PlanetData

Network of Excellence

FP7 – 257641

D1.5 Trend and anomaly detection in non-structured data

Coordinator: Alexandra Moraru

With contributions from: Janez Brank, Marko Grobelnik

1st Quality reviewer: Oscar Corcho
2nd Quality reviewer: Pablo Mendez

Deliverable nature:	R
Dissemination level: (Confidentiality)	PU
Contractual delivery date:	M24
Actual delivery date:	M24
Version:	1.0
Total number of pages:	29
Keywords:	Non-structured data, social media, data modalities, complex event processing, trend and anomaly detection, online learning, adaptive data summarization, network sampling, network evolution.

Abstract

Non-structured or unstructured data is data that doesn’t conform to an explicit and well-defined formal data model. This deliverable focuses on textual and network data. We discuss several statistical properties by which these types of data differ from more structured data. Trend and anomaly detection is the process of discovering patterns in the data that do not conform to normal or expected behaviour; it has many applications and draws upon techniques from several related disciplines. We present the state of the art and directions for future work in several areas relevant to trend and anomaly detection in textual and network data: text processing of informal documents, online learning, adaptive data summarization, event processing, social media management, network sampling, and network evolution.

Executive summary

Trend and *anomaly detection* is the process of discovering patterns in the data that do not conform to normal or expected behaviour of the data stream; the main difference between the two is that a trend is not merely an aberration from normal behaviour but an actual change in what normal behaviour is. Trend and anomaly detection draw upon techniques from several related disciplines, such as machine learning, data mining, text mining, statistics and information theory, natural language processing, etc. The approaches to trend and anomaly detection involve classification, clustering, nearest-neighbour approaches, statistical and information-theoretic approaches, and spectral methods. Applications of trend and anomaly detection can be found in numerous areas, such as sensor networks, cyber-intrusion detection, fraud detection, medicine, fault detection (in networks, manufacturing, etc.), image processing, sensor networks, topic detection (from news feeds) etc.

Non-structured or *unstructured* data is data that doesn't conform to an explicit and well-defined formal data model with relations, attributes etc. This includes textual data, multimedia data (images, video), and networks. Nowadays an increasing amount of relevant information is available from non-structured data sources, leading to an increasing need for data analysis tools and techniques that can deal with non-structured data. Text processing of *informal documents* is an area of text mining that focuses on short informal user-generated documents such as tweets, product reviews, blog and forum posts and comments, or profiles on social networking sites. It addresses problems such as sentiment detection, summarization, classification, tagging, retrieval and recommendation, discourse analysis, and domain adaptation.

Online learning is a sequential prediction protocol in which the learner always has access to a source of supervised information. The predictive model is thus trained incrementally, and may easily adapt to changes in the data source. When applied to evolving data streams, online algorithms typically assess the confidence of their own predictions in order to trade off the amount of supervised information accessed with the average accuracy of their predictions.

Adaptive data summarization is the task of summarizing an evolving stream of data. *Coresets* are a concept initially from computational geometry, where the coreset of a set of points is a smaller but representative subset of the entire set, and is used in further processing instead of the original set to obtain results that are adequate approximations of the results that would have been obtained on the full set. Coresets lead naturally to streaming algorithms and have recently been generalized to other problem domains such as document summarization.

Complex event processing (CEP) is the analysis of events from different event sources in near real-time in order to generate immediate insight and enable immediate response to changing business conditions. It has applications in areas such as transportation, logistics, telecommunications and customer risk management. Events are classified into simple or atomic events and complex events. A simple event is atomic and does not contain further events. A complex event on the other hand is composed of other simple or complex events.

Social media management is an area of growing importance for organizations that are increasingly expected to use social media to communicate with their key stakeholders. In order to be able to use social media, organizations need to build up dedicated skills and resources, which are currently lacking in many organizations.

Network sampling is essential for efficient processing and analysis of large networks. We are often forced to pursue approximate results based on processing only one or more subsets of the graph, either because the full graph is untractably large, or because only local views of the graph are available. The aim of network sampling is to retain the structural properties of the full graph while also preserving a similar distribution of node and edge attributes to that of the full graph.

The study of *network evolution* is important as a lot of real-world networks change and evolve over time, as nodes and edges are being added or removed, or their attributes change. Several patterns that typically occur in such evolving networks have been identified, such as *densification* (as the network grows, the average degree of its vertices increases) and *diameter shrinkage* (the diameter, i.e. maximum length of a shortest path between two vertices, often tends to decrease slowly, due to the increasingly good connectivity of the graph). The study of network evolution can help us detect trends and anomalies in the network.

Document Information

IST Project Number	FP7 - 257641	**Acronym**	PlanetData
Full Title	PlanetData		
Project URL	http://www.planet-data.eu/		
Document URL	http://wiki.planet-data.eu/web/D1.5		
EU Project Officer	Leonhard Maqua		

Deliverable	**Number**	D1.5	**Title**	Trend and anomaly detection in non-structured data
Work Package	**Number**	WP1	**Title**	Data Streams and Dynamicity

Date of Delivery	**Contractual**	M24	**Actual**	M24
Status	version 1.0		final ☒	
Nature	prototype □ report ☒ dissemination □			
Dissemination level	public ☒ consortium □			

Authors (Partner)	Janez Brank(IJS), Marko Grobelnik (IJS), Alexandra Moraru (IJS)			
Responsible Author	**Name**	Alexandra Moraru	**E-mail**	alexandra.moraru@ijs.si
	Partner	IJS	**Phone**	+ 386 1 477 3144

Abstract (for dissemination)	Non-structured or unstructured data is data that doesn't conform to an explicit and well-defined formal data model. This deliverable focuses on textual and network data. We discuss several statistical properties by which these types of data differ from more structured data. Trend and anomaly detection is the process of discovering patterns in the data that do not conform to normal or expected behaviour; it has many applications and draws upon techniques from several related disciplines. We present the state of the art and directions for future work in several areas relevant to trend and anomaly detection in textual and network data: text processing of informal documents, online learning, adaptive data summarization, event processing, social media management, network sampling, and network evolution.
Keywords	Non-structured data, social media, data modalities, complex event processing, trend and anomaly detection, online learning, adaptive data summarization, network sampling, network evolution.

Version Log			
Issue Date	**Rev. No.**	**Author**	**Change**
2012-08-23	0.1	Alexandra Moraru	Initial draft of ToC
2012-08-28	0.2	Marko Grobelnik, Alexandra Moraru	Revised ToC
2012-09-03	0.3	Janez Brank, Marko Grobelnik	Initial draft for internal review
2012-09-13	0.4	Alexandra Moraru, Janez Brank	Revised according to review 1
2012-09-20	0.5	Alexandra Moraru	Revised according to review 2
2012-09-25	1.0	Alexandra Moraru	Prepare final version

Table of Contents

1 Introduction

Non-structured or *unstructured* data [1] is data that doesn’t conform to an explicit and well-defined formal data model with relations, attributes etc. This includes textual data, multimedia data (images, video), and networks. Nowadays an increasing amount of relevant information is available from non-structured data sources, leading to an increasing need for data analysis tools and techniques that can deal with non-structured data.

In practice, the term non-structured data is somewhat misleading, as data is rarely completely devoid of structure. However, it might happen that the structure is only implied in the data, i.e. it can be discovered by analysis but is not encoded explicitly at the information source; or that structure is available but is insufficient or irrelevant to the user’s needs, thereby making the data effectively unstructured or *semi-structured* for their purpose.

Trend and anomaly detection is the process of discovering patterns in the data that do not conform to normal or expected behaviour [2][3]. Another closely related problem is that of noise detection. The difference between these three is that in the case of noise detection, the anomalous data is not considered interesting but is only to be removed, whereas anomalies and trends are of interest to the user; finally, trends are those anomalies that signify an actual change in what is perceived as normal behaviour of the data and will come to be incorporated into an updated model of normal behaviour.

Trend and anomaly detection draw upon techniques from several related disciplines, such as machine learning, data mining, text mining, statistics and information theory, natural language processing, etc. Applications of trend and anomaly detection can be found in numerous areas, such as sensor networks, cyber-intrusion detection, fraud detection, medicine, fault detection (in networks, manufacturing, etc.), image processing, sensor networks, topic detection (from news feeds) etc.

2 Data Modalities

Textual data usually involves a corpus of natural-language text documents, where individual documents are generally not very long but the number of documents can be very large. Each document may include some “traditional” attribute-based data, but its main part is unstructured natural-language text. Two notable ways in which textual data differs from traditional structured data are *sparsity* and *power-law distributions*.

This deliverable deals with unstructured data, particularly textual and network data. Other data modalities, like sensor data (especially environmental sensor data) are covered in D1.3[4].These types of data differ in several respects from structured data that has traditionally been the subject of data mining work; these differences render some of the established method less suitable for handling text and network data.

In a traditional structured dataset, there is usually a low or moderate number of attributes and each instance contains values of all these attributes; by contrast, textual data consists of words, where the number of possible different words is very large (and new ones appear continuously as the dataset grows) but each individual document contains only a relatively small subset of these words (i.e. the data is sparse).

Furthermore the frequency of words varies greatly, there being a small number of high-frequency words (which occur in many or even all documents) and a large number of increasingly rarer words, many of which however are still important and must not be ignored when processing the data. The distribution of word frequencies generally has the form of a power-law distribution. The same observations hold if we take, as the basic building blocks of the texts, not words but sequences of adjacent words (known as *n-grams*), phrases, or even sequences of adjacent characters.

Another important aspect that makes textual data more challenging is that texts exist on a wide range of registers and styles, from formal and regular (e.g. news articles) to highly informal and irregular (e.g. tweets, comments on blogs and social networks, etc.), sometimes even within the same dataset. This poses an additional challenge to natural language processing techniques, which often have difficulty dealing with informal and irregular text.

