A SURVEY OF NETWORK MODELING

by

Unique Wells

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Computer Science

Charlotte

2018

Approved by:

_____

Dr. Aidong Lu

_____

Dr. Dewan Ahmed

_____

Dr. Weichao Wang

ABSTRACT

UNIQUE WELLS. A SURVEY OF NETWORK MODELING. (Under the direction of DR. AIDONG LU)

Network modeling is a set of techniques for extracting various patterns of interest from graphs. They can be used in many real-life analytic fields, such as sales and marketing, journal reports, government and law enforcement, social networking sites, and technical research. The papers in this survey contribute to advertising by user interest, detecting organizations in social networks, groups of shared political policies, etc.

This survey collects a number of network modeling papers from several research fields: statistics, data mining, machine learning, and signal processing, and three other related fields. We also summarize the papers based on the taxonomy and provide examples based on their tasks. We expect that this survey will be used to introduce complex network modeling techniques to the communities that are not familiar with network modeling techniques, and provide quick reference to identify suitable methods to different analysis tasks.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Aidong Lu, for helping and guiding me through this project, and my committee members Dr. Dewan Ahmed and Dr. Weichao Wang for their time and attention in my presentations and writings.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Network modeling is a topic within graph mining, which is to extract patterns of interest from graphs. This topic is meant to describe the underlying data and could be used for tasks like classification or clustering. It is an approach for "mining" the data from the graph structure representing it. However, real datasets are often massive, complex, multi-attributed, and ordered by a time sequence. This leads to several challenges for the graphical representations of the data, such as them appearing too large, messy, and dense.

Researchers are constantly constructing new ways to make network model analysis more efficient and less complex. As the development of networks continues to improve, so do the approaches for its modeling and analysis. These studies build upon each other, inspired by past accomplishes and adapting existing algorithms to improve a certain task. This purpose of this survey is to provide guidance for researchers of communities that are not familiar with network modeling techniques for better ways to model graph data. The main contribution of this survey is a taxonomy that categorizes and classifies a set of selected papers from four important research fields. It is divided by task and organized by published conferences.

Section 2 of the thesis discusses the main tasks of graph mining divided into static and dynamic network adaptations. Many of the papers involve and encompass each other with tasks such as anomaly detection, community identification, the capture of

node evolution, etc. The selected papers mostly share common groups of tasks for each network, but the few unique ones will be described in an "Other" subsection. Section 3 is where the compilation starts, beginning with static network analyses. The discussion is broken up by similar tasks, and ended with the group of different objectives per paper. The layout is similar in Section 4, under dynamic network analyses. Finally, Section 5 contains the conclusion and talk of future works.

CHAPTER 2: CLASSIFICATION OF NETWORK MODELING TECHNIQUES

Network model analysis is all about being able to derive meaning from data representations of a graph. Based on the data features, we mainly divide all the work in this survey into two groups: static and dynamic networks. Static network models focus on individual sets of data, while dynamic network models observe the change of data over time. As networks have developed from mere static snapshots to how they are now in, for example, social networks, there are more datasets available for dynamic model analysis.

The selected papers have been further divided into the common main tasks: community & community evolution detection, anomaly detection, entity linking, event and pattern detection, and other miscellaneous tasks. These tasks were reoccurring themes in the research papers, and many handled identical or very similar datasets, which showed showed the significance of the tasks to network modeling. Many of the researches' contributions are encompassing, as in their models and frameworks may fulfill several tasks, but they have been grouped by which task they concentrate on.

In the following, we provide a list of important tasks.

Community Detection In complex networks, community detection or structure is defined as, "the organization of vertices in clusters, with many edges joining vertices of the same cluster and comparatively few edges joining vertices of dif-

ferent clusters [For10]." Such communities should appear as components of the overall graph. Under a computer science study, an example would be members of an organization in a social network such as Facebook. In dynamic models, the studies are concerned with detecting the evolution of communities.

Anomaly Detection Anomaly detection can be viewed a branch of event and pattern detection, but focuses on data considered odd or wrong, such as fraud [EH07], or even scanning shipments. This is particularly useful in scientific fields such as biology where human physiology is often visualized. Following that, detecting illness in the human body would be example an example of anomaly detection.

Entity Linking "Many real-world domains are relational in nature, consisting of a set of objects related to each other in complex ways [TfWAK03]." For the sake of this survey, entity or actor linking uses relational data for finding groups and the existence and connections between entities, latent or not. This is moreso used among textual data, such as in [BC11]. Examples here would be in finding research papers with linked citations, and Google searching with relevant keywords.

Event Detection Event detection is useful in transportation networks, news networks, etc.. It is not necessarily a task like anomaly detection, but behavior tracking. For example, an incident gaining traction and popularity and turning into news. On Twitter, trending events have their own tab where users can inform themselves of the incidents. Pattern detection is similar, but it focuses

more on the regularities in data. In this survey, event and pattern detection have been paired for synonymous brevity.

Others Though these few research papers do not neatly group into the other tasks, they still relate and are important in this compilation. The other tasks in the static networks collection involve improved models and frameworks as contributions for signal processing and handling static big data. In the dynamic networks collection, the other tasks involve improvements for spectral clustering and solutions for its problems, reviewing statistical models for networks, and the dynamic approach for analyzing big data.

2.1 Taxonomy

The constructed taxonomy shown in Table 1 lists the papers and which tasks they fulfill. The table is ordered by their publications in conferences or journals focused on networking, sociology, statistics, data mining, machine learning, and signal processing. These fields contribute to network modeling either by methodology, approaches, or analysis. Whatever the perspective might be, from several different research fields, they are all trying to reach the same goal of understanding complex, real data. Much like the tasks of each paper, some of the publications may be interdisciplinary. They were organized and labeled to the best of my understanding.

2.2 Common Data

Table 2 lists the data that each paper experimented with, in alphabetical order of data used. Some notes: DBLP is bibliography website for computer science information. Enron short for Enron email is a dataset containing about 500,000 emails

from senior employees of the old company. The “monks in a monastery” refers to a collection of information taken at an American monastery. NIPS datasets are neural information processing systems papers found in the website, Kaggle.

Table 1: 33 selected papers about network and analysis organized by fields and tasks. Field key: Red = Other (Networking, INFOCOM, Transportation); Yellow = Sociology; Orange = Statistics; Green = Data Mining; Blue = Machine Learning; Purple = Signal Processing.

Network Modeling										
Paper	**Static Network**					**Dynamic Network**				
	Community ID	Anomaly ID	Entity Link	Event Detect	Other	Commu. Evol. ID	Anomaly ID	Entity Link	Event Detect	Other
[DAM+14]		*		*	*					
[LCZ+08]						*		*		
[PBL14]	*		*	*						
[HRT07]			*							
[HX07]								*		
[Sni11]								*	*	
[AX08]								*		
[BC11]			*							
[CSZ+07]										*
[KHRH09]	*		*							
[RGNH13]							*		*	
[SDB+16]	*				*					
[STH+10]						*			*	
[TLZN08]						*		*	*	
[TPS+08]	*		*	*		*		*	*	
[XZYL08]								*		*
[YL15]	*									
[YML14]	*									
[FSX09]						*	*	*		
[GNC10]					*					
[HXA15]						*		*		
[NXC+10]									*	
[YCZ+11]						*			*	
[ABFX09]			*							
[CK03]		*		*						
[HVG09]				*	*					
[KL13]								*		
[OKI13]							*	*		
[QVUARHS12]									*	
[SM14]					*					*
[SNF+13]					*					
[XI14]								*		*
[Xu14]								*		

Table 2: Summary of data and models presented in the survey.

Data/Model		
Paper	**Data**	**Model**
[Sni11]	N/A	Social net statistical models
[STH+10]	DBLP	Evo-NetClus
[FSX09]	Enron	dynamic MMB (dMMSB)
[OKI13]	Enron	Latent variable models & methods
[RGNH13]	Enron	Dynamic behavioral mixed-membership model (DBMM)
[TLZN08]	Enron & DBLP	Evol. Multi-mode Cluster
[XI14]	Enron	Multi-layer extension of DSBM
[SNF+13]	Image filters, brain graph	signal processing tutorial
[QVUARHS12]	IRVINE & METAFILTER	Regression-based framework
[HXA15]	MIT Reality Mining	Multi-graph SBM
[HRT07]	Monks in a monastery	Latent position cluster model
[KHRH09]	Monks in a monastery	Latent cluster rand. effects model
[SM14]	Nationwide temperatures	Discrete signal process. paradigm
[CSZ+07]	NEC blogs	PCQ & PCM
[LCZ+08]	NEC blogs	FacetNet
[NXC+10]	NEC blogs	Incidence vector/matrix models
[XZYL08]	News Groups	DPChain & HDP-EVO
[KL13]	NIPS & DBLP	Dynamic Multi-group Membership Graph Model
[ABFX09]	Protein interactions	Mixed Member. BM
[BC11]	Protein interactions	Block-LDA
[HX07]	Senate proposals	Exponential rand. graph (ERGM)
[PBL14]	Social networks	Descriptive Community Mining
[Xu14]	Social networks	stochastic block transition model (SBTM)
[YCZ+11]	Social networks	Dynamic stochastic blockmodel (DSBM)
[YL15]	Social networks	Ground-truth net
[YML14]	Social networks	CESNA
[HVG09]	Swiss roll	Spectral graph wavelet transform (SGWT)
[CK03]	Traffic network	Spatial traffic analysis
[DAM+14]	Traffic network	Spectral graph wavelets analysis
[SDB+16]	Traffic, genes, Wiki, blogs	Sparse graph wavelet transform (SWT)
[TPS+08]	Traffic network	Colibri
[AX08]	Restaurant seating	Temporal Dirichlet Process Mixture (TDPM)
[GNC10]	USPS benchmark	Harmonic analysis framework

CHAPTER 3: STATIC NETWORKS

Many basic network models are static in nature, especially from earlier times of such technical research. The analysis of static network models focus on certain network statistics, say a snapshot at a certain time, in hopes to find the main components of actual networks. Brief introductions of the each paper's approaches for each task will be provided before going into detail.

3.1 Community detection

Communities are groups of vertices that share common properties or play similar roles [For10]. For examples, Tong et. al [TPS+08] approached community detection by preserving the sparsity of large, sparse graphs, unlike preceding research. They find an independent basis by removing linearly dependent attributes, which improves space and time needed for computation.

Before the research of Bonchi et. al [PBL14], there was negligence towards informational descriptions beyond graph structure, which lead to the use of additional information (user behavior, interests, etc.) and overlapping community detection. They contributed explainable models with a new framework that pairs k possible overlapping communities with their corresponding descriptions, with k as a given number, using a fast-and-active algorithm.

Silva et. al [SDB+16] tried to solve the problem of computing wavelet trees that

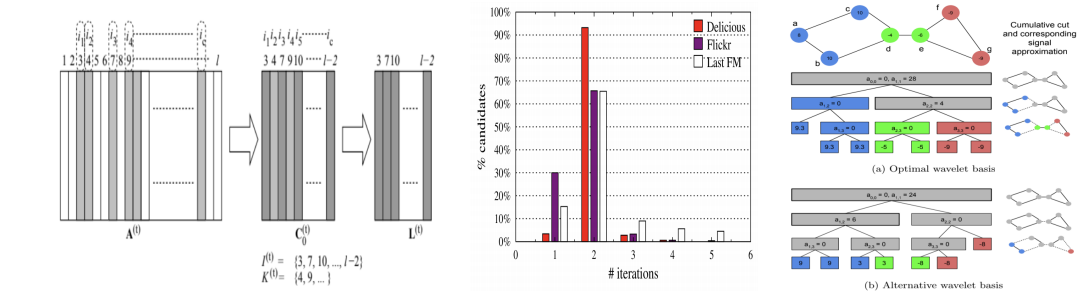
encode both the graph structure and the signal information. They created a powerful framework for modeling complex data, with wavelet trees that produce fast decaying coefficients to support a low-dimensional representation of graph signal.

Yang and Leskovec [YL15] studied community detection and identification through ground-truths. They defined ground-truth communities using nodes that explicitly state their group memberships in a set of 230 different networks.

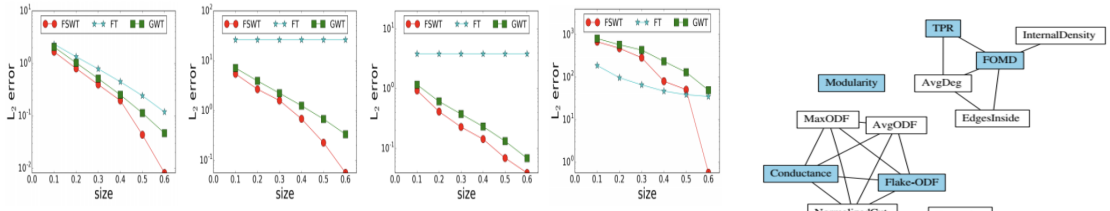
In another study, they along with Julian McAuley [YML14] developed a model to incorporate two major themes of community detection in complex networks: the detection algorithms that focus on the network structure, and the clustering algorithms consider node attributes.

Being able to detect and identify communities offers structure and knowledge of organizations, such the organization of food groups. Some problems in this task within network modeling involves model inefficiency, readability, and representation. A visual summary of the data and experiments of each study can be found in Figure 1.

