Mining Social and Information Networks: Dynamics and Applications

Fragkiskos Malliaros

Mining Social and Information Networks: Dynamics and Applications

by

FRAGKISKOS MALLIAROS

A dissertation submitted in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

ÉCOLE POLYTECHNIQUE
LABORATOIRE D'INFORMATIQUE

September 2015

MINING SOCIAL AND INFORMATION NETWORKS: DYNAMICS AND APPLICATIONS

by

FRAGKISKOS MALLIAROS

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Subject: Computer Science

Date of defense: September 16, 2015

DISSERTATION COMMITTEE

Michalis Vazirgiannis (Advisor)	École Polytechnique
Jean-Loup Guillaume (Reviewer)	University of La Rochelle
Jie Tang (Reviewer)	Tsinghua University
Christos Faloutsos (Examiner)	Carnegie Mellon University
Aristides Gionis (Examiner)	Aalto University
Balázs Kégl (Examiner)	CNRS, University of Paris-Sud
Frank Nielsen (Examiner)	École Polytechnique
Marc Tommasi (Examiner)	University of Lille 3

ÉCOLE POLYTECHNIQUE
LABORATOIRE D'INFORMATIQUE

September 2015

ABSTRACT

Networks (or graphs) have become ubiquitous as data from diverse disciplines can naturally be mapped to graph structures. Characteristics examples include social and technological networks, collaboration networks, or even networks generated by textual data. The problem of extracting meaningful information from large scale graph data in an efficient and effective way has become crucial and challenging with several important applications and towards this end, graph mining and analysis methods constitute prominent tools. The goal of this dissertation is to study and address challenging problems in this area, focusing on both the design of new graph mining algorithms and tools, as well as on studying the dynamics and properties of real-world networks.

In the first part of the thesis, we study the structure and dynamics of real-world social graphs, focusing on the property of engagement. Typically, engagement refers to the degree that an individual participates (or is encouraged to participate) in a community and is closely related to the departure dynamics of nodes, i.e., the tendency of individuals to leave the community. Building upon recent game-theoretic studies, we propose measures for characterizing the engagement at both node and graph level, that are based on the core decomposition of the underlying graph. We have performed experiments on a multitude of real graphs, observing interesting connections with other graph characteristics, as well as a clear deviation from the corresponding behavior of random graphs. Then, we study a cascading effect on the network, where the departure of a user may affect the engagement level of his neighbors in the graph, that successively may also decide to leave, leading to a disengagement epidemic. We introduce a new concept of vulnerability assessment under cascades triggered by the departure of nodes based on their engagement level. Our results indicate that social networks are robust under cascades triggered by randomly selected nodes but highly

vulnerable in cascades caused by targeted departures of nodes with high engagement level.

In the second part, we focus on the problem of information spreading and more precisely on how to identify influential nodes in complex networks. This constitutes a crucial task in many application domains, including viral marketing and information diffusion. Our goal is to locate individual influential nodes in the network, which are able to perform fast and wide epidemic spreading. To that end, we capitalize on the properties of the K-truss decomposition, a triangle-based extension of the core decomposition of graphs. Our analysis on real-world networks indicates that the nodes belonging to the maximal K-truss subgraph of the network show better spreading behavior compared to previously used importance criteria, including node degree and k-shell index. Additionally, not only more nodes get infected during the outbreak of the epidemic, but also the total number of nodes infected at the epidemic's fadeout is greater. We further show that nodes belonging to such dense subgraphs, dominate the small set of nodes that achieve the optimal spreading in the network.

In the last part, we investigate how graph mining tools can be used to enhance traditional text mining problems and specifically the one of text categorization, i.e., the supervised learning task of assigning a textual document to a set of predefined categories. In the typical Bag-of-Words model, the text is represented as a multiset of its terms, disregarding dependence and ordering of the words, but keeping only information about the frequency of appearance. We propose a framework for text categorization adopting a graph-based representation of documents that encodes relationships between the different terms. Based on this formulation, we treat the term weighting task as a node ranking problem in the interconnected feature space defined by the graph; the importance of a term is determined by the importance of the corresponding node in the graph, using node centrality criteria. We also introduce novel global weighting schemes at the document collection level in order to penalize commonly used terms. Furthermore, we propose an unsupervised graph-based feature (i.e., term) selection approach, based on the properties of the k-core decomposition. The significance of our approach stems from the fact that we augment the unigram feature space of

the learning task with weights that implicitly consider n-gram information in the document – as expressed by paths in the graph – without increasing the dimensionality of the problem. Our results indicate that the proposed weighting mechanisms produce more discriminative feature weights for text categorization, outperforming existing frequency-based criteria.

RÉSUMÉ

Les réseaux (ou graphes) sont devenus omniprésents comme en attestent les données utilisées dans de nombreuses et diverses disciplines qui peuvent être naturellement représentées sous forme de graphes. Les réseaux sociaux, technologiques et collaboratifs ainsi que les graphes de données textuelles en sont des exemples typiques. Le but de cette thèse est d'étudier et de répondre à des problématiques liées aux graphes sous deux aspects: la mise au point de nouveaux outils et algorithmes de fouille de données graphiques et l'étude des dynamiques et des propriétés des réseaux existants dans la vie réelle.