Network data generally consists of a graph with various attendant and supporting data. The graph consists of a set of nodes or vertices, with a set of links or edges connecting them. Each vertex represents some entity in the domain of interest, and the edges indicate relationships between them. Often some additional data may be attached to a vertex or an edge, either as traditional structured (attribute-based) data, or often also in the form of unstructured text. For example, vertices may represent web pages, with edges representing hyperlinks between those pages. In this case a vertex might include attributes such as the URL of the web page and its full textual contents; an edge might include attributes such as the context within which the link occurred. Sometimes a network is the outcome of an analysis of some source data where it wasn’t explicitly present to begin with; for example, one might form a graph by extracting named entities from a corpus of documents and introduce edges on the basis of co-occurrences, to link entities that are often mentioned close together in the source documents.

Many networks arising from real-world phenomena (such as communications, traffic, web, social networks, product reviews, citation networks) have particular statistical characteristics which need to be taken into account by methods that aim to analyze network data. For example, vertex degrees (i.e. the number of edges incident on a vertex) often follow a power-law distribution: there would be a small number of vertices with a very high number of links, and a large number of vertices with an increasingly small number of links. This phenomenon often holds across a wide range of scales and is sometimes referred to as *self-similarity*.

Network and textual data naturally occur together in many real-world problem domains, for example in social networking and blogging, where actions such as commenting or replying generate at the same time a short textual document and a link (relationship) between the participants.

3 Defining trend and anomaly detection

Chandola et al. [2] define anomalies as patterns in data that do not conform to a well defined notion of normal behaviour. Anomaly detection differs from simple noise detection/removal chiefly by the fact that the the user is actually interested in knowing about the anomaly and understanding it, whereas in the case of noise the goal is simply to remove it as being of no interest to the user. By contrast, trend detection or novelty detection [3][6][7] aims to discover patterns that, while they might have started as anomalous departures from normal behaviour, will become incorporated into a revised understanding of what normal behaviour is.

Anomaly and trend detection are used in numerous application domains, such as sensor networks, cyber-intrusion detection, fraud detection, medicine, fault detection (in networks, manufacturing, etc.), image processing, sensor networks, topic detection (from news feeds) etc. They draw upon techniques from several related fields, such as machine learning, data mining, statistics, and information theory:

- Classification based: these methods assume that a training set is available in which the instances have already been labelled as normal or anomalous. A predictive classifier is then trained which can distinguish between normal and anomalous classes. Various machine learning techniques can be used for this, such as neural networks, Bayesian classifiers, support vector machines, classification rules, etc.
- Clustering based: based on the assumption that if we cluster the data, normal data instances will end up together in large and dense clusters, while anomalies will belong to small clusters and/or sparse clusters (where the difference from the instance to its cluster centroid is large).
- Nearest neighbour approaches: based on the assumption that normal data instances occur in dense neighbourhoods, while anomalies occur far from their closest neighbours. Thus, either the distance to the *k*-th nearest neighbour, or the overall density of an instance's neighbourhood, can be used as indicators of anomalousness.
- Statistical approaches: based on the assumption that the instances are sampled from a probability distribution, whose parameters we can estimate from the available training data. These approaches assume that normal instances will lie in areas of high probability density, so if an instance lies in an area of low probability density, it is likely to be anomalous.
- Information-theoretic approaches: based on the assumption that anomalies in the data induce irregularities in the information content of the data set (i.e. the number of bits needed to represent the data). For example, if removing a few instances leads to a substantial decrease in the information content of the set (e.g. as measured by its Kolmogorov complexity), those instances can be considered as having been anomalous.
- Spectral methods: based on the assumption that we can project the data into a suitable lower-dimensional subspace in which normal instances and anomalous instances appear substantially different. For example, we might project our data into the subspace defined by the first few principal components, i.e. those directions that explain most of the variance in the data; then instances with large absolute values of their projections can be considered anomalous.

Sometimes a data instance might be recognized as anomalous by itself (known as a *point anomaly*), but often an anomaly can only be recognized when it manifests in a number of data instances (*behavioural anomaly*) or when it appears in the context of normally-behaving data (*conditional* or *contextual anomaly*; for example, a high temperature reading might be normal in summer but anomalous during winter).

4 Progress beyond the state of the art

4.1 Text processing of informal documents

4.1.1 Current state of the art

Informal user-generated documents (often very short ones), such as tweets, product reviews, blog and forum posts and comments, as well as user profiles on social networking sites, are an important part of Web 2.0 and have been the subject of very active research in recent years. A number of problems are now being addressed specifically in the context of informal texts:

- **Sentiment detection.** [31] mined online product reviews to detect the users' sentiments about various aspects of the product; [11] studied tweets to detect users' sentiments about certain topics, using a predefined keyword list; [14] took a more fine-grained approach in the area of blog posts, trying to identify passages within a post that are relevant to a given query and also express a sentiment about it. On a semi-related note, [21] present an approach involving a number of NL technologies to discover blog posts that express a user's experience (knowledge referring to an "activity or event that [the user] has actually undergone").
- **Summarization.** [30] presents an interesting approach to summarize the users' opinion of an individual song, given their reviews of entire albums. [27] discusses summarization of all tweets relevant to a given query topic. [19] uses language modelling to identify the main headlines of the day, given that day's blog posts. [20] summarizes YouTube comments with a tag cloud, separating positive from negative sentiments. [34] generates/selects a few tags to describe a user's interests given his/her tweet stream.
- **Classification.** [17] classified MySpace profiles into genuine users and spam profiles. [25] used latent Dirichlet allocation to cluster a user's tweet stream into several topics. [26] classified tweets across users into predefined classes such as news, events and opinions. [18] studied the internal structure of e-mail messages and tried to identify messages that contain a request for action.
- **Tagging.** [15] trained taggers for semantic classes such as "human", "animal", "disease", with application to posts on a veterinary forum. [29] identified sarcastic sentences in product reviews based on a nearest-neighbour classifier and the presence of certain word patterns. [28] presents a very lightly supervised approach for named entity extraction from advertisement slogans.
- **Retrieval and recommendation.** [12] trained a language model for each hash-tag on twitter and used these models to retrieve thematically relevant tweets that had not been tagged by the users. [22] improved the choice of snippets presented to the user by a blog post search engine by using the comments under blog posts. [32] showed how to improve a news recommender system by taking into account the text of users' comments under each news article.
- **Discourse analysis.** [24] modelled the flow of conversation on tweeter using a Markov chain based approach. [33] used latent semantic analysis to study initiation-response pairs (e.g. question-answer, assessment-agreement, blame-denial) in a corpus of political newsgroup posts. [23] discusses new topic detection from tweets, with a new locally sensitive hashing approach for greater scalability.

Besides works such as those mentioned above, which try to address a specific problem on a specific informal text corpus, there has also been some research that tried to take a broader look at the characteristics of informal text corpora and the ways in which they differ from traditional corpora. [16] studied statistical properties of informal texts; [9] discusses the influence of spelling errors (typical of informal texts) on machine translation; [13] investigated parser performance on informal texts and demonstrated that it's better if the parser itself is trained on corpus exhibiting similar characteristics.

Finally, adapting natural language models to informal texts can be as a special form of domain adaptation, where the lexical and syntactic patterns suffer a variation with respect to an original source. There has been recent work in inducing representations that are invariant with respect to the domains being used [10][8]. Taggers and parsers are learned on the induced representation using supervision from only one of the domains, while the induced representation is learned using some unsupervised signal (such as topic or lexical similarity).

4.1.2 Progressing beyond the state of the art

Some of the interesting directions for further research in this area include:

- extracting information from informal documents (such as blogs and social media networks);
- discovering new terms (and their meaning) used in informal sources using statistical methods;
- to find regularities between formal and informal sources which can be captured by language models for spelling correction, and probabilistic dictionaries mapping formal to informal words and phrases;
- domain adaptation methods based in semi-supervised learning. Following [8], one approach is to learn a representation of lexical items and syntactic patterns which is shared by the formal and informal texts, then obtain document similarities by distance functions looking at lexical and shallow annotations. These similarities would be used as weak signal to supervise the induction of a shared representation, robust to the noise of informal genres.

4.2 Online learning

4.2.1 Current state of the art

Online learning is a sequential prediction protocol in which the learner always has access to a source of supervised information. The predictive model is thus trained incrementally, and may easily adapt to changes in the data source. When applied to evolving data streams, online algorithms typically trade off the amount of supervised information accessed with the average accuracy of their predictions. This trade-off is achieved adaptively through a notion of predictive confidence. Namely, the algorithm has a way of computing a confidence level on its own predictions; whenever this level drops below a certain threshold, the algorithm asks for more supervised information. The incremental nature of this form of learning implies that online algorithms learn through local changes, which involve the solution of a local optimization problem. Hence, online algorithms are significantly more efficient than statistical learning algorithms, which are based instead on the solution of global optimization problems. The ability of incrementally reacting to changes in the data source brings another advantage besides that of computational efficiency. Namely, online algorithms enjoy good theoretical performance guarantees under much weaker conditions than statistical learning algorithms. For example, in a sequential classification problem the number of mistakes can be bounded irrespective of the nature of the source generating the data.

In this respect, online learning lends itself very well to applications involving analysis of massive data streams: data are operated on locally and efficiently, nonstationary data sources are naturally supported, supervised information is accessed in a controlled way, and memory requirements can be taken into account by intervening on the local optimization model. However, despite these strong points, online learning still has several limitations that prevent its application to a large variety of problems.