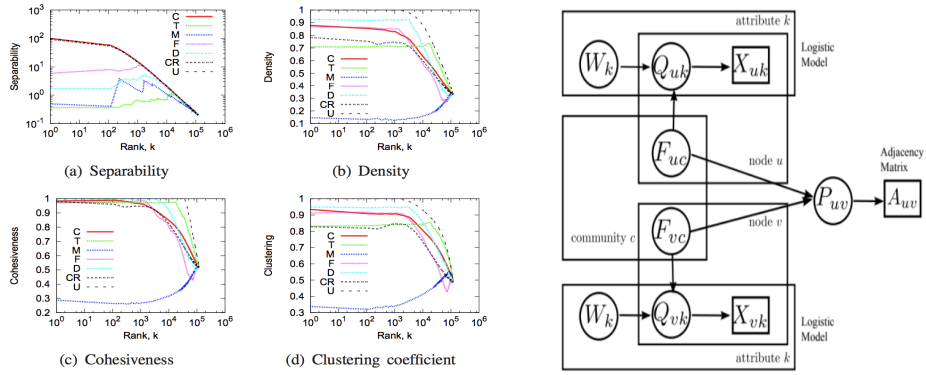
Tong et. al [TPS+08] proposed a Colibri methodology to do fast mining on large graphs, both static and dynamic. Their main goal was to answer how to find communities, anomalies, and detect patterns in large sparse graphs. For static graphs, the Colibri-S iteratively finds a non-redundant basis to preserve the sparsity of large graphs. The process is illustrated in Figure 1(a). The shaded columns are in the initial sample, while dark shaded columns are linearly independent throughout the process. The evaluation for Colibri used a network traffic dataset from the backbone router of a class-B university network, with the data turned into a binary matrix. Pit against similar algorithms, CUR [TfWAK03] and CMD [DKMM04], Colibri-S was



(a) (b) (c)



(d) (e)



(f) (g)

Dataset	N	E	C	K	S	A
Facebook	4,089	170,174	193	175	28,76	1.36
Google+	250,469	30,230,905	437	690	143.51	0.25
Twitter	125,120	2,248,406	3,140	33,569	15.54	0.39
Philosophers	1,218	5,972	1,220	5,770	6.86	6.87
Flickr	16,710	716,063	100,624	1,156	28.91	174.08

(h)

Figure 1: Graphical representations and visuals used in each study for static community detection. (a) Colibri-S process. (b) Iterations until convergence. (c) Two graph wavelet bases of a signal with a cut of size 4. (d) Compression results of the four datasets. (e) Clusters of community scoring correlations. (f) Average of goodness metrics for LiveJournal communities. (g) Illustration of the CESNA model (h) Network information from social network data.

proven to run significantly faster and more accurate.

Bonchi et. al [PBL14] proposed an algorithm that works with rich data on social

sites to effectively detect benign communities. Community detection in this instance means being able to find good communities that are associated with positive attributes per user information. **Models** The DCM method, short for Descriptive Community Mining, alternates between maximizing the community score and inducing a fitting concise description. Starting with a set of candidate communities, the first algorithm iterates through each one until convergence is found Here, attribute vectors are used to make to make informed guesses about the best vertex to add to the initial candidates with a distance function on the vertices attribute vectors to compute the distance to all neighbours of a candidate vertex, then form a pair of the candidate vertex with its nearest neighbor. The next algorithm, dubbed MAXIMUM_COMMUNITY_SCORE (MCS), considers only the structure of the graph, instead of the attribute data. After the algorithm constructs a list of all possible modifications, it chooses the one that maximizes the change in community score, which becomes applied only if it improves the score. Once the maximum score is found, the algorithm stops. **Results** The algorithms were evaluated using three datasets from different social networks. Figure 1(b) shows that most candidate communities have converged to a stable state after only two iterations, even though the maximum number of five iterations was imposed to ensure a stopping point. The community hillclimber and description induction method are robust, leading to stable solutions within a few iterations. The MCS routine is quick, can easily perform on largescale graphs, but the scalability of the description induction method depends much on the data at hand.

Silva et. al [SDB$^+$16] studied the problem sparse graph wavelet transform (SWT), which involves identifying a sequence of sparse graph cuts that leads to the minimum

error in the reconstruction of a given graph signal. The challenge here was being able to compute wavelet trees that encode both the graph structure and the signal information for graph wavelet transforms. They introduced an algorithm for computing SWT, via spectral graph theory. A fast graph wavelet transform (FSWT) was also created, using techniques such as Chebyshev Polynomials and the Power method, to improve the new algorithm's computational efficiency. **Models** The relationship between a wavelet tree and its graph structure is measured through using sparse cuts (those with a small number of edges). Figure **??** shows two candidate wavelet trees with a cut size of 4. The error of 1% of (a) in Figure 1(c) demonstrates a good basis, which generates sparse transforms, which will maximize the amount of energy from the signal that is conserved in a few coefficients. (b) gives a 22% error, which is alternative basis. The evaluation for the algorithms for computing sparse wavelet bases was applied using four datasets: Traffic (road networks from California), Human (gene network for Homo Sapiens), Wiki (sample of Wikipedia pages), and Blogs (a network of blogs with political leaning). The results can be seen in Figure 1(d), where the proposed approach (FSWT) outperforms the other two algorithms in most settings.

Node groups with highly concentrated edges provide a challenge with being able to identify communities of nodes, since there are many different definitions for a community, intractability of algorithms, issues with evaluation, and the lack of a reliable, fixed standard ground-truth (information provided by direct observation). Yang and Leskovec [YL15] proposed a method which allows to compare and quantitatively evaluate how different structural definitions of network communities correspond to

ground-truth communities. They considered over 200 datasets from social, collaboration, and information networks with defined ground-truth communities. **Models** Figure 1(e) shows clustered connections between scoring functions with correlation ≥ 0.6 (on the LiveJournal network). The scoring function is meant to characterize the community likeness of a connectivity structure for a set of nodes, so nodes with high scores can be considered to be parts of a community. They used four goodness metrics for community detection: *Separability* captures the intuition that good communities are well-separated from the rest of the network, i.e., they have relatively few edges pointing from set S to the rest of the network. *Density* builds on intuition that good communities are well connected. *Cohesiveness* shows the internal structure of the community. The *clustering coefficient* is based on the premise that network communities are manifestations of locally inhomogeneous distributions of edges, because pairs of nodes with common neighbors are more likely to be connected with each other. **Results** Figure 1(f) shows the results by plotting the cumulative running average of separability for LiveJournal ground-truth communities ranked by each of the six community scoring functions. "U" represents the upper bound (plots the cumulative running average of separability when ground-truth communities are ordered by decreasing separability). Each definition of network communities is appropriate for different networks. Conductance is the best scoring function for network containing well-separated and non-overlapping communities, and Triad Participation Ratio defines the most appropriate notion of a community when the network contains dense, heavily overlapping communities.

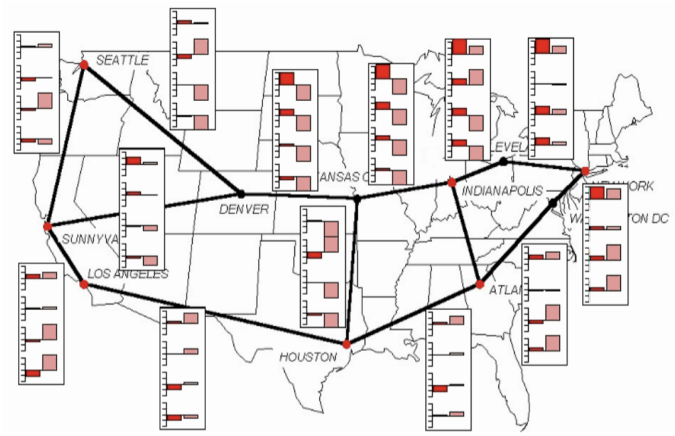
Network structure and the features and attributes of nodes are two possible sources

of information when it comes to community detection. Yang et. al [YML14] developed Communities from Edge Structure and Node Attributes (CESNA), an algorithm for detecting overlapping communities in networks with node attributes that considers both sources of information. CESNA uses statistics to model the interaction between the network structure and the node attributes. It has a linear runtime ($O(n)$) in the network size and is able to process networks of larger extent than similar approaches. This algorithm also finds relevant attributes from each community, which helps with the interpretation of detected communities by relevance. **Model** Figure 1(g) is an illustration of the CESNA model. X_{uk}: k-th attribute of node u; W_k: Logistic weight vector for attribute k; Q_{uk}: Probability that $X_{uk} = 1$; F_{uc}: Membership strength of node u to community c; A_{uv}: Indicator for whether the nodes u and v are connected; P_{uv}: Probability that $A_{uv} = 1$. The rectangles represent the node attributes and the observed network adjacency matrix. The circles show latent variables: community memberships F and logistic weights W. **Results** CESNA was evaluated through social, information, and content-sharing network data from Facebook, Google+, Twitter, Wikipedia (Philosophers), and Flickr. The datasets contained network information and node attributes, summarized in Figure 1(h). Key N: Number of nodes; E: Number of edges; C: Number of communities; K: Number of node attributes; S: Average community size; A: Community memberships per node.

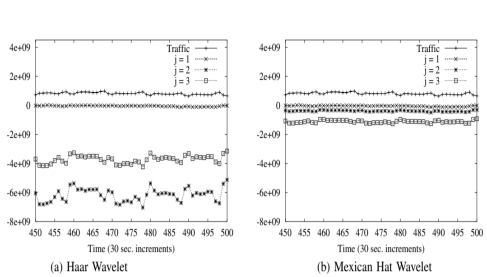
3.2 Anomaly detection

Anomaly detection is being able to find odd patterns, or outliers, in networks. For example, outliers can be abnormal traffic patterns [CK03], vandal users [PBL14] (in which the MCS returns negative values), etc. This task is useful in behavior detection and especially important in areas like security. Defining what "normal" behavior should be in relation to the data and ensuring security accuracy are a couple of challenges in anomaly detection. For example, Crovella and Kolaczyk [CK03] and Mohan et. al [DAM+14] used finding anomalies for tracking traffic congestion using graph wavelets. Their approach is through using spatial dependencies and signals as opposed to traditionally temporal methods.

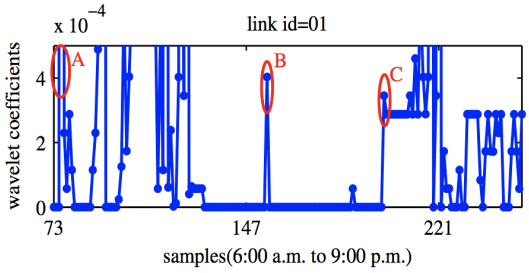
Spatial traffic analysis is the comparison and analysis of traffic patterns across multiple network links simultaneously, which can be used to find traffic patterns within topologically localized sets of links of a network, which is especially useful for traffic engineering. Crovella and Kolaczyk [CK03] approached the need of spatial traffic analysis with graph wavelets that generalize the traditional wavelet transform, namely the Haar and "Mexican hat" wavelets. This new framework can be applied to data elements connected via an arbitrary graph topology, able to simplify information and patterns in the data to treat network traffic data streams as a multivariate time series on collections of links. Their approach experiments with measurements from the Abilene network. Figure 2(a) shows two sets of wavelet coefficients for each link in the network. The left hand bars show coefficients before the service outage event, and the right hand bars show coefficients during the service outage event.



(a)



(b)



(c)

Figure 2: Graphical representations and visuals used in each study for anomaly detection in static networks. (a) View of wavelet coefficients. (b) Comparison of Haar and Mexican Hat Wavelets by the Atlanta-Houston Link. (c) Link-level graph wavelet analysis of East Coast Parkway road network.

Results The regions around the Atlanta-Houston link were used to consider how the choice of wavelet function ψ affects the properties of the transform. Figure 2(b) shows that Haar wavelet is useful for precise distinctions between rings, while the Mexican Hat wavelet is more gradual and better suited for finer study of the successive neighborhoods of a particular node.

The research by Mohan et. al [DAM$^+$14] focuses on the detection of anomalous traffic events, like congestion. They analyze wavelet coefficients, generated by graph wavelet operators applied to spatial signals, to extract information such as origin, propagation, and the span of traffic congestion are inferred. They reviewed theoret-

ical aspects of classical wavelet transform, such as weighted graph and graph signal, graph laplacian, graph fourier transform, and spectral graph wavelet transform. They apply the spectral graph wavelet analyses to road network data, consisting of several expressways and downtown roads in Singapore. The subnetwork G_1 contains road segments from East Coast Parkway, while the subnetwork G_2 contains road segments from Central Expressway. Through link-level graph wavelet analysis, the life span of events can be viewed in Figure 2(c), where anomalies of traffic A, B, and C are distinguished in red from the usual commute. A and C indicate the start of the morning and evening congestions respectively. B is another anomaly that occurred approximately around noon.

3.3 Entity linking

Being able to accurately link entities is a challenge; for example, when "Paris, France" is mentioned, it is not to be linked with the person, Paris Hilton. Some more examples of entity linking, provided by Airoldi et. al [ABFX09]: scientific literature connecting by citation, the Web connects pages by links, and protein-protein interactions. They developed models for relational data and accurate pairwise measurements. Their models display interaction matrices based on the multiple roles that objects exhibit in interaction with others, and the relationships between those roles.

Balasubramanyan and Cohen [BC11] expand on the previous model by jointly modeling links and text about the entities that are linked.

Handcock et. al[HRT07] use two estimation methods in their proposed model to make it efficiently represent transitivity, homophily by attributes, and clustering.