La première partie de ce mémoire est consacrée à l'étude de la structure et des dynamiques des réseaux sociaux existants, en mettant l'accent sur la notion d'engagement. En effet, cette dernière désigne la propension d'un individu à participer ou non à une communauté et elle est liée à la dynamique de départ des nœuds dans un graphe. En se basant sur des travaux récents en théorie des jeux, nous proposons de nouvelles mesures de l'engagement à l'échelle des nœuds et du graphe tout entier basées sur la décomposition en k-core. Nous expérimentons sur des graphes réels, observant des corrélations intéressantes avec d'autres propriétés ainsi qu'un comportement significativement différent de celui des graphes aléatoires. Ensuite, nous étudions l'effet dit de cascade où le départ d'un individu peut affecter l'engagement de ses voisins et ainsi de suite, créant une épidémie de départs par contagion. Nous introduisons un nouveau concept d'évaluation de la vulnérabilité des réseaux en cas de cascades déclenchées par le départ d'individus et ce en fonction de leur engagement. Les résultats indiquent que les réseaux sociaux sont robustes en cas de cascades déclenchées en sélectionnant aléatoirement les individus (c'est-à-dire les sommets du graphe) mais très vulnérables lorsque les individus à l'origine des cascades ont un degré d'engagement élevé.

La deuxième partie est consacrée à l'identification des nœuds influents (c'est-à-dire les individus capables d'initier rapidement des épidémies de grande ampleur) dans les réseaux complexes, tâche cruciale dans de nombreuses applications telles que le marketing viral et la diffusion d'information. Nous utilisons les propriétés de la décomposition en K-truss, une récente extension de la décomposition k-core basée sur les triangles. Notre étude des réseaux réels indique que les sommets appartenant au sous-graphe maximal de plus grande K-truss ont un meilleur comportement de diffusion comparés aux sommets de plus grand degré ou indice k-shell. En outre, on observe que le nombre de sommets infectés au déclenchement et à la fin (nombre total d'individus) de l'épidémie est plus grand.

Dans la dernière partie, nous nous intéressons à l'application des outils de fouille de données graphiques au problème de classification de textes. Dans la représentation par sac-de-mots classique, un texte est représenté par un multiensemble de termes, ignorant l'ordre et les relations de dépendance entre les mots. Nous proposons une approche alternative basée sur les graphes où chaque document est représenté par un graphe qui encode les relations de co-occurrence entre les mots, représentation dite par graphe-de-mots. En utilisant cette formulation, nous traitons la pondération des termes d'un document comme un problème de classement des sommets dans l'espace de caractéristiques interconnecté défini par le graphe. L'importance d'un mot est déterminée par l'importance du sommet associé, mesurée en utilisant un critère de centralité. L'intérêt de notre approche réside dans le fait que nous enrichissons l'espace des unigrammes par des poids qui considèrent implicitement des n-grammes dans le document, sans pour autant augmenter la dimension du problème et nos résultats suggèrent que les mécanismes de pondération proposés produisent des caractéristiques explicatives plus discriminatives pour la classification de textes.

LIST OF PUBLICATIONS

The following publications and submissions under review are included in parts or in an extended version in this thesis:

- Fragkiskos D. Malliaros and Michalis Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics Reports* 533:4 (2013), pp. 95 –142.

- Fragkiskos D. Malliaros and Michalis Vazirgiannis. To Stay or Not to Stay: Modeling Engagement Dynamics in Social Graphs. In *CIKM '13: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, 2013, pp. 469–478.

- Maria-Evgenia G. Rossi, Fragkiskos D. Malliaros, and Michalis Vazirgiannis. Spread It Good, Spread It Fast: Identification of Influential Nodes in Social Networks. In *WWW '15: Proceedings of the 24th International Conference on World Wide Web Companion*, 2015, pp. 101–102.

- Fragkiskos D. Malliaros and Michalis Vazirgiannis. Vulnerability Assessment in Social Networks under Cascade-based Node Departures. *EPL (Europhysics Letters)* 110:6 (2015), p. 68006.

- Fragkiskos D. Malliaros and Konstantinos Skianis. Graph-Based Term Weighting for Text Categorization. In *ASONAM '15: 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Social Media and Risk Workshop*, 2015.

- Fragkiskos D. Malliaros, Konstantinos Skianis, and Michalis Vazirgiannis. A Graph-Based Framework for Text Categorization: Term Weighting and Selection as Ranking of Interconnected Features. *Manuscript* (2015).

- Fragkiskos D. Malliaros, Maria-Evgenia G. Rossi, and Michalis Vazirgiannis. Locating Influential Nodes in Complex Networks. *Manuscript* (2015).

In addition to the topics studied in this dissertation which are mentioned above, during my Ph.D. studies, I worked on several other problems including community detection and robustness estimation in large networks, leading to the following publications:

- Fragkiskos D. Malliaros, Vasileios Megalooikonomou, and Christos Faloutsos. Estimating Robustness in Large Social Graphs. *Knowledge and Information Systems* (2014), pp. 1–34.