First of all, essentially all online learning techniques are based on combining attributes linearly. This is adequate in many applications such as image or document analysis, but it is at odds with domains where attributes are of different nature (e.g., numerical vs. categorical). Indeed, heterogeneous attributes cannot be directly compared one against each other as in a linear combination, and one should resort to different methods such as decision trees which treat each attribute independently. This way of treating attributes independently also guarantees the interpretability of the resulting classification model, which is not the case when using linear models.

Another problem arises when the inference task is defined in terms of correlations between data items in the stream, or events. This is the case in complex event detection tasks, when the decision must be made by taking into account a possibly large number of interdependent events. This scenario can be made more difficult when there are simultaneously multiple streams (e.g., social networks generate streams of tweets and streams of link updates) and multiple event detection tasks defined across these streams (e.g., detecting new trends and communities).

A third problem is more technological and stems from the fact that data intensive tasks are generally supported by a multi-core architecture. As a consequence, the algorithms must be amenable of an efficient distributed implementation without jeopardizing their predictive performance.

4.2.2 Progressing beyond the state of the art

The state of the art in online learning and massive stream analysis can be extended in different directions. In particular, the three main problems identified in the previous section should be addressed.

- The problem of dealing with heterogeneous attributes can be attacked by representing the training of a decision tree as a form of gradient descent in a suitable linear space. Once this is done, we would be able to apply the large body of available online learning technology to the analysis of decision tree learning, and possibly relate to popular decision tree learning heuristics in stream analysis, such as the VFDT algorithm [35].

- The traditional algorithmic setting of online algorithms can then be extended to cover more general inference tasks. More specifically, one could consider multiple correlated classification tasks with hierarchically organized classes. Partial results in this directions are contained in [36][37]. More importantly, online algorithms could be extended to deal with structured prediction tasks that involve detection and classification of complex events, possibly involving multiple streams with temporal and geolocational information. The kind of complex events we plan to deal with are essentially dynamic probabilistic models abstracting certain nonstationary properties of the stream (e.g. emergence and disappearance of a certain trend).

- Finally, algorithms and interaction modalities could be designed in order to accomodate distributed implementations on large multi-core architectures. This would necessarily require a modification of the analysis, as in a distributed environment training signals will diffuse through the communication network gradually, and learning agents at each node will access their supervised information with a certain latency.

4.3 Adaptive data summarization

4.3.1 Current state of the art

Our approach towards summarizing massive data sets is based on the notion of *coresets*. This notion was originally developed in computational geometry. Approximation algorithms in this area often make use of random sampling, feature extraction, and ε-samples [44]. Coresets can be viewed as a general concept that includes all of the above, and more. A comprehensive survey on this topic appears in [40]. Coresets have been the subject of a number of recent papers and several surveys [38][39]. They have been used to great effect for a variety of geometric and graph problems, including *k*-median [41], *k*-mean [42], *k*-center [45], *k*-line median [43], subspace approximation [46], etc. Coresets also imply streaming algorithms for many of these problems [38][41][42][47]. A framework that generalizes and improves several of these results has recently appeared in [40].

The classical notion of submodular set functions has proved to be a very useful abstraction for a number of problems related to optimized information gathering (such as sensor placement, active learning and autonomous exploration) [66] and document summarization [49], leading to principled algorithms with state-of-the art empirical performance. A major challenge in the past was how to incorporate feedback into the selection process. Recently this has been addressed by *adaptive submodularity*, a generalization of submodular set functions to adaptive policies [48].

4.3.2 Progressing beyond the state of the art

- A very interesting direction of progressing beyond the state of the art is to use the concept described above as the basis of an approach towards optimizing adaptive interaction. The main extension here would be to lift the concept of coresets from geometric problems to tasks involving statistical and probabilistic inference. Extensions include generalization of smart sampling in order to build coresets for distributions other than Gaussians as well as being able to handle correlated data streams rather than ones which are sampled in an independent and identically distributed fashion.

- Another direction is to extend the ideas in adaptive submodularity to be able to deal with a wide class of feedback to enable us to not only adapt the cost function but also combine it with the subsampling ideas to obtain large gains in efficiency.

4.4 Event processing

4.4.1 Current state of the art

Event Representation

Complex Event Processing (CEP) is the analysis of events from different event sources in near real-time in order to generate immediate insight and enable immediate response to changing business conditions [50]. For example Transportation and Logistics, Financial Front Office, Telecommunication or Customer Risk Management are successful application areas of CEP. Events are classified into simple or atomic events and complex events. According to [51] an event indicates a significant change in the state of the universe. Sensor data, network signals, credit card transaction or application processing information are examples for simple events. A simple event is atomic and does not contain further events. A complex event on the other hand is composed of other simple or complex events. For example credit card fraud is a complex event since it can only occur if many other events (simple or complex) have occurred before. In [52] events are classified into two classes:

- External events which are pushed to the CEP engine by external sources in runtime.
- Internal events which are inferred events generated within the CEP engine.

Every event is represented by an event instance containing all relevant information about the event. This information includes the occurrence time of the event, data that is relevant to applications that react to the event, and additional data that is needed in order to decide if a situation (complex event) has occurred.

Event data models

A data modelling approach was taken by [53] applying it to events using classification, aggregation, generalization and association abstractions in the event world.

- *Classification:* Events with similar characteristics are put in one event class. An event is a member of a single class
- *Aggregation:* An event is a set of attributes each of them with a value specific to this event. The attributes are defined at the class level, where the instances of these attributes are considered as a part of the event values. Attributes may have different types such as: numeric, string and references to objects and events.
- *Generalization:* A generalization is a subset hierarchy relationship among two event classes. It means that each of the elements in the set E1,…,En is a generalization of E.
- *Association:* An association is a relationship between two classes and a conditional expression

These abstractions can be placed into the following knowledge representation schema: an event has a set of attributes that are aggregated for each event instance including a temporal dimension (occurrence and event detection time) and a set of generalized events and a set of associated events.

[54] describes the importance of event representation and semantic enrichment for managing and reviewing emergency incident logs created by Emergency Response (ER) personnel to document the emergencies. Managing and reviewing these logs is a critical task for understanding and improving the implemented ER actions. Extensive manual effort is necessary to identify critical information, such as person names and locations, in order to align and merge the incoming log entries to make them suitable for review. To represent these entries, they employ the Event-Model-F [55] to model the participation in the events and attach further metadata and documentary evidence to support the event. The Event-Model-F allows the captures and represents the human experience through events. The model is based on the ontology DOLCE+DnS Ultralite (DUL) and provides support to represent time and space, objects and persons, as well as causal, and correlative relationships between events. The main application area of Event-Model-F is important in domains like emergency response, sports, news, and law.

[56] introduce the notion of time-annotated RDF (TA-RDF) as an extension of the RDF model which provides basic functionality for the representation and querying of time-related data. The time domain is a discrete, totally ordered infinite set intended to represent discrete time. The inspiration for time-annotating resources instead of triples is the fact that RDF is often employed for metadata annotations of data stored outside of RDF, so it useful to have a mechanism to refer to them according to their position in time.

Event Representation in Existing CEP Systems

There are other approaches for event representation. Distributed publish/subscribe architectures like presented in [57] and [58] represent an event as a set of typed attributes. AMIT ([53],[52]) is an event stream engine targeting the high-performance situation detection mechanism. It can model business situations based on event, situations, lifespan and keys. This approach is close to ours in the sense that they describe the importance of generalization and specialization of events.

The Aurora/Borealis system [59] has the goal to support various real-time monitoring applications. Data items are composed of attribute values called tuples. All tuples on the same stream have the same set of attributes. The system provides a toolset including a graphical query editor that simplifies the composition of streaming queries using a boxes and arrows interfaces.

The Cayuga system [60] can detect patterns in the event stream. An event is a tuple attribute value pair according to a schema. The goal of Stream project [61] is to be able to consider both structured data stream and stored data together. The Esper [62] CEP engine supports events in XML, Java-objects and simple attribute value pairs. There are also many CEP vendors like Tibco, Coral8/Aleri, StreamBase or Progress Apama providing tool suites including a development environment and a CEP engine. Most of the described systems use different SQL-like or XML based complex event pattern definitions and do not provide a semantic model for events and event patterns either.

Complex Event Processing

Complex event detection in Ode [63] is implemented using automata. Input for the automata is a stream of simple events. Ode transforms complex event expressions into deterministic finite automata. For sub-expressions which are complex events themselves, the process is done recursively. Atomic simple events are ultimately represented as automata of three states; a start state, an accepting state, entered upon detection of the simple event occurrence, and a non-accepting state, entered upon detection of any other simple event. Apart from providing the implementation, automata are a convenient model to define semantics of complex event operators. A downside of automata is that an automaton cannot accept overlapping occurrences of the same complex event.

Complex event detection in SAMOS [64] is implemented using Petri nets. Each primitive event type is represented by a Petri net place. Primitive event occurrences are entered as individual tokens into the network. Complex event expressions are transformed into places and transitions. Where constituent events are part of several expressions, duplicating transitions are used to connect the simple event with the networks requiring it. This results in a combined Petri net for the set of all event expressions. Petri nets, like automata provide a model of the semantics of event operators. Also the detection of overlapping occurrences is possible.

4.4.2 Progressing beyond the state of the art

We see two main areas of interest: using background knowledge to reason about event streams coming from a service instance and using statistical knowledge to approximate the matching/detection of the pattern of interest in event streams.