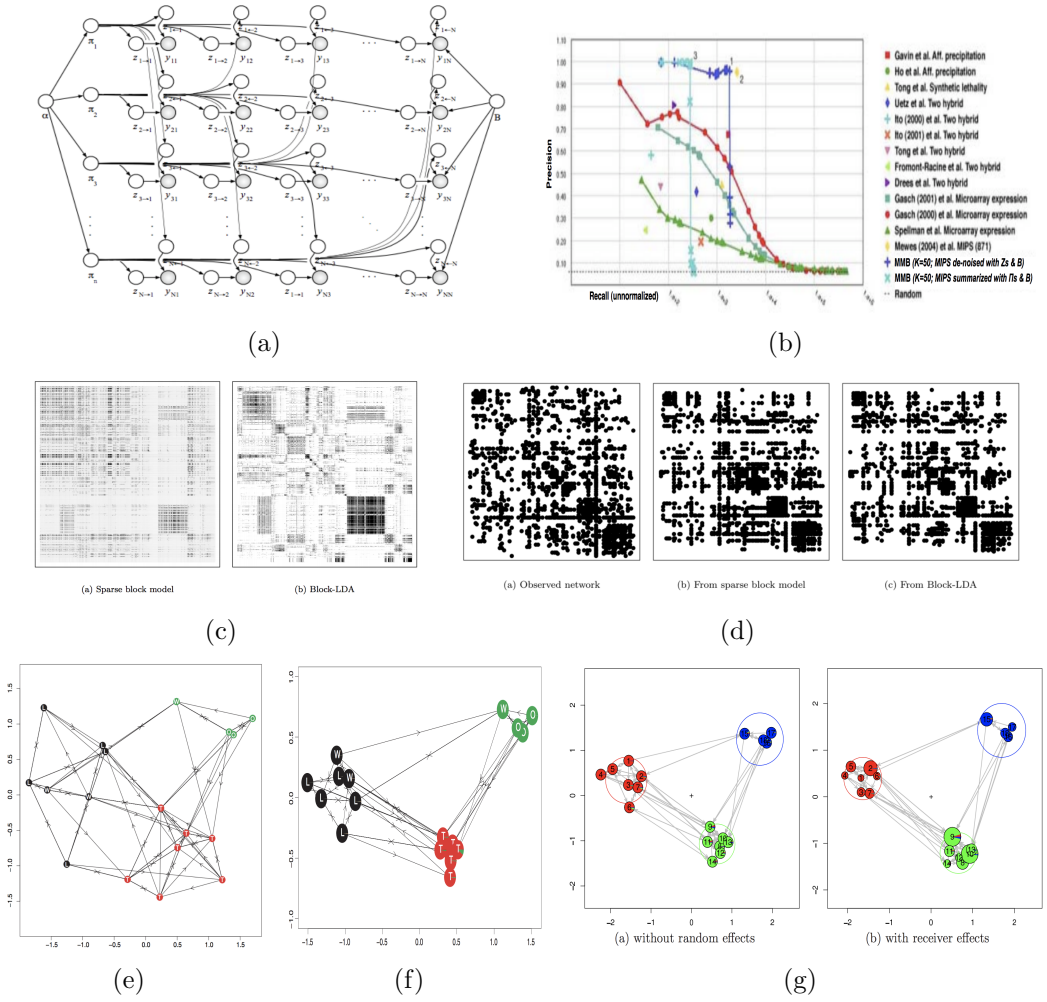


Figure 3: Graphical representations and visuals used in each study for static network entity linking. (a) Graphical model of the mixed membership blockmodel. (b) Protein-protein interaction evaluation of MMB versus related approaches. (c) Inferred protein-protein interactions; Block-LDA vs. Sparse graph. (d) Enron network and the de-noised retreived versions. (e) Monks' links with the maximum likelihood latent space positions. (f) Monks' links monks with the Bayesian estimation. (g) Monks in a monastery with the Latent Cluster Random Effects Model.

Airoldi et. al [ABFX09] created a new class of variable models, called mixed membership stochastic blockmodels (MMB), for improving the efficiency of pairwise measurements. Stochastic blockmodels provide group structuring by fixing paired measurements into groups and connecting those pairs of the groups, but limit each unit to instantiate the connection patterns of a single group. Mixed membership

structures associate each unit with several groups instead of just one, able to capture multiple roles. **Model** Figure 3(a) represents MMB and its process for a graph $G = (N, Y)$, with all the pairwise measurements $Y(p, q)$ as an element of 0, 1 of blockmodel B, a directed graph. In the figure, latent node distributions are drawn from i for node i, where $\pi i, g$ denotes the probability of node i belonging to group g. The matrix of Bernoulli rates $B(K \times K)$ defines the probabilities of interactions with different groups. The probability of having a connection from a node in group g to a node in group h is represented by $B(g, h)$. The indicator vector $zp \rightarrow q$ denotes the specific block membership of node p when it connects to node q, while $zp \leftarrow q$ denotes the specific block membership of node q when it is connected from node p. **Result** The MMB was evaluated using a yeast genome database of physical protein interactions, by the Munich Institute of Protein Sequencing (MIPS), to summarize and de-noise the complex connectivity patterns quantitatively. The performance is shown in Figure 3(b). The results show that the MMB (light and dark blue lines) successfully reduces the dimensionality of the data, while discovering information about the multiple functionality of proteins (yellow diamonds) that can be used to inform follow-up analyses.

Balasubramanyan and Cohen [BC11] approach the problem of latent group identification through pairwise measurements with the model Block-LDA, which combines the mixed membership stochastic blockmodels [ABFX09] with topic models. It is meant to improve entity-entity link modeling by jointly modeling links and text about the entities that are linked. This model allows for information sharing between links of entity pairs and text documents through dormant topics. **Model and Result**

The model enables the sharing of information between both blocks through shared latent topics. Using a dataset of protein interactions from MIPS, Figure 3(c) shows a posterior likelihood comparison of protein-protein interaction matrices using the sparse blockmodel and the proposed Block-LDA model. Figure 3(d) shows the Enron dataset in (a) its original state, (b) the de-noised version using a sparse blockmodel, and (c) the de-noised version from using the Block-LDA.

Handcock et. al [HRT07] proposed the latent position cluster model, where the probability of a tie between two interacting units depends on the distance between them. **Model** The cluster model represents network data through transitivity (the chance a third node will be connected to other groups if that node relates to either one of them), homophily (bonding with similar actors through relatable characteristics), and clustering simultaneously, and does not require the number of clusters to be known. They proposed two estimation methods for the model. The first computes the maximum likelihood estimator of the (non-clustering) latent space model, and then computes the maximum likelihood estimator for the mixture model applied to the resulting estimated latent positions. **Model and Result** The data experimented with was the "monks in a monastery" collection, labeled with T (Turks), L (Loyal Opposition), O (Outcasts) and W (Waverers), and connected by right arrows (ties). Figure 3(e) represents the maximum likelihood latent space positions from the first stage of a proposed two-stage maximum likelihood method. This method first computes the maximum likelihood estimator of the (non-clustering) latent space model, and then computes the maximum likelihood estimator for the mixture model applied to the resulting estimated latent positions. "We say that a monk has the social rela-

tion of like to another monk if he ranked that monk in the top three monks for positive affect in any of three interviews given over a 12-month period." The second method is fully Bayesian and uses Markov chain Monte Carlo sampling. It estimated the latent positions and the clustering model simultaneously. The estimation is illustrated using the monk data in Figure 3(f). The Bayesian estimate method appears to produce a greater distinction between the groups than the previous two-stage estimate method.

Krivitsky et. al [KHRH09] sought to represent the same relations as [HRT07], plus the heterogeneity of actors in social network data. Their model, the Latent Cluster Random Effects Model, supports those four features through adding random sending and receiving effects to the latent position cluster model of [HRT07]. Their Bayesian approach estimates the latent positions, the clustering model, and the actor-specific effects simultaneously. Using the MCMC algorithm, it iterates over the and parameters shown and in turn updates the variables, and block-updates those which are expected to be correlated. **Results** The Latent Cluster Random Effects Model was evaluated using the isolated American monastery data, shown in Figure 3(g). Panel (a) shows a similar performance to the model proposed in [HRT07], as a latent cluster model without random effects. Notice the contrast of Monk 1 (red node) in both graphs, seeming less than popular among the groups, due to the unbalanced out-ties to in-ties of other members. Without receiver effects, he is pushed to the edge of the L group. With receiver effects, the monk moves toward the center of the Loyal Opposition group because of his out-ties, along with a small receiver effect to compensate. The receiver effect absorbs his unpopularity and the model gives him a more informative position in the social space.

3.4 Event detection

Event detection in this survey is synonymous with pattern recognition. It was exercised in [PBL14], where the community detection involves finding benign groups thus examining patterns of behavior. [DAM+14] also incorporated this task within the focus of finding traffic anomalies.

Hammond et. al [HVG09] provided a flexible model, able to be used for just about any domain. Their approach uses the connectivity information encoded in the edge weights of weighted graphs, not reliant on other attributes of the vertices (like their positions), as other approaches would be.

Hammond et. al [HVG09] proposed a method to construct a spectral graph wavelet transform (SGWT) of functions defined on the vertices of an arbitrary finite weighted graph, for the task of pattern recognition. **Model** Figure 4(a) represents the scaling function of the SGWT. "The scaling functions help ensure stable recovery of the original signal f from the wavelet coefficients when the scale parameter t is sampled at a discrete number of values t_j." Function key Scaling function $h(\lambda)$ (blue curve), wavelet generating kernels $g(t_j\lambda)$, and sum of squares G (black curve), for $J = 5$ scales, λmax $= 10$, $K = 20$. The Swiss roll dataset is used to demonstrate building wavelets in a point cloud domain, shown in Figure 4(b). The spectral graph wavelets at four different scales localized at the same location. (a) vertex at which wavelets are centered (b) scaling function (c)-(f) wavelets, scales 1-4. $\sigma = 0.1$ was used for computing the underlying weighted graph, and $J = 4$ scales with $K = 20$ for computing the spectral graph wavelets. The figure shows how "spectral graph wavelets

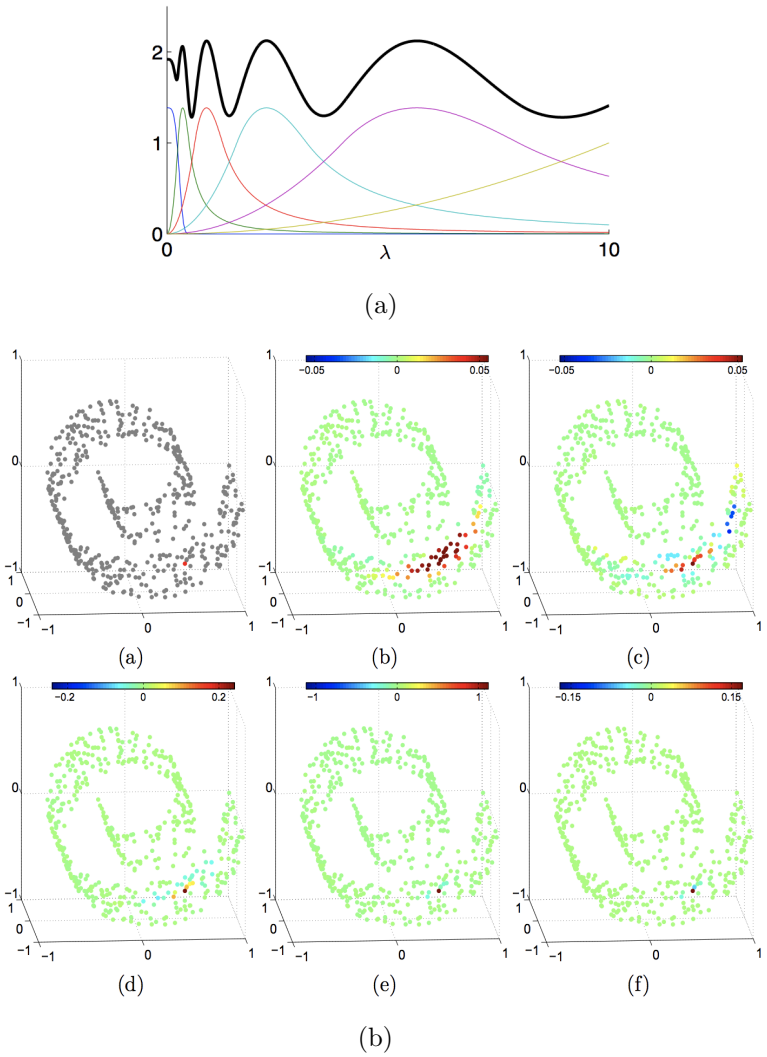


Figure 4: Graphical representations and visuals used in the study for event detection and pattern recognition in a static network. (a) Scaling function of the SGWT. (b) Spectral graph wavelets on Swiss Roll data.

can adapt to the underlying manifold structure of the data in an implicit way. The support of the coarse scale wavelets diffuse locally along the manifold, and do not 'jump' to the upper portion of the roll."

3.5 Other tasks in a static network

The next papers for static networks do not fit into the other tasks, but they are important extensions for high-dimensional data analysis to networks. For example,

Gavish et. al [GNC10] introduced a harmonic analysis (a branch of mathematics used for the representation of functions or signals as the superposition of basic waves) approach for playing a bigger role in problems involving complex data encoded as graphs or networks, processing data on graphs.

