Proc. 6th International Conference on Principles of Knowledge Representation and Reasoning (KR-98), Trento, Italy, 1998. DOI: 10.3166/jancl.12.469-494

[Roche-Lima et al.,2014] Roche-Lima, A., Domaratzki, M., Fristensky, B. (2014) Metabolic network prediction through pairwise rational kernels. Submitted BMC Bioinformatics

[Sarawagi and Bhamidipaty, 2002] Sarawagi S, and Bhamidipaty A (2002). Interactive deduplication using active learning, Proceedings of the eighth ACM SIGKDD International 1 Conference. Knowledge Discovery and Data Mining (KDD ’02), 269–278.

[Scherp et al,. 2009] A. Scherp, T. Franz, C. Saathoff, and S. Staab.(2009). F—A Model of Events based on the Foundational Ontology DOLCE+ Ultra Light. In 5th International Conference on Knowledge Capture (K-CAP’09), Redondo Beach, California, USA,

[Schwering and Raubal, 2005] Schwering, A., Raubal, M. (2005). Spatial relations for semantic similarity measurement. In: Akoka, J., Liddle, S.W., Song, I.-Y., Bertolotto, M., Comyn Wattiau, I., van den Heuvel, W.-J., Kolp, M., Trujillo, J., Kop, C., Mayr, H.C. (eds.) Perspectives in Conceptual Modeling. LNCS, vol. 3770, pp. 259–269. Springer, Heidelberg

[Shariff et al., 1998] Shariff, A.R., M.J. Egenhofer, and D.M. Mark.(1998). Natural-Language Spatial Relations Between Linear and Areal Objects: The Topology and Metric of English Language Terms. International Journal of Geographical Information Science, 12(3): p. 215-246.

[Sinha and Mihalcea, 2007] R. Sinha and R. Mihalcea. (2007). Unsupervised graphbased word sense disambiguation using measures of word semantic similarity. In Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007), Irvine, CA, USA

[Tibshirani, 1996] Tibshirani R.(1996). Regression Shrinkage and Selection via the LASSO," Journal of the Royal Statistics Society, Series b (Methodological), Volume 58, pp. 267-288.

[Tversky, 1977] Tversky A. (1977). Features of similarity. Psychological Review. ; 84(4):327–352.

[Vapnik, 1999] Vapnik, V . (1999). An Overview of Statistical Learning Theory. IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 10, NO. 5, SEPTEMBER

[Vert et al., 2004] J. P. Vert, K. Tsuda, B. Scholkopf,.(2004). A primer on kernel methods Kernel Methods in Computational Biology, pp. 35-70

[Wang et al., 2004] Wang H, Azuaje F, Bodenreider O, Dopazo J. Gene expression correlation and gene ontology-based similarity: an assessment of quantitative relationships. Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB '04); October 2004; IEEE; pp. 25–31.

[Wilke and Bergmann, 1996] W. Wilke and R. Bergmann. (1996). Considering decision cost during learning of feature weights. In Proceedings of the 3rd European Workshop on Case-Based Reasoning. Springer.

[Wille, 2005] R. Wille.(2005). Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies. Lecture Notes in Computer Science, 3626:1–33.

[Wordnet, 2010] Princeton University "About WordNet." WordNet. Princeton University. 2010. <http://wordnet.princeton.edu>

[Wu and Palmer, 1994] Wu, Z., Palmer, M.(1994). Verb semantics and lexical selection. In: Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics. 133–138

[Xie et al., 2013] K. Xie, C. Xia, N. Grinberg, R. Schwartz, and M. Naaman. (2013). Robust detection of hyper-local events from geotagged social media data. In Proceedings of the 13th Workshop on Multimedia Data Mining in KDD.