One avenue of future work is to investigate a combined probabilistic- and logic-based event processing framework. Our proposal is motivated by a few reasons. Use of formal specification of events and event-driven changes enable formal methods, particularly reasoning and verification, to achieve ultimate control of event-driven monitoring. In the proposed combined probabilistic and logic-based framework, complex events are detected through an inference procedure. Further, they are put in an appropriate context (current state) of the patient status. Reasoning about detected complex events with respect to current context enables discovery of suitable changes (demanded by a given situation of the system). Detection of complex events, reasoning about context, and a formal verification of proposed adaptations are all possible for realization in a logic-based Complex event Processing.

This sort of combined framework could be based on an existing CEP engine, such as the logic-based CEP engine ETALIS [65], which could be extended towards a probabilistic setting. ETALIS is a dedicated CEP system where complex event patterns are defined as logic rules. ETALIS Language for Events defines a set of operators and enables the specification of complex events from other atomic or complex events. ETALIS is built on novel algorithms for efficient event-driven computation. Since it is a logic-programming system, ETALIS features reasoning capabilities over background (domain) knowledge which could in the future get extended towards soft (possibly induced by machine learning) knowledge. Hence it can be a good starting point for developing a CEP system for above mentioned probabilistic pattern detection methods.

4.5 Social media management

4.5.1 Current state of the art

Social media management methodology

Recent technology trends and especially the emergence of social media have changed the face of online marketing over the last few years. The importance of social media to communicate with an organization's key stakeholders increases. In order to be able to use social media, organizations need to build up dedicated skills and resources. Currently, many organizations are lacking the dedicated skills and resources as shown for example by Fink *et al*. [67]. Companies start to be present in multiple platforms and are slowly learning what the benefits and risks of their online presence can be. The trend moves clearly towards the usage of new media possibilities, in some branches customers start expecting a company to be represented in various online media. This demand for an online presence brought a manifold of free as well as commercial social media suites on the market. Tools like HootSuite, SproutSocial, Sendible, Ping.FM (see Table 1) and many others aim to help their users shaping their social online presence by providing the capabilities to manage a selected set of social media channel via one access point. Still, a standard functionality with a consistent level of qualitative results has yet to be built [68].

Table 1 Social media management tools

Tool	Channels	Publishing	Feedback and Statistics
HootSuite[1]	a, b, c, d, g, h, i, mixi, (Google+)	Multi channel publishing, truncation of long text, not supported features are left out	Channel based feedback and customizable statistics
HubSpot[2]	a, b, c, d, e, f	Step-by-step guided publishing	Key-word based channel statistics and monitoring
SproutSocial[3]	a, b, c, d, Foursquare, Gowalla	Posting limited to 140 characters, a picture and shortened URLs.	Channel based feedback and customizable statistics, Google Analytics
Seesmic[4]	a, b, c, d, i, l , + 18 more	Multi channel publishing, common denominator is posted	Channel based feedback, simple statistics
Sendible[5]	a, b, c, h, i, j, k, l, m, + 11 more	Multi channel publishing, no content adaptation	Google Analytics for Sendible blog, monitoring for mentions, simple statistics
Moderation Market[6]	a, b, c, d	Multi channel publishing, no content adaptation	Channel based feedback and statistics

[1] http://hootsuite.com/

[2] http://www.hubspot.com/

[3] http://sproutsocial.com/

[4] https://seesmic.com/

[5] http://sendible.com/

[6] http://moderationmarketplace.com/

Ping.FM[7]	a, b, c, g, h, j, l, m, + 24 more	Multi channel publishing, truncation of too long texts	Not available
Media Funnel[8]Error! Reference source not found.	a, b, f, + 3 more	Single channel publishing, Restrictions of original channel	Channel based feedback and statistics, brand- and mention monitoring

a) Facebook, http://facebook.com; b) Twitter, http://twitter.com; c) LinkedIn, http://linkedin.com; d) Support for RSS feeds; e) Support for Email; f) YouTube, http://youtube.com; g) WordPress, http://wordpress.com; h) MySpace, http://myspace.com; i) ping.fm, http://ping.fm; j) flickr, http://flickr.com; k) SlideShare, http://SlideShare.com; l) Google Buzz, http://www.google.com/buzz (about to be shut down); m) FriendFeed, http://friendfeed.com/

Influence and dynamics in social media

Analyzing social media (articles, blog posts, tweets, etc.) has been a very popular yet challenging topic, especially when modeling the inherent dynamics, spreading of information and influence.

Social and mainstream media. There have been several approaches [70][71] which attempt to find a unifying global model of temporal variation, capturing patterns of response dynamics. Yang and Leskovec [72] study temporal patterns associated with online content from tweets, blog posts and news media articles from a different perspective. They translate this to a time series clustering problem. In order to scale to large datasets, they consider an adaptive wavelet-based incremental approach to clustering.

Social networks. Among the fundamental processes that occur in networks are diffusion and spreading of ideas, innovation, information, influence. Information and influence propagation in social networks has been studied in various fields such as sociology, communication, marketing, political science and physics. A number of studies [73][74] identified highly connected individuals as key information propagators. Jansen et al. [75] analyzed Twitter from the point of view of word-of-mouth advertising, considering a number of brands and products and examine the structure of the postings and the change in sentiments. Romero et al. [76] analyze the propagation of web links on Twitter over time, in order to capture the influence and the attention users receive. The model for influence is based on the concept of passivity in a social network and it utilizes both the structural properties of the network as well as the diffusion behavior among users; influence is quantified using an efficient algorithm similar to the HITS algorithm [77]. There are two fundamental challenges when studying network diffusion: tracking cascading processes by identifying the contagion in a network and further tracing the contagion. Gomez-Rodriguez et al. [78] develop a method based on a generative probabilistic model for tracing paths of information diffusion and influence through networks as well as inferring the networks over which contagions propagate. Their method is applied in the case of information cascades in blogs and news articles. In the case of news, the diffusion network tends to have a core-periphery structure — a small set of core media sites diffuse information to the rest of the Web; there are stable circles of influence with more general news media sites acting as connectors between them. Yang and Leskovec [79] propose a Linear Influence Model and focus on representing the global influence of a node in relation to the rate of diffusion through the (implicit) network, allowing capturing temporal dynamics of information diffusion. Their analysis shows that patterns of influence of individual participants differ significantly, depending on the topic of the information and the type of the node.

Blogs. Some approaches to analyzing blogs [80] have been to simultaneously model the topology and temporal dynamics of the blogosphere using a generative model for each individual blog. Agrawal et al. [81] investigated how to identify influential bloggers in the blogosphere, and discovered that the most influential bloggers were not necessarily the most active.

4.5.2 Progressing beyond the state of the art

When applying the appropriate social media activities, an organization needs to leverage the resources and create the desired value as Valos et al. [69] mention. Therefore, a significant direction for future work is to develop a social media management methodology that would support the selection and optimization of social media activities based on an organization's business objectives and context. In order to achieve this, a model of business objectives that organizations may want to achieve by applying social media and a social media

[7] http://ping.fm/

[8] http://sti.mediafunnel.com/

taxonomy which categorizes available social media activities, supports their specification and allows their optimization, would need to be developed.

The aforementioned approaches analyze social media from the point of view of an observer. A novel approach would be to combine control theory and social media analysis techniques in order to study a much more complex system. In this dynamic system the social media analysis results are used to influence the social media stream by publishing new information, the effects are further processed and provided as feedback to the system.

Emergence of preferred scales in nature

An assumption is that scale selection reflects the amplitude of coupling between parts of the system, i.e. how efficiently information is transferred between these parts. The way the information content of a quantity decays after coarse graining determines the scope of this quantity and reveals the significant scales. Similarly, identifying the parts of a system for which the information flow is weakly coupled reveals a possible decomposition into sub-systems. In other words, the damping of information across the scales and the amplitude of information flow across a system determines the splitting of a multiscale problem into many single-scale sub-problems. To clarify and quantify these ideas, numerical simulations of large scale microdynamical models can be considered, such as multi-agent or cellular automata models on graph. Then the goal would be to show how such models could be reformulated as a set of coupled single-scale models, with each sub-model acting at possibly different scales and with possibly a different dynamics. The space of possible partitioning (i.e. possible scale splitting) could be explored systematically and for each of them we can analyse its adequacy to represent the original system, in correlation with the way information connects these parts. Also, the concept of dissipation length and dissipation time of information will be considered as emerging scales that could hint at preferred scales of a system.

Criticality, Emergence, Tipping points

We plan to study critical effects in systems consisting of weakly-coupled subsystems (hierarchical multilevel structures). Criticality would be meant either as phase transitions or extreme phenomena while adaptation could take into account biologically-based population dynamics. Various types of topologies and mutual interactions would be considered, as in our earlier work on Ising systems. [83][84] Conclusions from such theoretical investigations would be applied for understanding of crisis propagation in economical networks. In this sense we will use the information from biological stability for economical systems.

Our hypothesis is that a combination of dissipation time and dissipation length of information, named information locality, must exceed some critical dissipation threshold before a critical transition can happen. This threshold may turn out to be a universal constant, and the corresponding information locality a universal property of any dynamical system. In other words, the information locality is a function of system parameters and may reach the critical dissipation threshold as parameters change, such as the temperature in a lattice of Ising spins or the concentration of nutrition in plankton populations [85].

More precisely, we feel that the product of the dissipation length and dissipation time of information has a threshold value above which elements can change state in a synchronized way; and below which elements behave either too randomly or too independently. These are all relevant questions in critical systems/transitions structures, which would give more insight in the 'tipping points' phenomena we have mentioned in the beginning. The general theoretical framework of information processing should/will be flexible enough to describe such changes in scales and will allow us to analyse critical transitions regarding information exchange leading to, and resulting from it.