- Christos Giatsidis, Fragkiskos D. Malliaros, Dimitrios M. Thilikos, and Michalis Vazirgiannis. CoreCluster: A Degeneracy Based Graph Clustering Framework. In *AAAI '14: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014, pp. 44–50.

- Jordi Casas-Roma, Fragkiskos D. Malliaros, and Michalis Vazirgiannis. k-Degree Anonymity on Directed Networks. *Manuscript* (2015).

As you set out for Ithaca
hope the voyage is a long one,
full of adventure, full of discovery.
Laistrygonians and Cyclops,
angry Poseidon – don't be afraid of them:
you'll never find things like that on your way
as long as you keep your thoughts raised high,
as long as a rare excitement
stirs your spirit and your body.
...

Hope the voyage is a long one.
May there be many a summer morning when,
with what pleasure, what joy,
you come into harbors seen for the first time;
may you stop at Phoenician trading stations
to buy fine things,
mother of pearl and coral, amber and ebony,
sensual perfume of every kind –
as many sensual perfumes as you can;
and may you visit many Egyptian cities
to gather stores of knowledge from their scholars.

Keep Ithaca always in your mind.
Arriving there is what you are destined for.
But do not hurry the journey at all.
Better if it lasts for years,
so you are old by the time you reach the island,
wealthy with all you have gained on the way,
not expecting Ithaca to make you rich.

Ithaca gave you the marvelous journey.
Without her you would not have set out.
She has nothing left to give you now.

And if you find her poor, Ithaka won't have fooled you.
Wise as you will have become, so full of experience,
you will have understood by then what these Ithakas mean.

– Ithaca, C. P. Cavafy

ACKNOWLEDGMENTS

This dissertation constitutes the end of an eleven-year student period. At this point, I would like to heartily thank the people that have supported me all those years.

First and foremost, I would like to express my sincere gratitude to my advisor Prof. Michalis Vazirgiannis, without whom this dissertation would not have been possible. Under his supervision, I enjoyed two key elements that I could ever have imagined as a graduate student: limitless academic freedom, but simultaneously, absolute academic care. By spending plenty of time on our long brainstormings, I have really admired his ability to simplify difficult concepts, while being always "to the point". His energy, curiosity, patience, constant encouragement and support about any of my steps, as well as his values of life in general, have been proven particularly significant points for my work and my personality in general. I am deeply indebted to him for making my Ph.D. studies a pleasant and memorable journey.

I would also like to express my gratitude to the members of my Ph.D. thesis committee, Prof. Jean-Loup Guillaume (University of La Rochelle), Prof. Jie Tang (Tsinghua University), Prof. Christos Faloutsos (Carnegie Mellon University), Prof. Aristides Gionis (Aalto University), Dr. Balázs Kégl (CNRS, University of Paris-Sud), Prof. Frank Nielsen (École Polytechnique) and Prof. Marc Tommasi (University of Lille 3), for their valuable feedback on my work and their thought-provoking questions during my thesis defense.

I am especially grateful to Prof. Christos Faloutsos for the various interactions that we had over the last five years. I would like to thank him for not only being a research collaborator and member of my Ph.D. committee, but also kindly advising me for my further career, and suggesting me interesting directions for my work.

My academic journey started at the University of Patras, Greece, where I did my first steps in research, initially as an undergraduate and then as a Master's student. The excellent environment and professors at the

Department of Computer Engineering and Informatics provided me with a solid academic background necessary for my future steps.

A sincere thank you goes to Prof. Vasileios Megalooikonomou who was my advisor at the University of Patras. He introduced me to research and taught me how to work methodologically in order to address a problem. I would also like to thank him for encouraging me to pursue graduate studies and for his advices about my plans.

I have been truly lucky to interact with many brilliant people at École Polytechnique. I would like to thank all the members of the DaSciM group, current and past, for all I learned from them: Konstantinos Skianis, Maria-Evgenia Rossi, Panagiotis Korvesis, Dr. Christos Giatsidis, Dr. François Rousseau, Marc Mitri, Dr. Jordi Casas-Roma, Evangelos Anagnostopoulos, Dr. Stamatina Thomaidou, Giannis Bekoulis, Giannis Nikolentzos, Polykarpos Meladianos, Jonghoon Kim, Emmanouil Kiagias, Dr. Nikolaos Tziortziotis and Dr. Antoine Tixier.

During my academic years, I was extremely fortunate to have amazing collaborators (listed in chronological order of collaboration): Vasileios Megalooikonomou, Christos Faloutsos, Christos Giatsidis, Amalia Charisi, Evangelia I. Zacharaki, Dimitrios M. Thilikos, Maria-Evgenia Rossi, Konstantinos Skianis, Jordi Casas-Roma, Marc Mitri, Giannis Nikolentzos and Apostolos N. Papadopoulos. I would like to express my gratitude to all of them for their various contributions to the work presented in this dissertation and beyond.

I would also like to thank Dr. Juan (Julia) Liu and Dr. Kumar Sricharan for their great hospitality at the Palo Alto Research Center (PARC) in California, where I spent the summer of 2014 as a research intern.