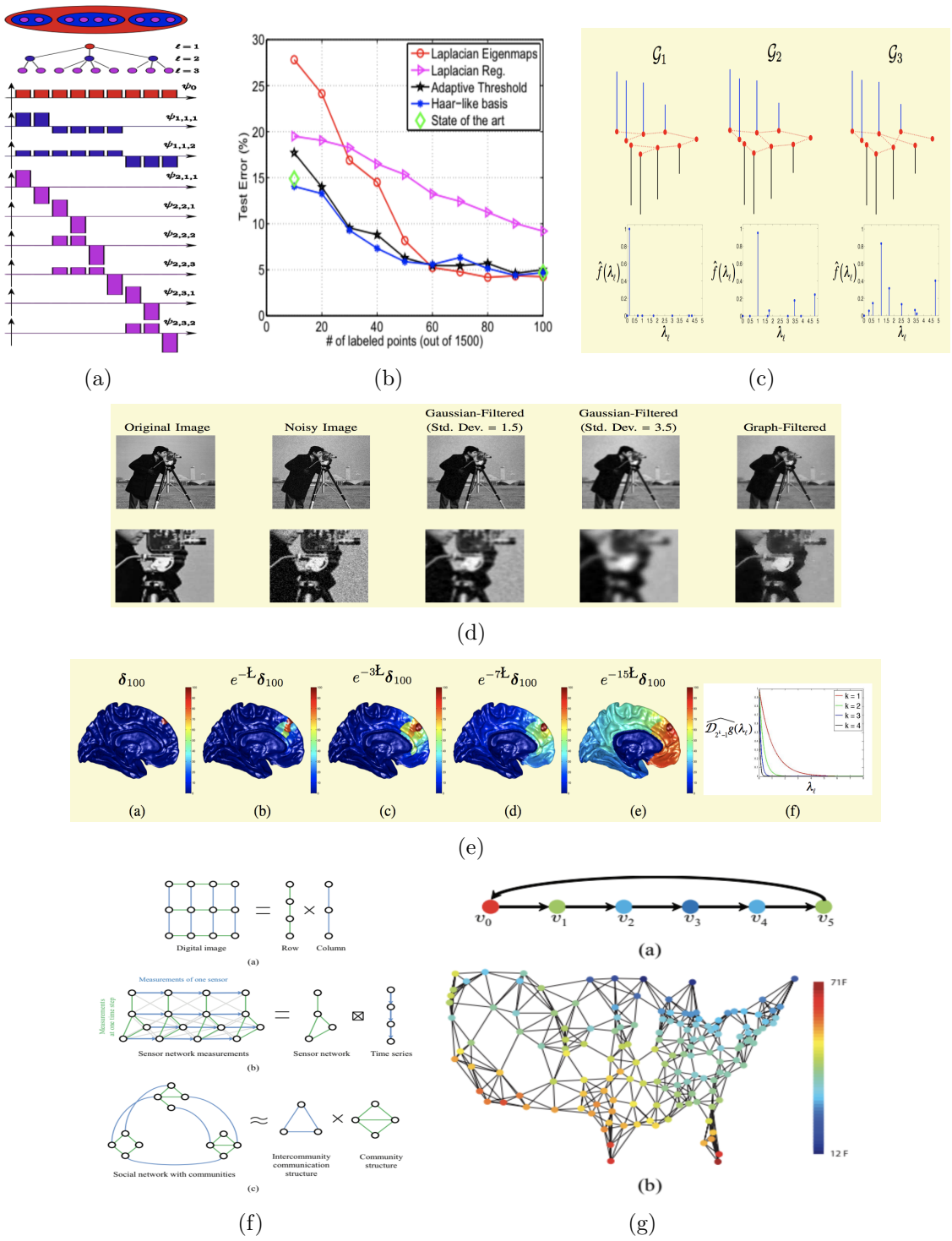
Shuman et. al [SNF+13] include the previous approach [GNC10] in their tutorial overview of signal processing on graphs with high-dimensional data.

Sandryhaila and Moura [SM14] further used signal processing methods to develop a method to analyze and process very large datasets, or Big Data. Their model is appropriate for both static and dynamic networks.

Gavish et. al [GNC10] developed a harmonic analysis framework to apply to the challenges of the analysis high dimensional data or data encoded as graphs or networks. **Model** A data adaptive Haar-like wavelet base was built for functions over the dataset, illustrated in Figure 5(a). **Result** The dataset in the USPS benchmark contained 1500 grey scale 16x16 images of the digits 0 to 9. The task was to distinguish the digits 2, 5 from the rest. Figure 5(b) shows that the Haar-like classifier dominates the Laplacian Eigenfunction when labeled points are few and is comparable when they are many.

Shuman et. al [SNF+13] offered a signal processing tutorial overview of the analysis of data on graphs. In their study, applications common data processing tasks are applied, such as filtering, denoising, inpainting, and compressing graph signals. They address such challenges:

1. deciding how to construct a weighted graph that captures the geometric struc-



(a) (b) (c)

(d)

(e)

(f) (g)

Figure 5: Graphical representations and visuals used in each study of various unique tasks fulfilled for static networks. (a) Haar-like basis. (b) Results on the USPS benchmark. (c) Importance of the underlying graph (d) Filtering an image using Tikhonov regularization. (e) Diffusion operators and dilution. (f) Examples of product graphs with various data. (g) Examples of graph signals on product graphs.

ture of the underlying data domain, in cases where the graph is not directly dictated by the application

2. incorporating the graph structure into localized transform methods

3. simultaneously, leveraging intuitions developed from years of signal processing research on Euclidean domains

4. developing computationally efficient implementations of the localized transforms to extract information from high-dimensional data on graphs and other irregular data domains

First they went over different ways to encode graph structure and define graph spectral domains. Figure 5(c) demonstrates how both the smoothness and the graph spectral content of a graph signal depend on the underlying graph. The signal $\mathbf{f}$ is plotted on three different unweighted graphs with the same set of vertices, but different edges. The top row shows the signal in the vertex domains, and the bottom row shows the signal in the respective graph spectral domains. The smoothness and graph spectral content of the signal both depend on the underlying graph structure. Next, they discussed different ways to generalize downsampling, filtering, dilation, and other fundamental operations to the graph setting. In Figure 5(d), the bottom row of images shows close-ups for detailed looks at the top row of images. Comparing the results of the Gaussian and Graph-filtered filtering methods, the Gaussian filter also smooths across the image edges, which sufficiently smooths the smoother areas of the image. Then, they provided examples of graph transform designs. These designs can be categorized into vertex domain designs and graph spectral domain designs.

Figure 5(e) shows the application of different powers of the heat diffusion operator can be interpreted as graph spectral filtering with a dilated kernel. The localized basis functions at each resolution level are downsampled and then orthogonalized through a variation of the Gram-Schmidt orthogonalization scheme.

Sandryhaila and Moura [SM14] used a paradigm based on discrete signal processing on graphs (DSP_G) to approach the challenge of analyzing and processing Big Data. The implementation is based on parallelization and vectorization. Product graphs offer versatile graph models to represent complex datasets. **Models** Figure 5(f) gives three examples of product graphs. 5(f)-(a) shows digital images reside on rectangular lattices that are Cartesian products of line graphs for rows and columns. 5(f)-(b) is a dynamic example, to be described in Section 4. 5(f)-(c) is a social network with three similar communities is approximated by a Cartesian product of the community structure graph with the intercommunity communication graph. **Results** These three graphs can also support graph signals, shown in Figure 5(g), which come from website features (topic, view count, relevance) and social networks, where characteristics of individuals (opinions, preferences, demographics) are connected. 5(g)-(a) is dynamic, and will be described in Section 4. The test used a dataset containing daily temperature measurements collected by 150 weather stations across the United States, represented by Figure 5(g). 5(g)-(b) was constructed by connecting each sensor to 8 of its nearest neighbors with undirected edges with weights. It shows measurements from a single day as well as the sensor network graph.

CHAPTER 4: DYNAMIC NETWORKS

In the language of networks, one way dynamics components can be translated is into the birth and death of edges and nodes. For example, in an academic network, say, a university, new nodes or students may be introduced at any time and old nodes or graduates may drop out due to inactivity; links of friendships and alliances may be even more brittle. Over time, there has been a shift of interest to dynamic graphs, which reflects on the face that real-world networks are continuously going through change.

4.1 Community evolution detection

In dynamic networks, the task of community detection concentrates on its evolution through time. The main purpose is to find what behaviors within a network have changed changed, which entities died or came to exist, etc. A visual summary of the research papers of this task is presented in Figure 6

Lin et. al [LCZ+08] offered a unified process for analyzing communities and their evolutions through their proposed model, which differs from the deviates from the traditional two-step approach, wherein communities are first detected for each time slice, and are then compared to determine correspondences.

Some of the papers propose new models that incorporate evolutionary community detection into heterogeneous networks, which "involve more than one type of actors

and multiple heterogeneous interactions between different types of actors [TLZN08]." Tang et. al [TLZN08] presented a framework to find community evolution for dynamic multi-mode networks where several different types of interaction may occur around multiple heterogeneous nodes.

Similarly, Sun et. al [STH+10] studied the problem of evolutionary community detection in heterogeneous networks and created a model to give structure to the generation of multi-typed communities, or net clusters.

The main feature in the dynamic side of the sparse-preserving model from aforementioned [TPS+08] is that it can support such temporal smoothness to quickly while maintaining the sparseness of the low rank approximation.

Yang et. al [YCZ+11] implemented a Bayesian approach in their model to capture uncertain parameter values, which made this proposal more robust to data noise than the point estimation approach. An algorithm for the Bayesian inference improved it to handle large sparse networks.

Lin et. al [LCZ+08] address the challenge of dynamic network community evolution analysis with their novel framework, FacetNet. Using FacetNet, communities generate evolutions and are regularized by the temporal smoothness of evolutions. This framework can discover communities that jointly maximize the fit to the observed data and temporal evolution. The approach is from formulating the problem in terms of non-negative matrix factorization, where communities and their evolutions are factorized in a unified way. To regularize the community structure at a given time, a cost function was introduced to measure the quality of community structure at the time. The cost consists of a snapshot cost and a temporal cost: $cost = \alpha \cdot \mathcal{CS}$

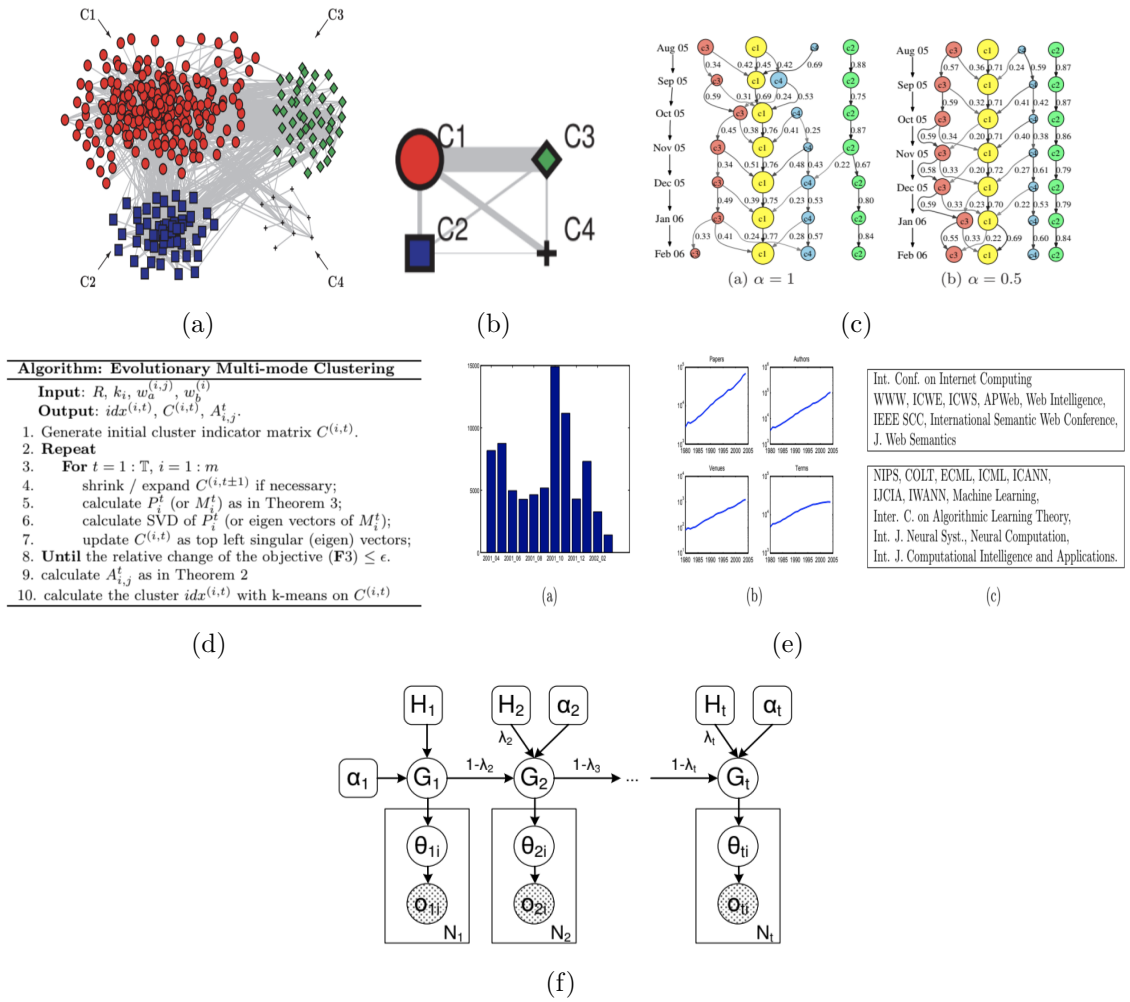
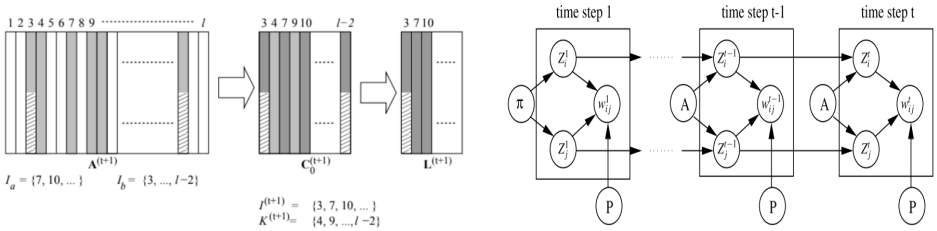


Figure 6: Graphical representations and visuals used in each study for dynamic community evolution detection (Part 1). (a) Four communities in the NEC dataset. (b) The Community Net of the NEC dataset. (c) The Evolution Net of the NEC dataset. (d) Description of the Evolutionary Multi-mode Clustering alg.. (e) Enron and DBLP data evaluated with the Evolutionary Multi-mode Clustering alg.. (f) Graphical Model for Evo-NetClus.