Resilience to noise

How resilient is a complex system to randomness or noise, or to random failure? We could define it as the rate of decrease of how long a system remembers its own state, as function of increasing noise or failure. What are the necessary conditions for critical transitions? Information-theoretical measures such as the information dissipation length/time will allow us to analyse the system's resilience to noise. The information dissipation time and information locality depend on the fraction of noise, or failure, as it grows to 100%. Each system will have its own characteristic rate of decaying capability of information processing as function of noise or failure. The higher this value is, the more resilient the system. The two envisioned measures information persistence and information redundancy might be universal properties of complex system.

Extension of information processing to multiscale systems, inter-scale information processing

Very related to the investigation of preferred scales in nature, we will extend the formalism to information processing across different spatio-temporal scales. We will investigate what it means when we split the whole range of scales into sub-ranges representing single-scale entities and their dynamics, leaving out intermediate parts. Likewise, we will study whether the complexity of such intermediate ranges moved to an inter-scale information processing. And what error we make by a multi-scale decomposition and how that error relates to the information processing within, and between, the sub-ranges. Can we find conservation laws or other relations, regarding the information and complexity in a multi-scale complex system?

Extraction of dynamic representations appearing at multiple scales of aggregation.

These representations allow for the discovery of the scales where phase transition-type of events occur in the data, as well as tracking the emergent behaviour through stable regions. The utility of multiscale representations in analysis is apparent by their ubiquity (wavelets, fractional cascades, etc.) In particular, we will focus on three types of multiscale representations (a) semantic, (b) temporal, and (c) graph based:

- *Semantic* multiscale representations will be derived from the text representations augmented with hierarchical ontological knowledge (for example, Cyc and Open Directory). This will allow the plain textual data to be mapped into a hierarchical conceptual space. This gives a natural representation of increasingly general concepts and relations, allowing for the use of information processing techniques. The hierarchy encoded in the corresponding ontology can help determine the most suitable abstraction for a given analytic problem (e.g. different levels can correlate with different signals from the observed environment). The use of tree-like hierarchies allows for the efficient aggregation of local scale events in a coherent way [86]. Furthermore, such conceptual representations will be used with logic or probabilistic reasoning systems to derive new fragments of information not being explicitly observed within the text, hence filling in parts of the signal which may be missing, making the full behavioural structure of the data clear. For the reasoning, we will use the system Cyc with the largest existing common-sense knowledge base.
- *Temporal* multiscale representations [87] are necessary to observe emergent events. Multiscale behaviour in the time domain has been widely observed, although perhaps the most well-known examples come from chaos theory [88]. In nature, just as the atomic interactions occur orders of magnitude shorter scale as protein interactions, small events in Social Media in the shorter term can accumulate into a profound, long lasting event. The representation will be derived from the vector-based and the dynamic graph-based representations of content streams and social media. The goal is to transform extracted data into multiple interlinked temporal resolutions allowing flexible and efficient transitions across the levels. Depending on the target problems to be solved, the most suitable combination of levels can then be selected, much like large coefficients are selected in applications of classical frequency transforms. Such a representation will homogenize the data in the temporal dimension. The key approaches towards such a representation will be density based (for streams of vectors) and graph summarization (for dynamic networks).
- *Graph* multiscale representations of content streams will be built upon static and (dynamic)graphs extracted from text corpora and social media which allows for the aggregation of information on multiple levels. Extracting graph summaries and simplified graphs is currently a hot topic of study for data analysis [89]. The goal is to extend these and other techniques for graph summarization based on various forms of segmentation/clustering to re-represent networks at different resolutions. To process the graphs we used and further extend software library SNAP (co-developed by Stanford and Jozef Stefan Institute), which allows dealing very large networks (in the range of billion nodes).

These three modalities are not orthogonal and contain correlated information, with the strength of these interactions changing over the different scales. These interactions must be taken into account in order to obtain the complete picture of the behavioural dynamics of the systems. Therefore, we will develop a hybrid approach for multiscale representations combining all three of the above multiscale representations. In such a scenario, the content and social media streams of data would get compressed into the most suitable combination of semantic, temporal and network resolutions to allow for the study of the underlying system.

4.6 Network sampling

4.6.1 Current state of the art

Graph sampling is essential for efficient processing and analysis of large networks. We are often forced to pursue approximate results based on processing only one or more subsets of the graph. To ensure those are representative, we could aim for a sample where individual node and edge attributes, if any, are distributed similarly as in the full dataset; this requirement is present everywhere in statistics and is well understood. But in addition to that, we wish to retain the structural properties of the full graph. Just what these “structural properties” are is an open research question: what statistical features capture the graph structure well, which ones are the most relevant and should be preserved, and on a related note, which ones are universally preserved among graphs in a given domain?

Sampling locally visible graphs. There are two common reasons for doing graph sampling. Firstly, the sheer size of the network might exceed our limited computational resources. This is especially common if we are deriving nontrivial statistics and/or need to perform the analyses in near real time. Secondly, the graph might be hidden and its data difficult or expensive to obtain in total; only local views of the graph are available and even obtaining those implies some cost. Consider the example of a web crawler that wants to map the entire World Wide Web but can only see a single web page and its outbound links at a time; or a spammer who, using his viral Facebook app, wants to obtain a copy of the Facebook social network; or a sociologist that can only observe one person at a time but would like to make conclusions about the global social network; or an autonomous robot vehicle that needs to map out an unknown building.

Let us first focus on the second kind of problems. Historically, the first to systematically approach graph sampling was Goodman [96] who wanted to estimate the number of reciprocal “top *t* friends” links between the average person and his friends. He proposed *snowball sampling*, where we start with a random person and then iteratively extend the sample with *k* random friends of people added in the last iteration, and derived some theoretical guarantees for his sample statistic estimated on such a sample.

The topic of sampling graphs with only local visibility was most actively researched in the context of web crawlers; see e.g. [97] for a discussion. Not surprisingly, what works very well in this setting is the *random surfer model:* we start from a random node and then, arbitrarily many times, either move to one of the current node's neighbors with probability p or jump to the start node with probability $1 - p$. The sample consists of the nodes and edges visited this way. This provides a sample that is biased towards high-degree nodes, which might even be desired as the degree tends to correlate with node importance. If we wish to sample the nodes uniformly, appropriate reweighting is easy to do after a completed random walk. Alternatively, we can use the very similar Metropolis-Hastings random walk which directly results in a uniform sample. In addition, random walks are efficient and are therefore still the method of choice today [95].

As the model is simple, it also lends itself relatively well to adaptations. For example, [94] discusses sampling in graphs with multiple types of edges, a scenario easily encountered in social networks (e.g. “being friends”, “having shared a message”, “belonging to a common user group”, …). [100] improves the performance when we only wish to sample from a (yet unindentified) subset of nodes with a specific property.

Sampling globally visible graphs. When the whole graph is known and easily accessible, but merely unwieldy to process in its entirety, additional sampling methods are known to perform reasonably well. [101] made the first overview of methods and, more importantly, proposed a number of both local and global features that can be used to measure the sample's quality. Some examples from their paper and others:

- in- and out-degree distributions over nodes;
- the distribution of the number of nodes separated by at most *h* hops for all *h*;
- the *effective diameter*, i.e. the 90th percentile of shortest path lengths;
- the frequency distribution of network motifs of size 3, 4 and 5;

- the *clustering coefficient* distribution over node degrees; the clustering coefficient is a measure of the local density of the graph.

Of the several seminal methods evaluated in this paper, the trivial ones (uniformly random node sampling, uniformly random edge sampling) were shown to not preserve graph properties well. The best performing methods were the *forest fire model* [102] and the random surfer model, just as in the case with only locally observable graphs.

These methods were later expanded on by several authors, though often in a more specific setting. A result that convincingly shows better performance was obtained by [98]. Their method works by directly optimizing the difference in statistics between the full graph and its sample. They start with a uniformly random subgraph and then iteratively improve it in an MCMC fashion (one node at a time) in combination with simulated annealing.

Evaluation. As the task of graph sampling is relatively new and the notion of a good sample is application-dependent, evaluation techniques are not very standardized. However, it is almost universal that authors try to measure how well their sampling method maintains some chosen graph statistic(s) with respect to the full graph, notably the node degree distribution. It has been discovered by Barabási [91] and later confirmed by many others that many technological, social, and biological networks follow the power law distribution. In other words, if $P(k)$ is the probability of a node having degree k, then $P(ak)/\mathrm{P}(k)$ is independent of k; for this reason, such graphs are also called scale-free. The fact that BFS or snowball sampling does not preserve this property was first noted surprisingly recently by [104].

For an eyeball estimate on the quality of results we can expect from graph sampling, we refer to [101]: for each of the graph statistics they observe, they compute the maximal discrepancy $|S'(x) – S(x)|$ between the statistic's distribution S in the full graph and S' in the sample. For a 25% sample, the discrepancy is below 0.20 for most of the statistics.

Dynamic Networks. As graphs observed in the real world tend to evolve through time, sampling dynamic graphs is another active research topic. Of particular interest is the study of propagation of a phenomenon — a piece of information, a virus, a structural change — across the network. [92] provides a good and very recent overview of related work, though it does not focus on sampling only.

[101] already identified several temporal characteristics of evolving graphs and included them in the evaluation of sampling methods; these ideas are further expanded in [102]. They also identified two separate sampling tasks: one where you want the sample to be a “small copy” of the original and one where you aim to obtain a sample that look like the original used to look back when it was that small; however, a necessary assumption was that the entire “final” graph is known. Recently, [90] proposed an online sampling method suitable for real-time sampling of highly dynamic networks and markedly improved the forest fire model baseline. They do assume the network only grows, never shrinks. The method iteratively adds vertices to the sample, remembering the (simulated) time at which each node was added. In the end, the sampled vertices are connected with all the possible edges that were created in the full graph after the time at which both endpoint vertices were sampled.