For my graduate studies, I had the honor to be awarded the Google European Doctoral Fellowship in Graph Mining, that supported most of my Ph.D. work. I would like to thank my mentor at Google, Dr. Ioannis Tsochantaridis (Google Research, Zürich) for the discussions that we had on topics of my dissertation, as well as Beate List for her kind assistance regarding various aspects of the fellowship.

Being a graduate student is not only about work, but also fun. I was fortunate to meet wonderful people in Paris and to make long standing

friendships. I would like to thank all my friends in Fondation Hellénique and in Paris in general, that made my stay in France unforgettable. Special thanks go to my closest friends Panagiotis (for our long standing friendship that started out from our years in Patras), Giorgos, Agape and Maro for the precious moments that we have shared and continue to share.

Many thanks go to my good old friends from Greece, Nikos, Takis, Vassilis and Mpampis. Even though we only meet once or twice per year, we are still enjoying our almost daily chats. Thank you for your friendship and the memorable moments that we will always remember.

I would also like to warmly thank Hanane for being in my life. Her support, patience and care during the writing of this dissertation and beyond are immeasurable. Thank you for everything you have brought to my life.

Last but not least, I would like to thank from the depths of my heart my Family back in Greece. My parents Dimitris and Stavroula, my sister Konstantina, my brother-in-law Spyros and my lovely niece Maria, as well as my grandparents Fragkiskos, Giannoula and Nikos and my late grandmother Konstantina. This dissertation was made possible due to your infinite and unconditional love, support and encouragement. Thank you for everything you have done and continue doing for me.

Fragkiskos Malliaros
Paris, Fall 2015

To my parents Stavroula and Dimitris
for their endless love and support.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

Networks have become ubiquitous as data from many different disciplines can naturally be mapped to graph structures [New10; CF12; New03]. *Social networks*, such as academic collaboration networks [Tan+08], sexual networks and interaction networks over online social networking applications are used to represent and model the social ties among individuals. *Information networks*, including the hyperlink structure of the Web and blog networks, have become crucial mediums for information dissemination, offering an effective way to represent content and navigate through it. A plethora of *technological networks*, including the Internet, power grids, telephone networks and road networks are an important part of everyday life. *Biological, ecological* and networks from the domain of *neuroscience*, such as protein-protein interaction, neural, gene regulatory, brain networks and food webs, can be used to model the function and interaction of natural entities towards a better understanding of phenomena that occur in nature. Additionally, networks can be used as proxies to deal with problems and data that do not inherently contain graph structure; an example of this case is the one of textual data, where information about the terms and their relationships is represented as a graph.

Real-world networks, as the ones presented above, are not classified as random (e.g., the Erdös-Rényi random graph model [ER60]); they present fascinating patterns and properties conveying that their inherent structure is not governed by randomness. The degree distribution is skewed, following a power-law [BA99; FFF99]; the average distance between nodes in the network is short (the so-called small-world phenomenon [Mil67; AJB99; LH08]); the ties between entities do not always represent reciprocal relations forming directed networks with non symmetric links [New03]; the edge distribution is inhomogeneous resulting in the existence of community structure [GN02].

Being able to analyze and understand the dynamics that govern real networks is a critical step for various real-world applications.

Due to the extent and the diversity of contexts in which networks appear as well as their far from random interesting properties, the area of graph mining and analysis has become crucial and challenging, with the following points being of particular interest:

(i) Design of effective and efficient graph mining algorithms.

(ii) Apply the algorithms to analyze and understand the features, the structure and the dynamics of complex systems.

(iii) Utilize the extracted knowledge for solving real-world problems.

That way, *Network Science* has emerged as an interdisciplinary area spanning traditional domains including computer science, mathematics, sociology, biology and economics. Motivated by the above points, the driving force behind this dissertation is the importance of graph mining techniques towards studying and analyzing social and information networks. To that end, we propose models and algorithmic tools to address various challenging problem which arise in this area.

1.1 THESIS STATEMENT AND OVERVIEW OF CONTRIBUTIONS

This dissertation contributes models, tools and observations to problems that arise in the area of mining social and information networks. We built upon computationally efficient graph mining methods in order to:

(i) Design models for analyzing the structure and dynamics of real-world networks towards unraveling properties that can further be used in practical applications.

(ii) Develop algorithmic tools for large-scale analytics on data with inherent (e.g., social networks) or without inherent (e.g., text) graph structure.

Our models and algorithmic tools utilize the concept of core decomposition, an efficient hierarchical representation of the graph with several interesting