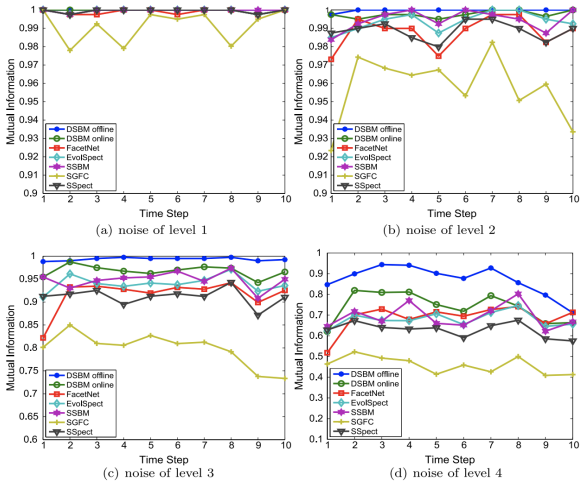
$+ (1 - \alpha) \cdot \mathcal{CT}$ The snapshot cost $\mathcal{CS}$ measures how well a community structure fits the observed interactions at a certain time. The temporal cost $\mathcal{CT}$ measures how consistent the community structure is with respect to historic community structure, at a time $t - 1$. The user sets the parameter α to control the level of emphasis on each part of the total cost. They introduced two concepts to represent community

Year	Training Type	Testing Type	Test Size 10% (cluster number K)	Test Size 20% (cluster number K)
1992	Term	Term	1.600 (4)	1.390 (4)
1992	Term+Author	Term+Author	2.205 (8)	1.697 (6)
1992	Term+Author+Conf.	Term+Author	2.434 (8)	2.095 (8)
1992|1991	Term+Author+Conf.	Term+Author	**2.8365** (8)	**2.671** (8)

(a)



(b) (c)



(d)

Figure 7: Graphical representations and visuals used in each study for dynamic community evolution detection (Part 2). (a) Comparison of compactness between Evo-NetClus and similar models. (b) Illustration of Colibri-D process. (c) Graphical representation of the Dynamic Stochastic blockmodel (DSBM) (d) Performance of DSBM against similar approaches.

structures and their evolutions: Community Net and Evolution Net. They use blog data collected by an NEC in-house blog crawler to test the performance FacetNet. In Figure 6(a), a graph was drawn of four communities from the thousands of entries. Figures 6(b) and 6(c) show the Community and Evolution Nets, representing the communities appropriately.

Identifying community evolution in dynamic multi-mode networks is a challenge

as both actor membership and interactions evolve. Tang et. al [TLZN08] addressed this challenge with the Evolutionary Multi-mode Clustering algorithm, which adopts a spectral clustering framework, to identify community evolution in dynamic multi-mode networks. Multi-mode networks could assist in making online marketing accurate with limited user information, by employing the relationship between different types of objects. **Model** Figure 6(d) describes the new model's algorithm in which R_{ij}^t: interaction between two modes of actors at t timestamp; R_{ij}: the interaction between two modes of actors; k_i: number of latent communities; $w_a^{(i,j)}$: weights associated with different interactions; $w_b^{(i)}$: the trade-off between the loss of interaction approximation and temporal regularization; $C^{(i,t)}$: latent cluster membership; A_{ij}^t: group interaction matrix. Here are Theorems 2 and 3 for reference: "THEOREM 2. If $C^{(i,t)}$ are given, the optimal group inter-action matrix $A_{i,j}^t$ can be calculated as $A_{i,j}^t = (C^{(i,t)})^T\ R_{i,j}^t C^{(j,t)}$," and "THEOREM 3. Given $C^{j,t}$ and $C^{(i,t\pm1)}$, $C^{(i,t)}$ can be computed as the top left singular vectors of the matrix P_i^t concatenated by the following matrices in column-wise: $[\{\sqrt{w_a^{(i,j)}}\ R_{i,j}^t C^{(j,t)}\}_{i<j},\ \{\sqrt{w_a^{(k,j)}}\ R_{k,j}^t C^{(k,t)}\}_{k<i},\ \{\sqrt{w_b^{(i)}}\ C^{(i,t\pm1)}\}]$." **Results** After proving that the proposed method would perform better than other similar ones using synthetic data, they experimented with Enron and DBLP data. Figure 6(e)-(a) shows Emails sent each month on Enron data, where the heavy amount of emails shown over those two months give so much data that it is not necessary to consider any temporal information, so the performance for them isn't as high as the other months. Figure 6(e)-(b) displays the number of papers, active authors, venues, and terms in each year using DBLP data. The blocks of Figure 6(e)-(c) show two examples of venue community that were extracted with the

proposed method over the year 2004 in DLBP, and they associate appropriately with the conference and journal topics.

Sun et. al [STH+10] studied the problem of multi-typed evolutionary community detection in a heterogeneous network. They proposed a Dirichlet Process Mixture Model-based generative model to model the community generations with a clustering of communities to explain the current and historical networks are automatically detected. In a Mixture model it is typically difficult for people to specify the correct cluster number. The Dirichlet Process Mixture (DPM) Model is a common way to solve the problem, where the cluster number is considered as countable infinite. The proposed model is an extension of the DPM Model: the Evo-NetClus, for evolutionary net-clusters. **Model** Figure 6(f) displays a graphical model for the Evo-NetClus model. At each time t a DPM is built for the target objects ($o_{i,t}$) in the network G_t, and $\theta_{i,t}$ is a parameter of the cluster associated with ($o_{i,t}$), following the distribution of G_t. H_t is an expectation distribution of the Dirichlet distribution, and α is the concentration parameter. **Results** Along with considering different types of objects, the Evo-NetClus's clustering results can carry historical information. The algorithm experiments against with three other degenerated clustering models using DBLP datasets, those fewer types of objects or not using historical priors, to compare similarity compactness. The results are shown in Figure 7(a), where the proposed Evo-NetClus model performed the best through considering both multiple types of objects and historical impact information, in regards to providing a better similarity feature.

Continuing on the Colibri study [TPS+08], Colibri-D is an updated Colibri-S algo-

rithm for dynamic, time-evolving graphs. It is more accurate than the static version, saving even more space and time in computations. Figure 7(b) displays the process illustrations for the dynamic algorithm. **Model and Result** The space cost for Colibri-D is not presented, as it has the same space cost as Colibri-S, so only the running time performance for dynamic graphs was tested. It exceeded both the competing algorithm and the proposed Colibri-S.

Yang et. al [YCZ+11] offered a dynamic stochastic blockmodel (SBM) for finding communities and their evolution in a dynamic social network. The proposed DSBM has two versions: 1) the online inference version progressively updates the probabilistic model over time; 2) the offline inference version retrospectively learns the probabilistic model with network data obtained at all time steps. This DSBM model adopts the Bayesian approach, which computes the posterior distributions for unknown parameters. This can predict community memberships and derive important characteristics of communities, such as structures and evolution. **Model** Figure 7(c) shows the graphical representation of the DSBM, where A is a transition matrix meant to capture the dynamic evolution of communities; a pair of nodes i and j; Z^t: community memberships; w_{ij}: link weight. **Results** The performance of the online and offline versions were compared to other methods for dynamic community analysis, which can be viewed in Figure 7(d). The data was synthetic, containing 128 nodes that belong to 4 communities, with 32 nodes in each community. The generated datasets with four different noise levels. As the charts show, the proposed DSBM algorithms have better accuracy than the others for all datasets. Also, the evolutionary algorithms (including [LCZ+08]'s FacetNet and [TLZN08]'s evolutionary

spectral clustering algorithm) are more accurate than the static versions in most cases, showing the advantages of the dynamic models in capturing community evolution in dynamic social networks.

4.2 Anomaly detection

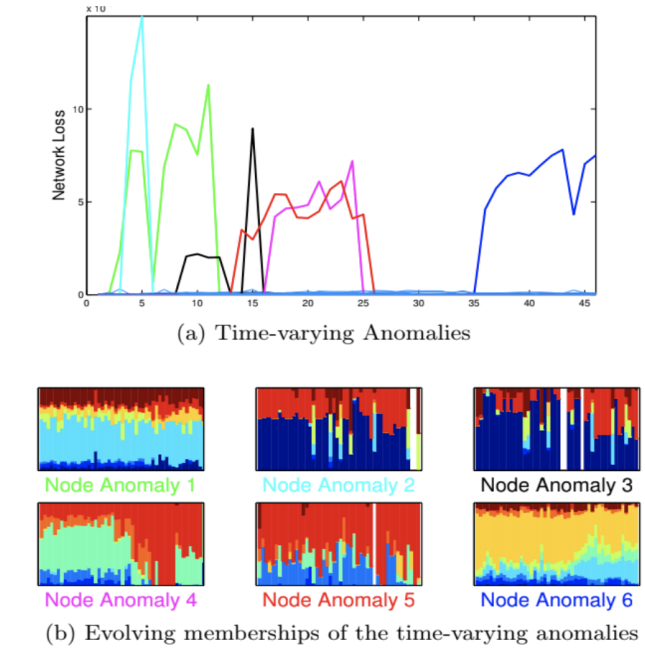
Using dynamic graphs, several new anomalies can be viewed over a temporal domain, such as tracking natural disasters, community activity, and economic downfalls.

Rossi et. al [RGNH13] create a model that outperforms the pre-existing dynamic mixed-membership stochastic blockmodels [FSX09] in a scalable, descriptive model that pays more attention to anomaly detection by capturing the roles of nodes and how they evolve.

Han et. al [HXA15] also create a new model based on the stochastic blockmodel (SBM) to be a foundation for application settings like dynamic and multi-layer networks.

Rossi et. al [RGNH13] proposed a dynamic behavioral mixed-membership model (DBMM) for large networks, with the task of capturing the roles of nodes and how they evolve. This DBMM identifies patterns and trends of nodes and network states based on temporal behavior, predicts future structural changes, and detects unusual temporal behavior transitions. It is possible to learn how the behavior of a network changes over time, given a sequence of dynamic behaviors $G = G_t : t = 1, ..., t_{max}$.

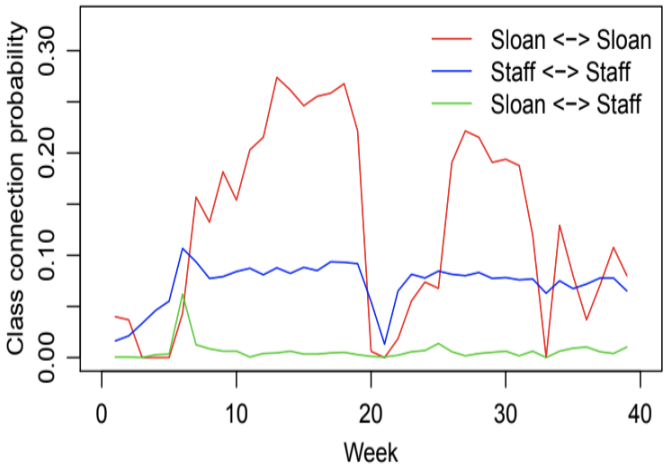
Model For the purpose of categorization, this particular study will be concentrated on anomaly detection, as they create an algorithm for network membership prediction. "The anomaly score is the difference between the predicted network mixed-



(a) Time-varying Anomalies

(b) Evolving memberships of the time-varying anomalies

(a)



(b)

Figure 8: Graphical representations and visuals used in each study for anomaly detection in dynamic networks. (a) Time-varying anomaly detection and evolution from Enron. (b) Connection probability estimation of a business school.

memberships and the ground-truth mixed-memberships. Therefore, the score represents the divergence of that nodes transitions from the entire network." **Results** Figure 8(a) shows nodes that were anomalous for very short periods of time in an Enron email network and their behavior over such time. Such anomalies would be im-

possible to detect in a static network, as their brief existence would be overshadowed by typical regular behavior.

Han et. al [HXA15] explored the multi-graph stochastic blockmodel for theoretical analysis, which is a foundation for application settings, such as dynamic and multi-layer networks. A multi-graph is a collection of networks over a common set of nodes, often worked into dynamic networks of time-evolving edges like time-stamped interactions between people, such as phone calls, text messages, and e-mails. The challenge with multi-graphs is being able to "extract common information across the layers of the multi-graph in a concise representation, yet be flexible enough to allow differences across layers." **Model** Implemented into the stochastic blockmodel, the multi-graph SBM splits nodes into classes that define blocks in the multi-graph. Among the consistency of two estimators for the multi-graph SBM (spectral clustering and the maximum-likelihood estimate), a variational approximation to the maximum-likelihood estimate (MLE) to work with large networks was proposed. At the time of the study, the MLE was computationally unfit for large networks. Variational approximation replaces the joint distribution with independent marginal distributions to approximate the MLE. **Results** The multi-graph SBM was evaluated using the MIT Reality Mining data set. It consists of 93 total students and staff at a business school over the course of 40 weeks (the 2004-2005 school year) where the values were represented by recorded cell phone proceedings. "Two communities were found: one containing 26 Sloan business school students, and one containing 67 staff working in the same building." In Figure 8(b), the profile MLE was used to estimate connection probabilities of the two communities. The over time varying probabilities proved the

importance of the varying class connection probability assumption.

4.3 Entity linking

"Clustering is an important data mining task for exploration and visualization of different data types like news stories, scientific publications, weblogs, etc. Due to the evolving nature of these data, evolutionary clustering has recently emerged to cope with the challenges of mining temporally smooth clusters over time" [AX08]. Entity linking is similar to clustering in the sense that related attributes are being connected for visual analysis. The figures of each study in entity linking has been compiled in Figures 9 and 10.

Ahmed and Xing [AX08] brought up typical entity linking examples in the beginning of their report. They introduced a framework for modeling complex longitudinal data, which tracks the same sample at different points in time. Their approaches differed from most other evolutionary clustering approaches so that their proposed model would not become orthogonal and highly interdependent.

Fu et. al [FSX09] presented a model for entity linking in evolving networks builds that was built on time the mixed-membership stochastic blockmodel [ABFX09].