Single-property oriented sampling. The methods discussed above mainly try to create a sample that emulates the full graph in several metrics at once, thus having to strike a compromises in accuracy but providing results that are more “interpretable” and “representative”. The objective is however often quite different when designing graph-theoretic algorithms — in those cases, we focus exclusively on (approximately) maintaining a *single property*, e.g. the min-cut weight or the minimum spanning tree cost. To illustrate the contrast with papers discussed so far, consider [105]: to find network motifs in the full graph, they first search for motifs in smaller subgraphs. The subgraphs however are obtained with a variant of BFS, which wrecks all the graph properties discussed before — except the network motifs. Graph sparsification is another sampling-based family of techniques that “destroys” the structure for a good purpose (Eppstein 1997, Spielman 2008); in this case, only edges get removed, leaving a sample with all the nodes but only $O(n \log^c n)$ edges.

In several of these stochastic algorithms, the main idea is, very broadly speaking, to solve the initial problem on several (perhaps carefully chosen) subsamples of the graph and then combine the results somehow. Alternatively, some approaches first sample the full graph, easily solve the initial problem on the subsample and then show that the solution need not, with high probability, be changed much in order to obtain the solution for the full graph.

Randomized algorithms based on graph sampling have had a lot of success recently; Karger's PhD thesis [99] is a small culmination, describing several fast and simple novel algorithms. As a somewhat famous example, the min-cut can be computed by iteratively and randomly collapsing graph edges until a single edge remains. Because the min-cut contains few edges, it will remain untouched with a reasonable probability until the very end and the two sets of points described by the two endpoints of the final edge are the min-cut. Rerun the algorithm several times and keep the best min-cut obtained this way.

Wavelets on Graphs. Just as sparsification changes the structure of a graph while preserving certain properties, wavelets on graphs [89] represent a harmonic approach to building smaller graphs. It defines an operator based on an elliptical differential operator, namely the diffusion map (or the related heat operator), and constructs a smaller graph whose operator norm is close to the original graph. By iterating this construction, it is possible to define wavelets on the graph and construct a multiscale representation of the graph.

Persistent Homology. Persistent homology was introduced in [108][109] and is an invariant from algebraic topology which is used in various forms in many areas of computer science (especially connected components in graph theory). The advantage of homology is that it can detect higher dimensional features such as holes and voids in continuous spaces. For graphs, we can construct proxies to approximate higher dimensional connectivity (called flag complexes) so that we can use homology to detect large features (e.g. a long cycle in the graph). Persistent homology gives a notion of scale to these features, allowing us to look at a sequence of graphs, or rather the graph at sequence of scales simultaneously. This make it possible infer the correct scale from the data. Furthermore, it is stable, so for a range of scale, we will compute the same structure, or put another way, the correct structure will be persistent. Persistent homology has found numerous applications based on graphs: to verify whether a sensor network has full coverage of a region [110][111], a variety of problems in networks [112], shape analysis [113], clustering [114], segmentation [115], and protein docking [116].

4.6.2 Progressing beyond the state of the art

- An extension of the state-of-the-art on network sampling would be to consider not only the structure of networks and their changing structure over time in an online model. While dynamic networks have been studied, subsampling or extracting simpler representation while taking the temporal domain into account is novel.

- Similarly, doing this in an online model is something which has not been looked at. To accomplish this, one could build on existing techniques as well as explore new directions for sampling based on vertex sparsification, graph wavelets and persistent homology, all of which capture the overall structure of a graph in different ways.

4.7 Network evolution

4.7.1 Current state of the art

As we saw earlier, networks are often used to model a set of entities from the problem domain together with some relationship between those entities. In many domains, these entities and relationships change over time; therefore the network itself also changes, leading to the concept of an evolving network, also known as a temporal network (or graph), time-varying graph [92], or volatile graph [124]. In an evolving infrastructural network, anomalies can occur in the form of disruptions and delays, the removal of nodes and edges, or, on the other hand, the appearance of new nodes and edges which present new opportunities for connectivity in the network. To be able to detect and handle such anomalies, many concepts and algorithms related to communications in a network, routing, information propagation etc. — such as paths, reachability, distance, diameter, connected components, or PageRank — can be adapted to the context of an evolving, changing network. The same concepts are also relevant in the study of evolving networks arising from patterns of social communication such as the web, blogs, social networks, phone and e-mail communications, and so on.

Network evolution also plays a prominent role in stochastic graph models, many of which try to model a random graph with similar characteristics as real-world social graphs by involving a stochastic generative process which gradually builds or evolves the network over time. Examples include the preferential attachment model [117][118], the small-world model [120], the community guided attachment model [119][119] and the forest fire model [105].

One of the patterns exhibited by many real-world networks during their evolution is known as *densification* [102]: as the network grows, the average degree of its vertices increases; thus the network becomes denser with time and the number of edges grows superlinearly in the number of nodes. A related phenomenon is *diameter shrinkage*, in which the diameter of the graph (maximum length of a shortest path between two vertices) often tends to decrease slowly, due to the increasingly good connectivity of the graph, rather than increasing as we might expect (due to the fact that the number of vertices is growing) [102].

There has also been work on how to compute PageRank in the context of an evolving graph [122], sometimes with the additional constraing that the graph is at any time only partly known to the computational process [121]. [123] has studied the problem of frequent subgraph mining in an evolving network.

An example of evolving network analysis that is very close to the problem of trend detection is the analysis of community evolution. This has been studied by [125], who developed decision-tree models for (1) predicting whether a user will join a community based on structural features such as the percentage of friends (i.e. neighboring nodes in the network) that are already part of that community, or the density of connections between those friends; and (2) predicting whether a community will grow significantly in the near future, based on structural features such as the size of the community's “fringe” (number of non-members of the community that have at least one friend in the community) relative to the size of the community itself. They identified “movement bursts” in which a number of users moved from one community to another within a short time frame.

A different approach to community evolution has been described in [128], which identified communities by clustering blog posts, and used the concept of *temporal smoothness* to make the clusters more stable through time.

[129] introduced a distinction between two types of community growth: *diffusion growth*, in which a user joins the community because of their connection to one or more existing members of the community, and *non-diffusion growth*, in which the user joins the community because of some feature of the community itself. They developed models that predict community growth and longevity based on a small set of structural features.

Another example of network evolution analysis has been based on latent semantic subspaces. [127] have described an approach in which each vertex is represented as a point in a “latent space”; these points have a probability of moving from one time step to another, and they have a probability of establishing edges depending on the distance betwen two points. A HMM-like approach is then used to estimate the coordinates of the points in latent space.

More recently, [124] described an integrated multi-level approach to anomaly detection in evolving networks. Their approach computes a number of metrics (global, community, and local ones) with several analysis techniques to identify anomalous changes in these metrics.

4.7.2 Progressing beyond the state of the art

Some of the interesting and important directions for progress beyond the state of the art in network evolution are:

- Develop distributed algorithms for network evolution, which would enable the real-time analysis of larger networks than hitherto [92].
- Optimization problems arising from network evolution. In some situations one has a degree of control over certain parameters of the network; this leads to the question of how to choose those parameters with a view of ensuring that the resulting network will have certain properties or satisfy certain constraints [92].
- Study the computation of PageRank and similar measures under different models of graph evolution [121].
- Extend the established graphical models for the evolution of topics over time in textual time series [126] with network-related features.
- Visualization. A suitable visualization method could potentially help the users understand the evolution of their network at a better level, but many of the established graph visualization methods are unsuitable for large complex networks that usually occur in network evolution problems.

5 Conclusions

We presented an overview of trend and anomaly detection on unstructured data, with an emphasis on text and network data, and discussed the state-of-the-art and directions of future work for several areas related to trend and anomaly detection on unstructured data: text processing of informal documents, online learning, adaptive data summarization, complex event processing, social media management, network sampling, and network evolution.

Trend and *anomaly detection* is the process of discovering patterns in the data that do not conform to normal or expected behaviour of the data stream; the main difference between the two is that a trend is not merely an aberration from normal behaviour but an actual change in what normal behaviour is. *Non-structured* or *unstructured* data is data that doesn’t conform to an explicit and well-defined formal data model with relations, attributes etc. This includes textual data, multimedia data (images, video), and networks.

Trend and anomaly detection draw upon techniques from several related disciplines, such as machine learning, data mining, text mining, statistics and information theory, natural language processing, etc. The approaches to trend and anomaly detection involve classification, clustering, nearest-neighbour approaches, statistical and information-theoretic approaches, and spectral methods. Applications of trend and anomaly detection can be found in numerous areas, such as sensor networks, cyber-intrusion detection, fraud detection, medicine, fault detection (in networks, manufacturing, etc.), image processing, sensor networks, topic detection (from news feeds) etc.

References

[1] M. Manuja, D. Garg. Semantic web mining of un-structured data: Challenges and opportunities. *International Journal of Engineering*, 5(3):268-276 (2011).

[2] V. Chandola, A. Banerjee, V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3), Article No. 15, July 2009.

[3] M. Markou, S. Singh. Novelty detection: A review. Part 1: Statistical approaches; Part 2: Neural network based approaches. Signal Processing, 83(12):2481-2521, December 2003.

[4] S.Sathe, et al. Efficient sensor data management techniques. Planet Data Deliverable 1.3, 2012.

[5] Efficient sensor data management techniques

[6] S. Marsland. Novelty detection in learning systems. *Neural Computing Surveys*, 3:1-39 (2002).

[7] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, J. Platt. Support vector method for novelty detection. *NIPS* 1999, pp. 582-588.