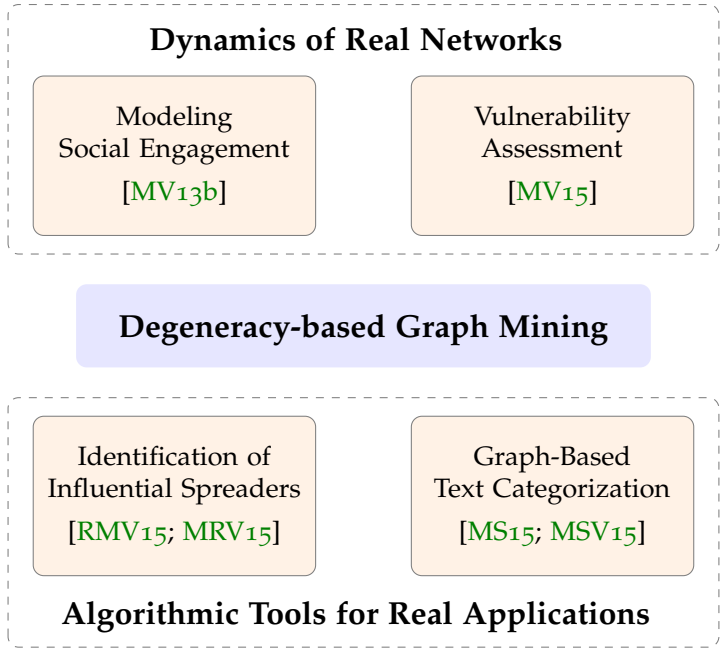


Figure 1.1: Overview of the contributions presented in the thesis.

properties. In particular, for the former point we show how to model the engagement dynamics of large social networks and how to assess their vulnerability with respect to user departures from the network. In both cases, by unraveling the dynamics of real social networks, regularities and patterns about their structure and formation can be identified; such knowledge can further be used in various applications including prediction, anomaly detection and building robust social networking systems.

For the latter, we examine how to identify influential users in complex networks, having direct applications to epidemic control and viral marketing and how to utilize graph mining techniques in order to enhance text analytics tasks including opinion mining and automatic news classification. Next, we provide an overview of the contributions of the dissertation with respect to the above points. Figure 1.1 depicts the overview of the thesis.

1.1.1 *Dynamics of Real Networks*

MODELING SOCIAL ENGAGEMENT. *Given a large social graph, how to model the engagement dynamics of individuals?*

Social engagement refers to the degree that an individual participates in the activities of a community and it is closely related to the departure dynamics of users, i.e., the tendency of individuals to leave the community. We propose local (i.e., node level) and global (i.e., graph level) models of engagement based on the properties of k-core decomposition of the underlying social graph and we examine whether they depend on structural and topological features of the graph. We perform experiments on a multitude of real graphs, observing interesting connections to other graph characteristics, as well as a clear deviation from the corresponding behavior of random graphs. The proposed models can further help us to better understand the structural dynamics of social networks, with direct implications on how to build more stable and robust social networking systems (Chapter 3).

VULNERABILITY ASSESSMENT IN SOCIAL NETWORKS. *How to model and assess the vulnerability of social networks under cascades of user departures?*

In social networks, new users decide to become members, but also current users may depart or stop being active. This phenomenon, also known as churn or attrition, has been an important topic in the business domain. We study a cascading effect on the network where the departure of a user may affect the engagement level of his neighbors in the graph, that successively may also decide to leave, leading to a disengagement epidemic. Being able to model and analyze such cascading processes in real social networks is an important task towards understanding the vulnerability of those social interaction systems under churn. To that end, we introduce a model to capture the social contagion effect and we propose a new concept of vulnerability assessment under cascades triggered by the departure of nodes based on their engagement level. Our experimental results indicate that social networks are robust under cascades triggered by randomly selected nodes but highly vulnerable in cascades caused by targeted departures of nodes

with high engagement level – complementing seminal results in network science (Chapter 4).

1.1.2 *Graph-Based Algorithmic Tools for Data Analytics*

IDENTIFICATION OF INFLUENTIAL SPREADERS. *How to locate influential users in social networks?*

Detecting influential spreaders in complex networks, i.e., individuals who are able to efficiently spread information, is a crucial task with many diverse applications, including viral marketing and control of disease propagation in epidemiology. Capitalizing on the properties of the K-truss decomposition, a triangle-based extension of the k-core decomposition, we propose a method to locate influential nodes. Our results on real networks indicate that the nodes identified by the proposed method show better spreading behavior compared to previously used importance criteria, leading to faster and wider epidemic spreading (Chapter 5).

GRAPH-BASED TEXT CATEGORIZATION. *How to use graph mining to enhance large-scale text analytics?*

Text categorization (or classification), i.e., the supervised process of assigning category labels to documents, is a core data analytics task which lies in the heart of many real applications, including opinion mining in product reviews and automatic news classification. Building upon the fact that graphs can be used to represent textual content, we propose a graph-based framework for text categorization. In particular, we extract graphs from text, where the nodes correspond to terms and the edges capture term co-occurrence relationships – addressing the term-independence assumption widely used in many text analytics tasks. That way, the term weighting process is treated as a node ranking problem in the feature space defined by the graph. We propose an unsupervised graph-based feature (i.e., term) selection approach, using the properties of the k-core decomposition. Our experimental results indicate that the proposed weighting mechanisms produce more discriminative feature weights for text categorization, outperforming existing frequency-based criteria (Chapter 6).

1.2 OUTLINE OF THE THESIS

The rest of the dissertation is organized as follows. In Chapter 2 we present basic concepts and background material that will be used throughout the dissertation. The next two Chapters are devoted to our work concerning the dynamics of social networks; in particular, Chapter 3 presents the work about social engagement and Chapter 4 the vulnerability assessment under disengagement social contagion. In Chapter 5 we present our work on locating influential users in complex networks. Chapter 6 describes how to apply graph-based methods for the text categorization task. Finally, in Chapter 7, we offer concluding remarks about the topics covered in the dissertation and future research directions.