Kim and Leskovec [KL13] created a model to handle evolutionary relational data. Their approach considers interactions between groups of nodes that evolve with time as well as single node arrival and departure dynamcis.

Hanneke et. al [HX07] presented a family of statistical models for social network evolution to answer the need of entities linked by relations in social network analysis.

Oselio et. al [OKI13] created a model for the mining of connections in multi-layer

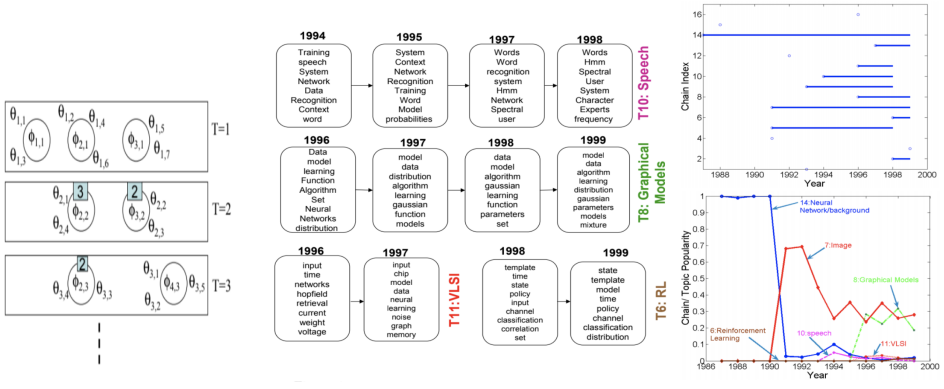
networks. It uses techniques from Bayesian Model Averaging, so the layers of the network become conditionally decoupled using a latent selection variable.

Xu and Hero [XI14] also extended the classic stochastic blockmodel (SBM) into a dynamic setting to predict dynamic links and examine temporal dynamics in such networks.

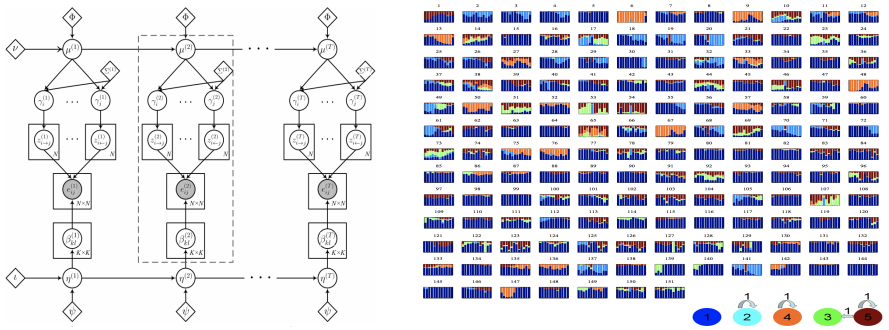
Xu [Xu14] proposed another dynamic SBM that allows the presence or absence of edges to directly influence future edge probabilities while retaining the explicability of the stochastic blockmodel.

Ahmed and Xing [AX08] introduced the temporal Dirichlet process mixture model (TDPM) as a framework for modeling complex longitudinal data. The recurrent Chinese restaurant process (RCRP) is used as a construction for the TDPM model. **Model** The RCRP in Figure 9(a) is analogous to seating customers at tables in a Chinese restaurant and operates in epochs, such as days. **Results** The TDPM was tested on real data by building a simple non-parametric dynamic clustering-topic model, and applying it to analyze the NIPS12 document collection. Figure 9(b) shows topic durations (illustrating the births and deaths of various topics), the popularity of certain topics within the document collection, and popular keywords of those topics as they evolve over time(capturing topic evolutions).

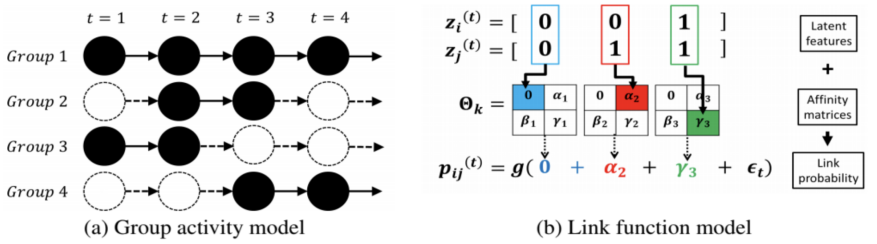
Fu et. al [FSX09] tackled the issue of themes concerning the functions of actors and their relations to each other being lost over a temporal process. As these themes are dynamic, a dynamic mixed membership stochastic blockmodel (dMMSB) was proposed. **Model** This model tracks across time for the evolving roles of the actors. It dissects the evolving functional composition of the actors based on their dynamic



(a) (b)



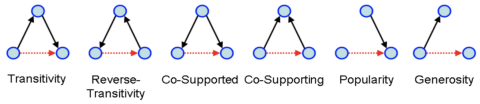
(c) (d)



(e)

Model	NIPS			DBLP			INFOCOM		
	TestLL	AUC	F1	TestLL	AUC	F1	TestLL	AUC	F1
Naive	-2030	0.808	0.177	-12051	0.814	0.300	-17821	0.677	0.252
LFRM	-880	0.777	0.195	-3783	0.784	0.146	-8689	0.946	0.703
DRIFT	-758	0.866	0.296	-3108	0.916	0.421	-6654	0.973	0.757
DMMG	**−624**	**0.916**	**0.434**	**−2684**	**0.939**	**0.492**	**−6422**	**0.976**	**0.764**

(f)



(g)

Figure 9: Graphical representations and visuals used in each study for dynamic entity linking (Part 1). (a) Recurrent Chinese restaurant process as a construction for the TDPM. (b) Results of the TDPM on the NIPS12 dataset. (c) Graphical representation of the dynamic mixed membership stochastic blockmodel. (d) Temporal changes in 150 mixed membership vectors for each actor. (e) The births and deaths of groups; Link functional model. (f) Missing link prediction performance for the Dynamic Multi-group Membership Graph Model. (g) Graphical representation of 3-node statistics.

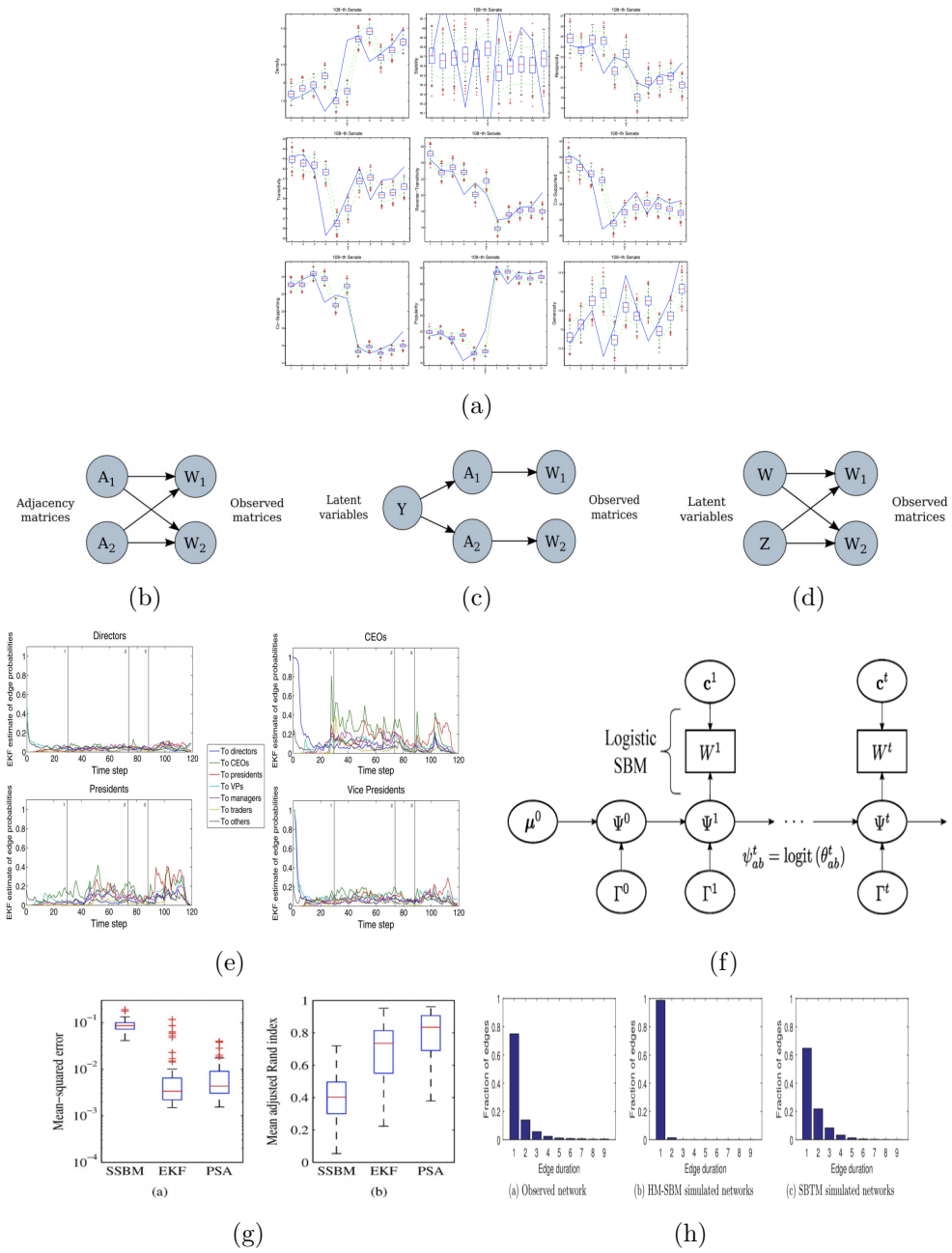


Figure 10: Graphical representations and visuals used in each study for dynamic entity linking (Part 2). (a) Comparison of statistic values from real and sampled networks. (b) Adjacency and Observed matrices. (c) Latent variables distributing from adjacent to observed matrices. (d) Latent variables described as similarity matrix and selection variable. (e) Combining the two layers of the Enron network with a parameter $\alpha = 0.5$. (f) Graphical representation of a temporal extension of the SBM. (g) Temporal extension's performance evaluation of mean-squared tracking error and and class estimation accuracy. (h) Histograms of edge durations in a Facebook dataset.

interactions. Figure 9(c) shows a graphical representation of the proposed dMMSB, where the dotted lines indicated a normal mixed membership stochastic blockmodel. The dMMSB was used to study Enron email communication networks. **Results** Figure 9(d) displays each member's role composition and the role compatibility matrix. Role 1 (blue) is inactive. Role 2 (cyan) actors only send email to persons of the same role, thus forming a cliqueand the same thing happens with Role 4 (orange). Those in Role 5 send emails to people of either Role 5 or Role 3 (green), to form a large clique. Role 3 corresponds to receivers and Role 5 corresponds to senders and receivers.

Kim and Leskovec [KL13] stated that an underlying problem in the analysis of time-varying network data is being able to summarize the common structure and dynamics of relations between the entities. A nonparametric multi-group membership model was proposed for dynamic networks, the Dynamic Multi-group Membership Graph Model. **Model** Figure 9(e) contains two components of the model: The birth and death of groups in regards to the dynamics of the network structure, and the connection between nodes memberships to groups and the links of the network. In the link function model, Θ_k parameters are linked within entries of the link affinity matrix between the combinations of members ($z_i^{(t)} = 1$) and nonmembers ($z_i^{(t)} = 0$). $z_i^{(t)}$ denotes binary node group memberships, and probability is symbolized by $p_{ij}^{(t)}$. The DMMG model was evaluated using three datasets for predicting missing links. The NIPS co-authorships and DBLP co-authorships networks connect two people if they appear on the same publication in the conferences. The INFOCOM dataset represents the physical proximity interactions between 78 students at the 2006 INFOCOM conference. **Results** Figure 9(f) shows average evaluation scores for

each model and dataset over 10 runs. The DMMG performs the best in all cases, in comparison to the other three models' baseline models.

For the purposes of scientific exploration, for flexible statistical models in social network analysis, Hanneke et. al [HX07] proposed several flexible statistical models for time-invariant networks. Each are represented as a single directed or undirected graph using. The model most relevant in regards to flexibility is the Exponential Random Graph Models (ERGM), as it offers interactivity through customized captures of a wide range of signature connectivity patterns in the network. Users can specify functions to represent sufficient statistics. [HX07] proposed a "model family" referred to as a temporal ERGM, or TERGM. This family can model network evolution while maintaining the flexibility of a fully general ERGM. **Model** The TERGM was tested to model the network transitions by a dataset that comes from the United States 108^{th} Senate, including $n = 100$ actors. Each time a proposal is made, a single Senator serves as the proposal's *sponsor*, and there may be several *cosponsors*. The first experiment assesses which statistics are important for modeling the network transitions, and the second assesses the quality of fit of a model with a cross-validation experiment. Nine statistics functions were used in total: Density, Stability, Reciprocity, Transitivity, Reverse-Transitivity, Co-Supported, Co-Supporting, Popularity, and Generosity. Figure 9(g) shows the level of support for the last six features. The black arrows show support, and the red arrows indicate an edge at a certain time. "*Popularity* says that if one [Senator] has a supporter, she or he is likely to have another supporter." **Results** Figure 10(a) presents the comparison of all 9 statistic values between ground-truth and sampled networks, using the estimated TERGMs

from the first test, the cross-validation process. The blue lines display the observed statistics (ground-truth), box plots for the sampled networks, and the green lines show the 5- and 95-percentiles.