[8] A Theory of Learning from Different Domains. Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jenn Wortman. Machine Learning. Special Issue on Learning From Multiple Sources.

[9] N. Bertoldi, M. Cettolo, M. Federico. Statistical Machine Translation of Texts with Misspelled Words. NAACL HLT 2010.

[10] John Blitzer. Domain Adaptation of Natural Language Processing Systems. PhD Thesis. University of Pennsylvania, 2008.

[11] B. O'Connor et al. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. ICWSM 2010.

[12] M. Efron. Hashtag retrieval in a microblogging environment. SIGIR 2010.

[13] J. Foster. “cba to check the spelling”: Investigating Parser Performance on Discussion Forum Posts. NAACL HLT 2010.

[14] S. Gerani, M. J. Carman, F. Crestani. Proximity-based opinion retrieval. SIGIR 2010.

[15] R. Huang, E. Riloff. Inducing Domain-Specific Semantic Class Taggers from (Almost) Nothing. ACL 2010.

[16] G. Inches, M. J. Carman, F. Crestani. Statistics of online User-generated short Documents. ECIR 2010.

[17] D. Irani, S. Webb, C. Pu. Study of Static Classification of Social Spam Profiles in MySpace. ICWSM 2010.

[18] A. Lampert, R. Dale, C. Paris. Detecting Emails Containing Requests for Action. NAACL HLT 2010.

[19] Y. Lee, H.-Y. Jung, W. Song, J.-H. Lee. Mining the blogosphere for top news stories identification. SIGIR 2010.

[20] M. Potthast, S. Becker. Opinion Summarization of Web Comments. ECIR 2010.

[21] K. C. Park, Y. Jeong, S. H. Myaeng. Detecting Experiences from Weblogs. ACL 2010.

[22] J. Parapar, J. López-Castro, A. Barreiro. Blog snippets: a comments-biased approach. SIGIR 2010.

[23] S. Petrović, M. Osborne, V. Lavrenko. Streaming First Story Detection with application to Twitter. NAACL HLT 2010.

[24] A. Ritter, C. Cherry, B. Dolan. Unsupervised Modeling of Twitter Conversations. NAACL HLT 2010.

[25] D. Ramage, S. Dumais, D. Liebling. Characterizing Microblogs with Topic Models. ICWSM 2010.

[26] B. Sriram et al. Short text classification in twitter to improve information filtering. SIGIR 2010.

[27] B. Sharifi, M.-A. Hutton, J. Kalita. Summarizing Microblogs Automatically. NAACL HLT 2010.

[28] S. Singh, D. Hillard, C. Leggetter. Minimally-Supervised Extraction of Entities from Text Advertisements. NAACL HLT 2010.

[29] O. Tsur, D. Davidov, A. Rappoport. ICWSM — A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews. ICWSM 2010.

[30] S. Tata, B. Di Eugenio. Generating Fine-Grained Reviews of Songs from Album Reviews. ACL 2010.

[31] W. Wei, J. A. Gulla. Sentiment Learning on Product Reviews via Sentiment Ontology Tree. ACL 2010.

[32] J. Wang, Q. Li, Y. P. Chen. User comments for news recommendation in social media. SIGIR 2010.

[33] Y.-C. Wang, C. P. Rosé. Making Conversational Structure Explicit: Identification of Initiation-response Pairs within Online Discussions. NAACL HLT 2010.

[34] W. Wu, B. Zhang, M. Ostendorf. Automatic Generation of Personalized Annotation Tags for Twitter Users. NAACL HLT 2010.

[35] P. Domingos and G. Hulten. Mining high-speed data streams. Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2000.

[36] G. Cavallanti, N. Cesa-Bianchi, and C. Gentile. Linear algorithms for online multitask classification. Journal of Machine Learning Research, 11:2597-2630, 2010.

[37] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni. Incremental algorithms for hierarchical classification. Journal of Machine Learning Research, 7:31--54, 2006.

[38] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Geometric approximations via coresets. Combinatorial and Computational Geometry - MSRI Publications, 52:1–30, 2005.

[39] A. Czumaj and C. Sohler. Sublinear-time approximation algorithms for clustering via random sampling. Random Struct. Algorithms (RSA), 30(1-2):226–256, 2007.

[40] D. Feldman and M. Langberg. A unified framework for approximating and clustering data. In Proc. 41th Annu. ACM Symp. on Theory of Computing (STOC), 2011.

[41] S. Har-Peled and S. Mazumdar. On coresets for k-means and k-median clustering. In Proc. 36th Annu. ACM Symp. on Theory of Computing (STOC), pages 291–300, 2004.

[42] D. Feldman, M. Monemizadeh, and C. Sohler. A PTAS for k-means clustering based on weak coresets. In Proc. 23rd ACM Symp. on Computational Geometry (SoCG), pages 11–18, 2007.

[43] D. Feldman, A. Fiat, and M. Sharir. Coresets for weighted facilities and their applications. In Proc. 47th IEEE Annu. Symp. on Foundations of Computer Science (FOCS), pages 315–324, 2006.

[44] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. Inf. Comput., 100(1):78–150, 1992.

[45] S. Har-Peled and K. R. Varadarajan. High-dimensional shape fitting in linear time. Discrete & Computa- tional Geometry, 32(2):269–288, 2004.

[46] M.W. Mahoney and P. Drineas. CUR matrix decompositions for improved data analysis. Proceedings of the National Academy of Sciences, 106(3):697, 2009.

[47] G. Frahling and C. Sohler. Coresets in dynamic geometric data streams. In Proc. 37th Annu. ACM Symp. on Theory of Computing (STOC), pages 209–217, 2005.

[48] D. Golovin, A. Krause, Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization, In Journal of Artificial Intelligence Research (JAIR), vol. 42, pp. 427-486, 2011.

[49] Hui Lin and Jeff Bilmes. A Class of Submodular Functions for Document Summarization. In The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL/HLT-2011), 2011.

[50] Luckham, D.C., 2001. The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems, Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc

[51] Chandy, K.M., Charpentier, M. and Capponi, A., 2007. Towards a theory of events. In DEBS '07: Proceedings of the 2007 inaugural international conference on Distributed event-based systems. New York, NY, USA: ACM, S. 180–187.

[52] Adi, A. and Etzion, O., 2004. Amit - the situation manager. The VLDB Journal, 13(2), 177–203.

[53] Adi, A., Botzer, D. and Etzion, O., 2000a. Semantic Event Model and its Implication on Situation Detection. In ECIS.

[54] Papadopoulos, S. u. a., 2010. Using event representation and semantic enrichment for managing and reviewing emergency incident logs. In Proceedings of the 2nd ACM international workshop on Events in multimedia. EiMM '10. New York, NY, USA: ACM, S. 41–46. Available at: http://doi.acm.org/10.1145/1877937.1877950.

[55] Scherp, A. u. a., 2009. F–a model of events based on the foundational ontology dolce+DnS ultralight. In Proceedings of the fifth international conference on Knowledge capture. K-CAP '09. New York, NY, USA: ACM, S. 137–144. Available at: http://doi.acm.org/10.1145/1597735.1597760.

[56] Rodriguez, A., McGrath, R. & Yong, L., 2009. Semantic Management of Streaming Data. In Workshop on Semantic Sensor Nets at International Semantic Web Conference.

[57] Carzaniga, A., Rosenblum, D.S. & Wolf, A.L., 2001. Design and evaluation of a wide-area event notification service. ACM Trans. Comput. Syst., 19(3), 332–383.

[58] Aguilera, M.K. u. a., 1999. Matching events in a content-based subscription system. In PODC '99: Proceedings of the eighteenth annual ACM symposium on Principles of distributed computing. New York, NY, USA: ACM, S. 53–61.

[59] Abadi, D. u. a., 2003. Aurora: a data stream management system. In SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data. New York, NY, USA: ACM, S. 666–666.

[60] Brenna, L. u. a., 2007. Cayuga: a high-performance event processing engine. In SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data. New York, NY, USA: ACM, S. 1100–1102.

[61] Arasu, A. u. a., 2004. STREAM: The Stanford Data Stream Management System, Stanford InfoLab. Available at: http://ilpubs.stanford.edu:8090/641/.

[62] Esper, 2009. Esper Version 3.2.0, EsperTech Inc., Available at: http://bit.ly/cGwY5W.

[63] Chakravarthy, 1997. S. Chakravarthy, SENTINEL: An Object-Oriented DBMS With Event-Based Rules, In SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, pages 572-575, May 13-15, 1997, Tucson, Arizona, USA. ACM Press, 1997.

[64] Gatziu u.a. 1994. S. Gatziu, K. Dittrich, Detecting composite events in active database systems using Petrinets, In Proc. Fourth International Workshop on Active Database Systems Research Issues in Data Engineering, pages 2-9, 1994.

[65] Anicic u. a. 2010. Darko Anicic, Paul Fodor, Sebastian Rudolph, Roland Stühmer, Nenad Stojanovic, Rudi Studer. A Rule-Based Language for Complex Event Processing and Reasoning, RR 2010: Proceedings of the International Conference on Web Reasoning and Rule Systems, 2010.

[66] Andreas Krause, Carlos Guestrin. Submodularity and its Applications in Optimized Information Gathering. In *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 4, 2011.

[67] See for example Stephan Fink, Ansgar Zerfass, Anne Linke: social media Governance 2011 – Expertise, Structures and Strategies of Companies, Governmental Institutions and Non-Profit Organizations communicating on the Social Web, 2011.