2

BASIC CONCEPTS AND PRELIMINARIES

In this Chapter we provide the basic concepts and background material that will be used throughout the dissertation. Initially, we give the definitions for basic graph theoretic and linear algebraic concepts. Special mention is given to the k-core decomposition in networks and its applications, since it constitutes a central topic of the dissertation. Finally, we describe the graph datasets used in Chapters 3, 4 and 5. Table 2.1 gives a list of used symbols along with their definition. For completeness in the presentation, in each Chapter we provide all the necessary symbols and background material. For a general introduction to the field of complex networks, the reader may refer to Refs. [Boc+06; New03; CF12].

2.1 BASIC CONCEPTS IN GRAPH THEORY

A *network* is usually represented by a *graph* (throughout the dissertation we use the terms network and graph interchangeably). A graph $G = (V, E)$ consists of a set of nodes V and a set of edges $E \subseteq V \times V$ which connect pairs of nodes (in some cases, the nodes and edges of a graph are also called vertices and links respectively). The number of nodes in the graph is equal to $n = |V|$ and the number of edges $m = |E|$. A graph may be *directed* or *undirected*, *unipartite* or *bipartite* and the edges may contain *weights* or not. Figure 2.1 depicts some examples of different types of graphs.

Definition 2.1 (Directed and Undirected Graph)**.** *In a directed graph $G_D = (V, E)$, every edge $(i, j) \in E$ links node i to node j (ordered pair of nodes). An undirected graph $G = (V, E)$ is a directed one where if edge $(i, j) \in E$, then edge $(j, i) \in E$ as well.*

Definition 2.2 (Bipartite Graph)**.** *A graph $G_B = (V_h, V_a, E_b)$ is called bipartite if the node set V can be partitioned into two disjoint sets V_h and V_a, where $V =$*

Symbol	Definition
$G = (V, E)$	Network
$G_B = (V_h, V_a, E_b)$	Bipartite network
$V,\ E$	Set of nodes and edges for network G
$\|V\| = n,\ \ \|E\| = m$	Number of nodes and edges in the network
$e = (u, v)$	Edge $e \in E$ from node u to node v
d_v	Degree of node v
$Nb(v)$	Set of neighboring nodes of v
$\mathbf{A}$	Adjacency matrix of a graph
A_{ij}	Entry of matrix $\mathbf{A}$
$\mathbf{A}^T$	The transpose of matrix $\mathbf{A}$
λ_i	i-th largest eigenvalue of a matrix
$\mathbf{u}_i$	Eigenvector corresponds to i-th eigenvalue
u_{ij}	i-th component of j-th eigenvector

Table 2.1: Symbols and definitions.

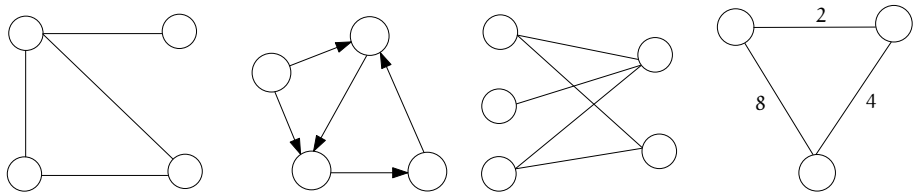
$V_h \cup V_a$, such that every edge $e \in E_b$ connects a node of V_h to a node of V_a, i.e., $e = (i, j) \in E \Rightarrow i \in V_h$ and $j \in V_a$. In other words, there are no edges between nodes of the same partition.

Every graph $G = (V, E)$ (directed or undirected, weighted or unweighted) can be represented by its *adjacency matrix* $\mathbf{A}$. Matrix $\mathbf{A}$ has size $|V| \times |V|$ (or $n \times n$), where the rows and columns represent the nodes of the graph and the entries indicate the existence of edges.

Definition 2.3 (Adjacency Matrix)**.** *The adjacency matrix $\mathbf{A}$ of a graph $G = (V, E)$ is an $|V| \times |V|$ matrix, such that*

$$A_{ij} = \begin{cases} w_{ij}, & \text{if } (i, j) \in E,\ \ \forall\, i, j \in 1, \ldots, |V| \\ 0, & \text{otherwise.} \end{cases}$$

The above definition is rather general and is suitable both for weighted and unweighted graphs. For the former case, each value w_{ij} represents the weight associated with the edge (i, j), while for the latter case of unweighted graphs the weight of each edge is equal to one (i.e., $w_{ij} = 1, \forall (i, j) \in E$). If the graph is undirected, the adjacency matrix $\mathbf{A}$ is symmetric, i.e., $\mathbf{A} = \mathbf{A}^T$, while for directed graphs the adjacency matrix is nonsymmetric.



(a) Undirected graph (b) Directed graph (c) Bipartite graph (d) Weighted graph

Figure 2.1: Examples of different types of graphs. In the case of directed graph (b), the arrows indicate the directionality of each edge. In the weighted graph (d) the values associated with each edge represent the weights (a weighted graphs can be directed or undirected).