Oselio et. al [OKI13] developed latent variable models and methods for mining multi-layer networks for connectivity patterns based on noisy data, in which there may be several sources of connective information between user groups. **Models** The general multi-layer graph 10(b) shows the observation matrices being affected by the latent adjacency matrices. The hierarchical model 10(c) shows the latent variable model set Y controlling distributions of the adjacency matrices and further to the observation matrices. The posterior mixture model 10(d) introduced the similarity matrix W and the selection variable Z as two independent latent variable models. **Result** Advancing on DSBM [XI14], a multi-layer extension was implemented for such class membership estimation. This approach was practiced on the Enron email network, with email communication networks between 150 senior employees. Two layers were extracted from the dataset. One is the relational (extrinsic) information between users, and the other represents behavioral (intrinsic) information. Figure 10(e) shows the DSBM parameters for mixing parameters at $\alpha = 0.5$.

Xu and Hero [XI14] sought to improve statistical models for analyzing data in the form of networks, e.g. social networks. They presented a state-space model that extends the typically static stochastic blockmodel to work for dynamic networks. This model is fit in a near-optimal way using the Extended Kalman Filter (a nonlinear estimate of the current mean and covariance). *Model and Result* This dynamic advancement is a temporal extension of the stochastic blockmodel, graphed out by

Figure 10(f), where the boxes represent observed quantities and the ovals represent unobserved quantities. The performance of this model was compared to with a PSA (probabilistic simulated annealing) algorithm, with the PSA performance used as a baseline to compare with the EKF-based algorithm. The performance measurements for the mean-squared tracking error and class estimation accuracy can be seen in Figure 10(g), evaluated with Enron. The EKF significantly outperforms the SSBM in both tracking and class estimation. The EKF performs slightly better than PSA in tracking and slightly worse in class estimation, but with much less computation time.

Xu [Xu14] proposed a new statistical model for dynamic networks, the stochastic block transition model (SBTM), inspired by the stochastic blockmodel (SBM) and his own past work with the dynamic stochastic blockmodel (DSBM) [XI14]. **Model** In this new model, the presence or absence of an edge between two nodes at any given time step directly influences the probability that such an edge would appear at the next time step. This denotes the probability of forming new edges within blocks and the probability of existing edges re-occurring within blocks. **Result** The SBTM approach appears to be more accurate. Figure 10(h) is a comparison against the HM-SBM algorithm, tested with a dataset in a dynamic social network of Facebook wall posts. The SBTM appears to be a better fit for ordering observed adjacency matrices, which makes it better for future interaction prediction.

4.4 Event detection

Event detection is prevalent in dynamic models, as the time evolution offers an order of relevance. A few examples of this are the news, or weather broadcasts, or prompt phone alerts for messages.

As a majority of pre-existing spectral clustering algorithms were unable to incrementally update the clustering results when a small change occurred in a dataset, Ning et. al [NXC+10] presented an incremental approach in an algorithm that continuously updates the eigenvalue system and generates instant cluster labels as the dataset evolves. Their incremental approach of updating the eigenvalue system achieves accurate results and does not demand a high computational cost.

Asuncion et. al [QVUARHS12] introduced a framework for modeling that incorporates time-dependent network statistics and time-varying regression coefficients. They used ideas from survival and event history analysis to create the continuous-time regression modeling framework for network event data.

Ning et. al [NXC+10] introduced the incidence vector/matrix to extend the standard spectral clustering for handling evolving data. It represents two kinds of dynamics in the same framework by incrementally updating the eigen-system: insertion/deletion of data points and similarity change of existing data points. "An incidence *vector* $r_{i,j}(w)$ is a column vector with only two nonzero elements: i-th element equal to $\sqrt{w}$ and j-th element $\sqrt{w}$, indicating data point i and j having a similarity w. The length of the vector is equal to the number of considered data points," and "An incidence *matrix* R is a matrix whose columns are incidence vectors." **Models and**

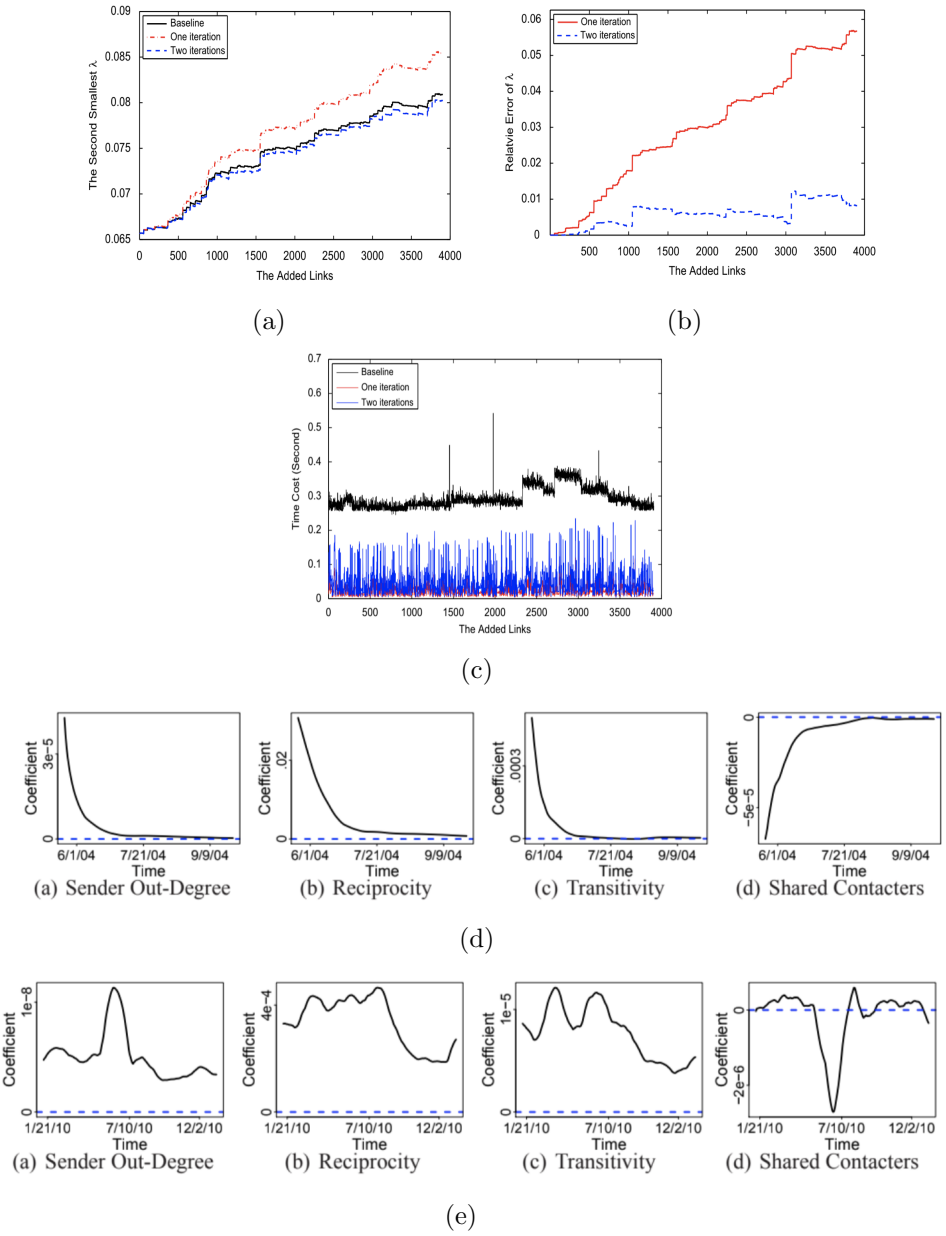


Figure 11: Graphical representations and visuals used in each study for event detection in dynamic networks. (a) The second smallest eigenvalue changing over two iterations. (b) Relative approximation error of the second smallest eigenvalue. (c) Plotted time cost of each added link. (d) Estimated time-varying coefficients on IRVINE data. (e) Estimated time-varying coefficients on METAFILTER data.

Results The proposed algorithm was tested using blog data collected by the NEC laboratories American, using a blog crawler. Figure 11(a) shows that the eigenvalue increases as more links are added, as the second smallest eigenvalue goes through two

iterative changes. Figure 11(b) shows the relative approximation error of the same eigenvalue, with the average relative error at 3.0% for one iteration and 0.59% for two iterations. Figure 11(c) shows the plotted time cost of each added link compared to a baseline. The computational cost for the baseline is much higher than that of the proposed incremental approach.

Asuncion et. al [QVUARHS12] analyzed the structure and evolution of network data. This is an extremely important task in fields such as biology and engineering. The problem in this case is that there is relatively little work to date on continuous-time models for large-scale longitudinal networks. Their solution involves a general regression-based modeling framework for continuous-time network event data. The methods for this framework are built up from past use of multiplicative and additive intensity functions that allow for the incorporation of arbitrary time-dependent network statistics. **Models** The proposed framework is evaluated with the IRVINE and METAFILTER datasets, shown in Figure 11(d) and 11(e). The IRVINE set is longitudinal, obtained from UC Irvine's social network. The METAFILTER data is from a social community blog where users share links and discuss web content. The time-varying coefficients are useful for interpreting network evolution. The coefficients of the IRVINE dataset seem to show two phases in the network's evolution. **Result** "In the first phase of network formation, the network grows at an accelerated rate. Positive coefficients for sender out-degree, reciprocity, and transitivity in these plots imply that users with a high numbers of friends tend to make more friends, tend to reciprocate their relations, and tend to make friends with their friends friends, respectively. However, these coefficients decrease towards zero (the blue line)

and enter a second phase where the network is structurally stable." Meanwhile, the METAFILTER coefficients continuously vary over time.

4.5 Other tasks in a dynamic network

These next papers focus less on the specific tasks discussed above and more on the approaches, such as evolutionary temporal clustering, statistical models for networks, and the handle and processing of big data.

Chi et. al [CSZ+07] contributed to evolutionary temporal clustering, fulfilling the purpose of good clustering results that fit the current used data but also does not deviate too heavily from recent history. They proposed two frameworks that incorporate temporal smoothness into their clustering, thus offering stable and consistent clustering results that are not as sensitive to short-term noises, while are also adaptive to long-term cluster drifts.

Snijders [Sni11] studied and shared statistical models for longitudinal social network data. His research may be seen as a dynamic side to the study of Handcock et. al [HRT07], where both research papers discussed representing network dependencies such as homophily and transitivity.

Continuing from Sandryhaila and Moura [SM14] static approach of signal processing methods to develop methods to analyze and process Big Data, I will move into their dynamic adaptation in handling big data.

Xu et. al [XZYL08] developed two different models for the challenges of evolutionary clustering. They are advancements in evolutionary clustering as both models can automatically learn the number of clusters and the cluster structure at each time

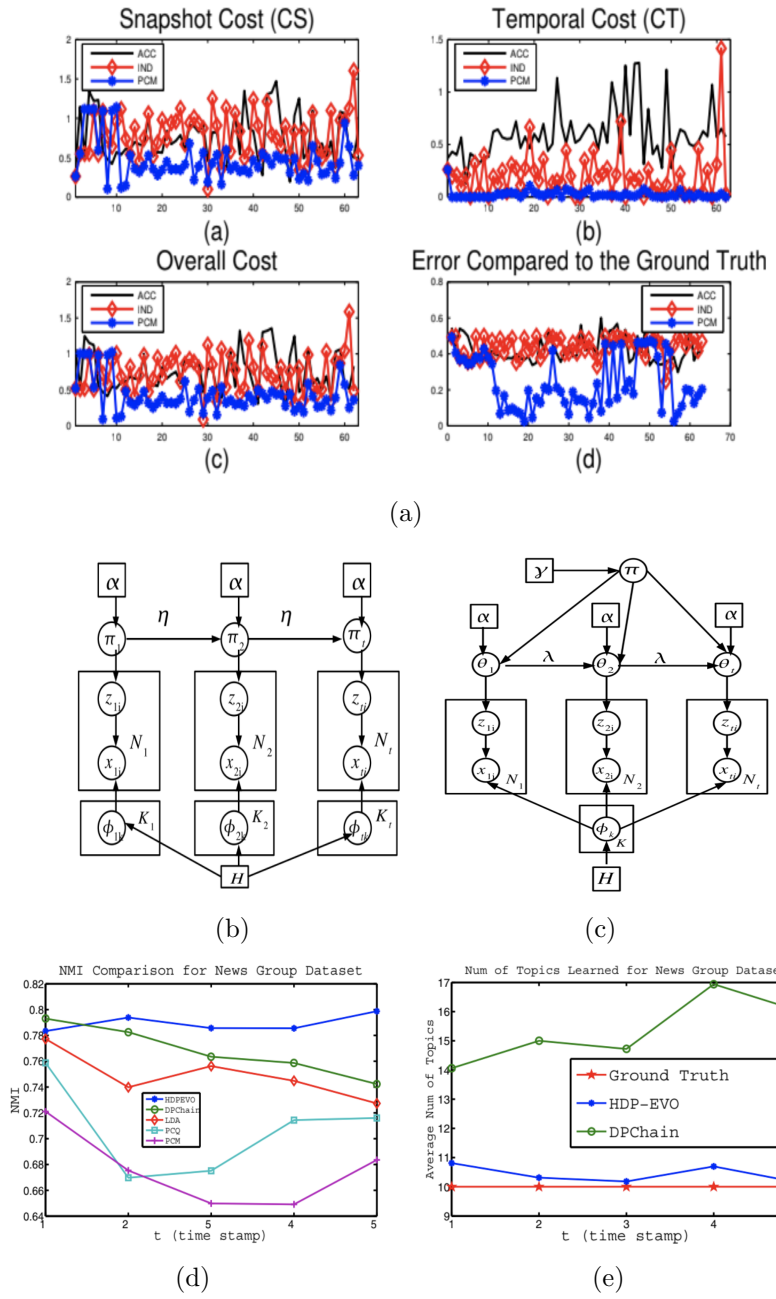


Figure 12: Graphical representations and visuals used in each study for various unique tasks in dynamic networks. (a) Performance of the PCM on the NEC data. (b) The DPChain Model. (c) The HDP-EVO Model. (d) NMI performance comparison of the five algorithms. (e) Cluster number learning performance withe a baseline.

during the evolution.