[Yang and Pierce, 1998] Y. Yang, T. Pierce, and J. G. Carbonell.(1998). A study of retrospective and on-line event detection. In SIGIR, pages 28–36, 1998

[Zacks and Tversky,2001] Zacks, J. M., & Tversky, B. (2001). Event structure in perception and conception. Psychological Bulletin, 3-21

[Zadeh, 1965] Zadeh, L.A.(1965). Fuzzy sets. Information and Control 8 338–353

[Zou and Hastie, 2005] Zou H. and Hastie T.(2005). "Regularization and Variable Selection via the Elastic Net," Journal of the Royal Statistical Society: Series B (Statistical Methodology), pp. 301–320.

Appendix – MATLLAB functions

```
%% Calculate similarity for morning and early morning

% loop through different algorithms
methods = {'chen','hamming','scgm','hsieh','overlap','sigma'};
sim_out= {};
for i=1:length(methods)
  disp(methods(i));
  sim = FindSimilarity( morning, earlyMorning,methods(i));
  sim_out{i} = sim;
  fprintf('sim %f\n',sim);

end




function [ sim ] = FindSimilarity( A, B, method )

wA=1; % confidence
wB=1;

a1=A(:,1);
a2=A(:,2);
a3=A(:,3);
a4=A(:,4);

b1=B(:,1);
b2=B(:,2);
b3=B(:,3);
b4=B(:,4);

% switch method
 % case 'chen' strcmp('str1', 'str2')
 if strcmp(method,'chen')
  d = abs(A -B) ;
  s = sum(d)/4;
  sim = 1- s;

 elseif strcmp(method,'hamming') %% Euclidean distance between two
fuzzy sets
   sim = 0 ;
   s=sqrt(sum((A-B).^2));
   sim= s/2;
 elseif strcmp(method,'scgm')
     %% check if 0 ? a1 ? a2 ? a3 ? a4 ? 1
```

```
    %% and 0 ? b1 b2 b3 b4 ? 1
    inqA = logical (a1 < a2 <a3 <a4);
    inqB = logical (b1 < b2 <b3 <b4);
   % if (inqA==1 & inqB==1)
      if (a1 ~= a4 )
        temp0 =a3-a2;
        temp1 = (a4-a1);
        temp = (a3-a2)/(a4-a1);
        temp2 = temp +2;
        yA = (wA*temp2)/6;
      % yA= wA *(((a3-a2)/(a4-a1))+2);
      else

       yA=wA/2;
      end
      xA = (yA*(a3+a2)+(a4+a1)*(wA-yA))/2*wA;

    %% compute for B
    if (b1 ~= b2 )
       yB= wB *((b3-b2)/(b4-b1)+2)/6;
      %        fprintf('yB: %f\n',yB);
      else
       yB=wB/2;
      end
      xB = (yB*(b3+b2)+(b4+b1)*(wB-yB))/2*wB;
  %  end ;
    %% compute B(S_A,S_B)
    sA = a4-a1;
    sB = b4-b1;
      if (sA+sB > 0 )
        Bab = 1;
      elseif (sA+sB == 0 )
      Bab = 0 ;
      end
  %% compute 1-sum
  d = abs(A -B) ;
  s = sum(d)/4;
  sim_AB = s;
  %% find min(yA,yB)
  minAB = min(yA,yB);
  maxAB = max(yA,yB);

  z = abs(xA -xB);
  zAB = (1 - z)^ Bab; %* minAB/maxAB);
  sim = (1-s)*zAB;
  sim = sim* minAB/maxAB;

elseif strcmp(method,'hsieh')
    PA = (a1+2*a2+2*a3+a4)/6;
```

```
     PB = (b1+2*b2+2*b3+b4)/6;
     d =abs(PA -PB) ;
     sim = 1 /( 1+ d );
 elseif strcmp(method,'overlap')
     intersect_AB = min(A,B);
     union_AB = max(A,B);
     sim = sum(intersect_AB)/sum(union_AB);
 elseif strcmp(method,'sigma') % sigma count
   sim =sum(min(A,B))/max(sum(A),sum(B));
 end

end
```