DEGREE. A basic property of the nodes in a graph is their *degree*. In an undirected graph G, a node has degree d if it has d incident edges. In the case of directed graphs, every node is associated with an *in-degree* and an *out-degree*. The in-degree d_i^{in} of node $i \in V$ is equal to the number of incoming edges, i.e., $d_i^{in} = \|j|(j,i) \in E\|$, while the out-degree d_i^{out} of node $i \in V$ equals to the number of outgoing edges, i.e., $d_i^{out} = \|j|(i,j) \in E\|$. In undirected graphs, the in-degree is equal to the out-degree, i.e., $d_i = d_i^{in} = d_i^{out}$, $\forall i \in V$. The *degree matrix* is defined as the diagonal $n \times n$ matrix $\mathbf{D}$, with the degree of each node in the main diagonal (zero entries outside main diagonal). Similarly, in directed graphs we can define the in-degree matrix $\mathbf{D}_{in}$ and out-degree matrix $\mathbf{D}_{out}$ for the in- and out- degrees respectively.

CONNECTEDNESS. Let $G = (V, E)$ be an undirected graph. A *path* is defined as a sequence of nodes $v_1, v_2, \ldots, v_{k-1}, v_k$, with the property that every consecutive pair of nodes v_i, v_{i+1} in the sequence is connected by an edge. Two nodes $i, j \in V$ are called *connected* if there is a path in G from node i to node j. The above definitions can be extended to directed networks, where in a *directed path*, a directed edge should exist from each node of the sequence to the next node.

CONNECTED GRAPH. An undirected graph $G = (V, E)$ is called *connected*, if for every pair of nodes $i, j \in V$ a path exists from node i to node j. In the case of directed networks, three different notions of connectivity can

be defined. A directed graph is called *strongly connected* if for every pair of nodes $i, j \in V$, there is a directed path from i to j and a directed path from j to i. A directed graph is *connected* if for every pair of nodes $i, j \in V$, it contains a directed path from i to j or from j to i. Lastly, a directed graph is called *weakly connected* if ignoring the directionality of the edges (i.e., replacing the directed edges with undirected), a connected graph is produced.

CONNECTED COMPONENT. A *connected component* in an undirected graph is a maximal subgraph where every pair of nodes is connected by a path. For directed graphs, the notions of *strongly connected component* and *weakly connected component* can be defined. In the former case, similar to the definition of strong connectivity that we described earlier, the edge directionality is taken into consideration, while a weakly connected component requires the existence of a path between every pair of nodes in the maximal subgraph without considering edge directionality.

SUBGRAPH AND INDUCED SUBGRAPH. A graph $H = (V_H, E_H)$ is a *subgraph* of $G = (V_G, E_G)$, denoted by $H \subseteq G$, if it is a subset of its edges and all their endpoints, i.e., $E_H \subseteq E_G$ and $V_H = \{u, v : (u, v) \in E_H\}$. A subgraph $H = (V_H, E_H)$ of $G = (V_G, E_G)$ is called *induced*, if it is a subset of its nodes $V_H \subseteq V_G$ and contains all the edges between the nodes of V_H, i.e., $E_H = \{(u, v) \in E_G : u, v \in V_H\}$.

COMPLETE GRAPH, CLIQUE AND TRIANGLE. A graph $G = (V, E)$ is called *complete*, if every pair of distinct nodes is connected by a unique edge. The complete graph of n nodes has $m = \binom{n}{2} = \frac{n(n-1)}{2}$ edges. A *clique* is defined as a subset of the nodes of a graph, such that its induced subgraph is complete. A *triangle subgraph* is a clique of three node, i.e., a triplet of connected nodes $(u, v, w) : (u, v), (v, w), (w, u) \in E$.

2.2 LINEAR ALGEBRA: EIGENVALUES AND SINGULAR VALUES

As we discussed earlier, every graph can be represented by a matrix, the so-called *adjacency matrix*. The adjacency matrix $\mathbf{A}$ of a graph $G = (V, E)$ is

the $|V| \times |V|$ matrix with elements $A_{ij} = 1$ if there exist an edge between nodes i, j in the graph. In the general case where the edges of the graph contain weights, the entries of the weighted adjacency matrix correspond to edge weights. For undirected graphs, the adjacency matrix $\mathbf{A}$ is symmetric (i.e., $\mathbf{A} = \mathbf{A}^T$), while for directed graphs the matrix is nonsymmetric.

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be a symmetric matrix. A vector $\mathbf{u}$ is defined as *eigenvector* of $\mathbf{A}$ if and only if $\mathbf{Au} = \lambda\mathbf{u}$, where λ is a scalar called *eigenvalue* corresponding to $\mathbf{u}$. Then, $\mathbf{A}$ can be written as $\mathbf{A} = \mathbf{U\Lambda U}^T$, where the orthogonal matrix $\mathbf{U}$ contains as columns the eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ of $\mathbf{A}$, correspond to real eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ and $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ the diagonal matrix with the eigenvalues as entries [GL96; Str88; Mie11]. The eigenvalues of the adjacency matrix define the *spectrum* of a graph and have close connections with several important graph properties.