Chi et. al [CSZ+07] proposed two frameworks Preserving Cluster Quality (PCQ)

and Preserving Cluster Membership (PCM) to incorporate temporal smoothness in evolutionary spectral clustering. **Model** In the PCQ, the temporal cost is expressed as how well the current partition clusters historic data. Example: Two partitions, Z_t and $Z_t^{'}$ cluster the current data at a certain time equally well. To cluster historic data at a time t-1, the clustering performance using partition Z_t must be better than using partition $Z_t^{'}$. In this case, partition Z_t is preferred because it is more consistent with historic data. The idea is translated into a k-means clustering problem. In the PCM, the temporal cost is expressed as the difference between the current partition and the historic partition. Example: Two partitions, Z_t and $Z_t^{'}$, cluster current data at time t equally well. When compared to the historic partition Z_{t-1}, Z_t is much more similar to Z_{t-1} than $Z_t^{'}$ is. Z_t is preferred over $Z_t^{'}$ because Z_t is more consistent with historic partition. **Results** In comparison of the two frameworks, PCQ aims to maximize the trace of a matrix, and PCM is to be maximized. The real data used in the experimental study was blog data that was collected by an NEC in-house blog crawler. It contains 148,681 entry-to-entry links among 407 blogs throughout 63 weeks. In the cost experiment, two baselines were constructed for evaluation: ACC and IND. ACC collects historic data before a certain time and uses the k-means algorithm on the data. IND independently applies the k-means algorithm on the data on that certain time and ignores the preceding data. Figure 12(a) displays the experiment performances, where the evolutionary spectral clustering has the best performance in all four measures. The PCM and PCQ models shared similar results.

Snijders [Sni11] studied statistical models for networks as outcome variables, with a focus on models relevant for social networks. He first discusses network dependencies

and how to represent them. Some of the major dependencies of social networks are reciprocation of directed ties, homophily, transitivity of ties, degree differentials, and hierarchies in directed networks. **One** approach for representing network dependencies in statistical models is to incorporate network structure through covariates or independent residuals. The second approach would be controlling certain aspects of network dependencies while not explicitly modeling them. Conversely, the third approach is to explicitly model the dependencies between tie variables. **Next**, he treated the main various types of statistical models for single, or cross-sectionally observed, networks. There are Conditionally Uniform Models, which considers a set of statistics to be controlled, then assumes that the distribution of networks is uniform, conditional on these controlled statistics. Latent Space Models postulate the existence of a space in which the nodes occupy latent positions, such that the tie indicators are independent conditionally on these positions. Exponential Random Graph Models are based on conditional independence assumptions between the observed tie variables. **Finally**, he discussed the three basic distinctions that can be made between statistical models for network dynamics. 1) Ties may have the nature of changeable states, like friendship. 2) There is a distinction between models where the changes are being driven by the network itself or by a different, perhaps latent, entity. 3) The time variable indexing the dynamic network may be discrete or continuous.

Sandryhaila and Moura [SM14] continued their framework for Big Data analysis with the dynamic approach. Figure 5(f)-(b) is a dynamic example of the product graph. Measurements of a sensor network were indexed by the strong product of the sensor network graph with the time series graph. An application example of DSP_G

on product graphs was considered with data compression. The test uses a dataset containing daily temperature measurements collected by 150 weather stations across the United States. Figure 5(g)-(b) was constructed by connecting each sensor to 8 of its nearest neighbors with undirected edges with weights. It shows measurements from a single day as well as the sensor network graph. The sensor network graph in was used with $N_1 = 150$ nodes and the time series graph in Figure 5(g)-(a) with $N_2 = 365$ nodes.

In evolutionary clustering, both the items and clusters of the collection of data are prone to changing over time, thus posing the challenge of evolutionary clustering in comparison with the traditional clustering. Xu et. al [XZYL08] developed the DPChain and HDP-EVO models as solutions. **Models** The DPChain was based off the DPM model, short for the Dirichlet Process Mixture Chain. In addition to being able to automatically learn the number of the clusters fro evolutionary data, the cluster mixture proportion information is used to reflect a smooth cluster change over the time. The model is illustrated in Figure 12(b). α denotes the concentration parameter for a Dirichlet distribution; $\phi_{t,k}$ denotes the parameter of cluster k of the data at time t; π_t is the cluster mixture proportion vector at time t, and $\pi_{t,k}$ is the weight of the corresponding cluster k at time t. The HDP-EVO, short for HDP Evolutionary Clustering, explicitly capture the cluster correspondence between the data collections of different times. The model is represented in Figure 12(c). λ is the concentration parameter of the Dirichlet distribution of π; ϕ_k is the parameter for a cluster with i.i.d. sampled from a distribution H. (Results) The datasets were evaluated using a constructed dataset based on a subset of the 20 Newsgroups

real data. The graph in Figure 12(d) shows the comparison of normalized mutual information (NMI) performance between the proposed models with three other similar methods (PCQ & PCM, [CSZ+07]), where the two exhibit better results. Figure 12(e) further reports the performance on learning the cluster numbers at different times for the two proposed models.

CHAPTER 5: CONCLUSION & FUTURE WORK

Network visualization is an everyday use, crucial for a significant number of fields in science, marketing, security, etc., when it comes to pursuing more knowledge. I summarized 33 published papers related to data mining, machine learning, and signal processing, all under the topic of graph mining. This survey presents several different, advanced approaches for modeling static and dynamic networks. Whether they decide to follow one or become inspired to make their own advancements, this survey is meant to provide guidance for researchers in the visualization community. The compilation of these studies, old to new, shows how advancements have progressed and will continue to. The models and frameworks presented in this thesis could be used to develop much more powerful and efficient visualization systems for data analysis.

The fields reviewed here are continuously undergoing further development and improvements, and the models treated are being extended in various ways. Some of the older papers, like [ABFX09] and [CSZ+07] have served the purpose of being a basis or root for further research, like [LCZ+08] and [TLZN08], to build upon. In turn, those papers have been even improved or expanded upon in later publications, such as [STH+10] and [Xu14]. Each paper considers new directions that the study could go into, which introduces more challenges to find solutions for. I will discuss a few for some recent publications.

For example, while Yang et. al [YML14] created a model that incorporated the

paramount features of the mixed membership stochastic blockmodel [ABFX09] and the Block-LDA [BC11], they know several improvements can be made for their CESNA model. Some future desires include being able to handle more attributes and information, leading to easier interpretation for their community detection.

Some researchers envisioned their own work being improved by combining their proposed models with others of related goals. [HXA15] considered applying different approaches to their multi-graph stochastic block model, such as the state-space model [XI14]. They suggested additional structures adaptability with additional information, all in all increasing the temporal smoothness and accuracy of results.

Yang and Leskovec [YL15] were able to define ground-truth communities through an intense study of 230 large networks, which was extremely beneficial as most real networks did not provide a ground-truth. Upon this contribution, they suggested future examination of how ground-truth communities are connected, further improving approaches for community detection.

REFERENCES

[ABFX09] Edo M Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. In *Advances in Neural Information Processing Systems 21*, pages 33–40. Curran Associates, Inc., 2009.

[AX08] Amr Ahmed and Eric Xing. Dynamic non-parametric mixture models and the recurrent chinese restaurant process: with applications to evolutionary clustering. In *Proceedings of the 2008 SIAM International Conference on Data Mining*, pages 219–230, 2008.

[BC11] Ramnath Balasubramanyan and William W. Cohen. Block-lda: Jointly modeling entity-annotated text and entity-entity links. In *Proceedings of the 2011 SIAM International Conference on Data Mining*, pages 450–461, 2011.

[CK03] Mark Crovella and E Kolaczyk. Graph wavelets for spatial traffic analysis. In *INFOCOM 2003 Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, volume 3, pages 1848 – 1857 vol.3, 01 2003.

[CSZ+07] Yun Chi, Xiaodan Song, Dengyong Zhou, Koji Hino, and Belle L. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 153–162, New York, NY, USA, 2007. ACM.

[DAM+14] Menoth Mohan Dhanya, Muhammad Tayyab Asif, Nikola Mitrovic, Justin Dauwels, and Patrick Jaillet. Wavelets on graphs with application to transportation networks. *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1707–1712, 2014.

[DKMM04] Petros Drineas, Ravi Kannan, Michael, and W. Mahoney. Fast monte carlo algorithms for matrices iii: Computing a compressed approximate matrix decomposition. *SIAM Journal on Computing*, 36:2006, 2004.

[EH07] William Eberle and Lawrence Holder. Anomaly detection in data represented as graphs. *Intell. Data Anal.*, 11(6):663–689, December 2007.

[For10] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.

[FSX09] Wenjie Fu, Le Song, and Eric P. Xing. Dynamic mixed membership blockmodel for evolving networks. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 329–336, New York, NY, USA, 2009. ACM.

[GNC10] Matan Gavish, Boaz Nadler, and Ronald R. Coifman. Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning. In *Proceedings of the 32nd International Conference on Machine Learning*, 2010.

[HRT07] Mark S. Handcock, Adrian E. Raftery, and Jeremy M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354, 2007.

[HVG09] David Hammond, Pierre Vandergheynst, and Rmi Gribonval. Wavelets on graphs via spectral graph theory. In *Applied and Computational Harmonic Analysis*, volume 30, pages 129–150, 12 2009.

[HX07] Steve Hanneke and Eric P. Xing. Discrete temporal models of social networks. In *Proceedings of the 2006 Conference on Statistical Network Analysis*, ICML'06, pages 115–125, Berlin, Heidelberg, 2007. Springer-Verlag.

[HXA15] Qiuyi Han, Kevin S. Xu, and Edoardo M. Airoldi. Consistent estimation of dynamic and multi-layer block models. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1511–1520, 2015.

[KHRH09] Pavel N. Krivitsky, Mark S. Handcock, Adrian E. Raftery, and Peter D. Hoff. Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social networks*, 31 3:204–213, 2009.

[KL13] Myunghwan Kim and Jure Leskovec. Nonparametric multi-group membership model for dynamic networks. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 1385–1393. Curran Associates, Inc., 2013.

[LCZ+08] Yu-Ru Lin, Yun Chi, Shenghuo Zhu, Hari Sundaram, and Belle L. Tseng. Facetnet: A framework for analyzing communities and their evolutions in dynamic networks. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 685–694, New York, NY, USA, 2008. ACM.

[NXC+10] Huazhong Ning, Wei Xu, Yun Chi, Yihong Gong, and Thomas S. Huang. Incremental spectral clustering by efficiently updating the eigen-system. *Pattern Recogn.*, 43(1):113–127, January 2010.

[OKI13] Brandon Oselio, Alex Kulesza, and Alfred O. Hero III. Multi-layer graph analytics for dynamic social networks. *CoRR*, abs/1309.5124, 2013.

[PBL14] Simon Pool, Francesco Bonchi, and Matthijs van Leeuwen. Description-driven community detection. *ACM Transactions on Intelligent Systems and Technology*, 5(2):28:1–28:28, April 2014.

[QVUARHS12] Duy Q Vu, Arthur U Asuncion, David R Hunter, and Padhraic Smyth. Continuous-time regression models for longitudinal networks. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, volume 24, 05 2012.

[RGNH13] Ryan A. Rossi, Brian Gallagher, Jennifer Neville, and Keith Henderson. Modeling dynamic behavior in large evolving graphs. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pages 667–676, New York, NY, USA, 2013. ACM.

[SDB+16] Arlei Silva, Xuan-Hong Dang, Prithwish Basu, Ambuj K. Singh, and Ananthram Swami. Graph wavelets via sparse cuts. *CoRR*, abs/1602.03320, 2016.

[SM14] A. Sandryhaila and J. M. F. Moura. Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure. *IEEE Signal Processing Magazine*, 31(5):80–90, Sept 2014.

[SNF+13] David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs. *IEEE Signal Process. Mag.*, 30(3):83–98, 2013.

[Sni11] Tom A.B. Snijders. Statistical models for social networks. *Annual Review of Sociology*, 37(1):131–153, 2011.

[STH+10] Yizhou Sun, Jie Tang, Jiawei Han, Manish Gupta, and Bo Zhao. Community evolution detection in dynamic heterogeneous information networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, MLG '10, pages 137–146, New York, NY, USA, 2010. ACM.

[TfWAK03] Ben Taskar, Ming fai Wong, Pieter Abbeel, and Daphne Koller. Link prediction in relational data. In *in Neural Information Processing Systems*, 2003.

[TLZN08] Lei Tang, Huan Liu, Jianping Zhang, and Zohreh Nazeri. Community evolution in dynamic multi-mode networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 677–685, New York, NY, USA, 2008. ACM.

[TPS+08] Hanghang Tong, Spiros Papadimitriou, Jimeng Sun, Philip S. Yu, and Christos Faloutsos. Colibri: Fast mining of large static and dynamic graphs. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 686–694, New York, NY, USA, 2008. ACM.

[XI14] Kevin S. Xu and Alfred O. Hero III. Dynamic stochastic blockmodels for time-evolving social networks. *CoRR*, abs/1403.0921, 2014.

[Xu14] Kevin S. Xu. Stochastic block transition models for dynamic networks. *CoRR*, abs/1411.5404, 2014.

[XZYL08] T. Xu, Z. Zhang, P. S. Yu, and B. Long. Dirichlet process based evolutionary clustering. In *2008 Eighth IEEE International Conference on Data Mining*, pages 648–657, Dec 2008.

[YCZ+11] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Detecting communities and their evolutions in dynamic social networks–a bayesian approach. *Machine Learning*, 82(2):157–189, February 2011.

[YL15] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, January 2015.

[YML14] Jaewon Yang, Julian J. McAuley, and Jure Leskovec. Community detection in networks with node attributes. *CoRR*, abs/1401.7267, 2014.