[68] Kasper, Harriet; Dausinger, Moritz; Kett, Holger; Renner, Thomas: Marktstudie social media Monitoring Tools. Stuttgart: Fraunhofer Verlag, 2010. http://www.e-business.iao.fraunhofer.de/publikationen/marketing/beschreibungen/orm.jsp (last checked 14.12.2011)

[69] Valos, Michael; Maritz, Alex; Frederick, Howard: The role of entrepreneurial marketing in social media. Proceedings of the 8th International Entrepreneurship Research Exchange, 2011.

[70] A.-L. Barabási. The origin of bursts and heavy tails in human dynamics. *Nature*, pp. 435-207, 2005.

[71] R. Crane and D. Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *In proceedings of PNAS*, 105(41), pp15649–15653, October 2008

[72] J. Yang, J. Leskovec. Patterns of Temporal Variation in Online Media. *In proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, 2011.

[73] Amit Goyal, Francesco Bonchi and Laks V.S. Lakshmanan. Learning Influence Probabilities in Social Networks. *In proceedings of the ACM International Conference on Web Search and Data Mining* (WSDM), 2010.

[74] P. Domingos and M. Richardson. Mining the network value of customers. *In proceedings of SIGKDD*, 2001.

[75] B. Jansen, M. Zhang, K. Sobel, and A. Chowdury. Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, 2009.

[76] D.M. Romero, W. Galuba, S. Asur, and B.A. Huberman. Influence and Passivity in social media. *In proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, 2011

[77] Jon Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM 46 (5)*: pp. 604 -632, 1999.

[78] M. Gomez-Rodriguez, J. Leskovec, A. Krause. Inferring Networks of Diffusion and Influence. *In press ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2011.

[79] J. Yang, J. Leskovec. Modeling Information Diffusion in Implicit Networks. *In proceedings of IEEE International Conference On Data Mining (ICDM)*, 2010.

[80] M. Goetz, J. Leskovec, M. Mcglohon, C. Faloutsos. Modeling Blog Dynamics. *In proceedings of AAAI Conference on Weblogs and social media (ICWSM)*, 2009.

[81] N. Agarwal, H. Liu, L. Tang, P. S. Yu. Identifying the Influential Bloggers in a Community. *In proceedings of the ACM International Conference on Web Search and Data Mining* (WSDM), 2008

[82] Enrycher, http://enrycher.ijs.si/

[83] Suchecki, K. and J.A. Hołyst, Ising model on two connected Barabasi-Albert networks. Physical Review E, 2006. 74(1): p. 011122.

[84] Suchecki, K. and J.A. Hołyst, *Bistable-monostable transition in the Ising model on two connected complex networks.* Physical Review E, 2009. **80**(3): p. 031110.

[85] Drake, J.M. and B.D. Griffen, *Early warning signals of extinction in deteriorating environments.* Nature, 2010. **467**(7314): p. 456-459.

[86] Kompatsiaris, Y. and P. Hobson, *Semantic multimedia and ontologies: theory and applications*.

[87] Lian, X., L. Chen, J. Yu, J. Han, and J. Ma, *Multiscale Representations for Fast Pattern Matching in Stream Time Series.* IEEE Trans. on Knowl. and Data Eng., 2009. **21**: p. 568-581.

[88] Gao, J., T. Cao, W. Tung, and J.Hu, *Multiscale Analysis of Complex Time Series: Integration of Chaos and Random Fractal Theory, and Beyond*

[89] Coifman, R.R., S. Lafon, A.B. Lee, M. Maggioni, B. Nadler, F. Warner, and S.W. Zucker, *Geometric diffusions as a tool for harmonic analysis and structure definition of data: Multiscale methods.* PNAS, 2005.

[90] N. Ahmed, F. Berchmans, J. Neville. Time-based sampling of social network activity graphs. *Learning with Graphs* (2010).

[91] A. Barabási. Emergence of Scaling in Random Networks. *Science* 286:509-512 (1999).

[92] A. Casteigts, P. Flocchini, W. Quattrociocchi, N. Santoro. Time-varying graphs and dynamic networks. *Proceedings of the 10th International Conference on Adhoc Networks and Wireless*, 2011, pp. 346–359.

[93] D. Eppstein, Z. Galil, G. F. Italiano, A. Nissenzweig. Sparsification — a technique for speeding up dynamic graph algorithms. *Journal of the ACM* (JACM) 44:669–696 (1997).

[94] M. Gjoka, C. T: Butts, M. Kurant, A. Markopoulou. Multigraph Sampling of Online Social Networks. *JSAC special issue on Measurement of Internet Topologies* 1-13 (2011).

[95] M. Gjoka, M. Kurant, C. T. Butts, A. Markopoulou. Practical Recommendations on Crawling Online Social Networks. *JSAC special issue on Measurement of Internet Topologies* 1-21 (2011).

[96] L. A. Goodman. Snowball sampling. *The Annals of Mathematical Statistics* 32:148–170 (1961).

[97] M. R. Henzinger, A. Heydon, M. Mitzenmacher, M. Najork. On near-uniform URL sampling. *Computer Networks* 33:295–308 (2000).

[98] H. C. Hübler, H.-P. Kriegel, K. Borgwardt, Z. Ghahramani. Metropolis Algorithms for Representative Subgraph Sampling. In: *ISCDM '08* (2008).

[99] D. Karger. *Random sampling in graph optimization problems* (1995).

[100] M. Kurant, M. Gjoka, C. T. Butts, A. Markopoulou. Walking on a Graph with a Magnifying Glass: Stratified Sampling via Weighted Random Walks Categories and Subject Descriptors. In: *SIGMETRICS '11* (2011).

[101] J. Leskovec, C. Faloutsos. Sampling from large graphs. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 631–636 (2006).

[102] J. Leskovec, J. Kleinberg, C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 2007.

[103] D. A. Spielman, N. Srivastava. Graph sparsification by effective resistances. In: *Proceedings of the 40th annual ACM symposium on Theory of Computing*, pp. 563–568 (2008).

[104] M. P. H. Stumpf, C. Wiuf, R. M. May. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proceedings of the National Academy of Sciences of the United States of America* 102:4221-4 (2005).

[105] R. Zou, L. B. Holder. Frequent subgraph mining on a single large graph using sampling techniques. In: *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pp. 171–178 (2010).

[106] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Multiscale methods. *Proceedings of the National Academy of Sciences*, 2005.

[107] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences*, 2005

[108] H. Edelsbrunner, D. Letscher, A. Zomorodian. Topological Persistence and Simplification. *Discrete and Computational Geometry*, 2002.

[109] G. Carlsson, A. Zomorodian. Computing Persistent Homology. *Discrete and Computational Geometry*, 2005.

[110] V. de Silva, R. Ghrist. Homological sensor networks. *Notices Amer. Math. Soc.* 54(1), 2007.

[111] V. de Silva, R. Ghrist. Coverage in sensor networks via persistent homology. *Alg. & Geom. Topology*, 2007.

[112] P. Skraba, Q. Fang, A. Nguyen, L. Guibas. Sweeps over wireless sensor networks. In *Int'l Conference on Information Processing in Sensor Networks* (IPSN), 2006.

[113] T. K. Dey, K. Li, C. Luo, P. Ranjan, I. Safa, Y. Wang. Persistent heat signature for pose-oblivious matching of incomplete models. In *Symposium of Geometric Processing*, 2010.

[114] F. Chazal, L. Guibas, S. Oudot, P. Skraba. Persistence-based Clustering on Riemannian Manifolds. In *Symposium of Computational Geometry*, 2011.

[115] P. Skraba, M. Ovsjanikov, F. Chazal, L. Guibas. Persistence-based Segmentation of Deformable Shapes. *3rd Workshop on Non-Rigid Shape Analysis and Deformable Shapes, Proc. CVPR*, 2010.

[116] P. K. Agarwal, H. Edelsbrunner, J. Harer, Y. Wang. Extreme Elevation on a 2-Manifold. *Discrete and Computational Geometry*, 2006.

[117] A.-L. Barabási, R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509-512, 15 December 1999

[118] S. Ravi Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. S. Tomkins, E. Upfal. The web as a graph. *Proc. of the 19th ACM Symposium on Principles of Database Systems*, PODS 2000, pp. 1-10.

[119] J. Leskovec, J. Kleinberg, C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD), 2005.

[120] D. J. Watts, S. H. Strozatz. Collective dynamics of “small-world” networks. *Nature*, 393:440-442, 4 June 1998.

[121] B. Bahmani, R. Kumar, M. Mahdian, E. Upfal. PageRank on an evolving graph. *KDD 2012*, pp. 24-32

[122] S. Chien, C. Dwork, S. Kumar, D. Simon, D. Sivakumar. Link evolution: Analysis and algorithms. *Internet Mathematics*, 1(3):277-304, 2003.

[123] A. Bifet, G. Holmes, B. Pfahringer, R. Gavalda. Mining frequent closed graphs on evolving data streams. *KDD 2011*, pp. 591-599.

[124] K. Henderson, T. Eliassi-Rad, C. Faloutsos, L. Akoglu, L. Li, K. Maruhashi, B. A. Prakash, H. Tong. Metric forensics: a multi-level approach for mining volatile graphs. *KDD 2010*, pp. 163-172.

[125] L. Backstrom, D. P. Huttenlocher, J. M. Kleinberg, and X. Lan. Group formation in large social networks: membership, growth, and evolution. In *KDD 2006*, pp. 44-54.

[126] X. Wang, A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. *Proc. 12th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2006.

[127] P. Sarkar, A. Moore. Dynamic social network analysis using latent space models. *SIGKDD Explorations*, 7(2):31-40, 2005.

[128] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In KDD, pages 153–162, 2007.

[129] S. Kairam, D. Wang, J. Leskovec. The Life and Death of Online Groups: Predicting Group Growth and Longevity. *ACM International Conference on Web Search and Data Mining* (WSDM), 2012.