As we stated above, in the case of directed graphs the corresponding adjacency matrix is nonsymmetric and therefore the eigenvalues can be complex. Thus, it is preferable to work with the singular values of the matrix which can be extracted by the *singular value decomposition* (SVD). That is, the SVD of a real matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ is defined as $\mathbf{A} = \mathbf{U\Sigma V}^T$, where $\mathbf{U} \in \mathbb{R}^{m \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times n}$ contain the left-singular and right-singular vectors respectively and $\mathbf{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_p) \in \mathbb{R}^{m \times n}$, $p = \min\{m, n\}$, the diagonal matrix comprised of singular values σ_i (note that, for symmetric matrices, the singular values correspond to the absolute values of the eigenvalues).

2.3 CORE DECOMPOSITION IN NETWORKS

In this section, we describe in detail the concept of *k-core decomposition* in networks [Sei83] – a central topic of this dissertation. Initially, we present the basic concepts, algorithms and extensions of the k-core decomposition; then, we refer to some key applications in several domains.

2.3.1 *Fundamental Concepts and Algorithms*

Informally, the k-core decomposition is a threshold-based hierarchical decomposition of a graph into nested subgraphs. The basic idea is that a threshold k is set on the degree of each node; nodes that do not satisfy the threshold, are excluded from the process. More precisely, let $G = (V, E)$ be an undirected graph and let k be a non-negative integer.

Definition 2.4 (k-core Subgraph)**.** *Let H be a subgraph of G, i.e., $H \subseteq G$. Subgraph H is defined to be a k-core of G, denoted by G_k, if it is a maximal connected subgraph of G in which all nodes have degree at least k.*

Definition 2.5 (Graph Degeneracy $\delta^*(G)$)**.** *The degeneracy $\delta^*(G)$ of a graph G is defined as the maximum k for which graph G contains a non-empty k-core subgraph.*

Definition 2.6 (Node's Core Number)**.** *A node i has core number $c_i = k$, if it belongs to a k-core but not to any $(k+1)$-core.*

Definition 2.7 (k-shell Subgraph)**.** *The k-shell is the subgraph defined by the nodes that belong to the k-core but not to the $(k+1)$-core.*

Based on the above definitions, it is evident that if all the nodes of the graph have degree at least one, i.e., $d_v \geq 1, \forall v \in V$, then the 1-core subgraph corresponds to the whole graph, i.e., $G_1 \equiv G$. Furthermore, assuming that $G_i,\ i = 0, 1, 2, \ldots, \delta^*(G)$ is the i-core of G, then the k-core subgraphs are nested, i.e.,

$$G_0 \supseteq G_1 \supseteq G_2 \supseteq \ldots \supseteq G_{\delta^*(G)}.$$

Typically, subgraph $G_{\delta^*(G)}$ is called *maximal k-core subgraph* of G. Figure 2.2 depicts an example of a graph and the corresponding k-core decomposition. As we can observe, the degeneracy of this graph is $\delta^*(G) = 3$; thus, the decomposition creates three nested k-core subgraphs, with the 3-core being the maximal one. The nested structure of the k-core subgraphs is indicated by the dashed lines on the plot. Furthermore, the color on the nodes suggest the core number c of each node. Lastly, we should note here that the k-core subgraphs are not necessarily connected.

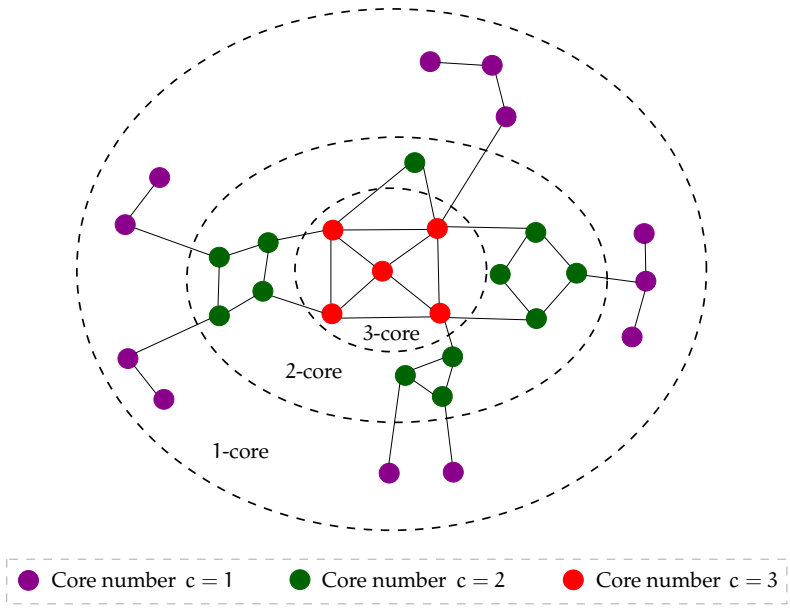


Figure 2.2: Example of the k-core decomposition.

The concept of degeneracy in graphs, as defined above, is also known as *width* [Fre82] and *linkage* [KT96]. It is also related to the *coloring number* α of a graph [EH66], which is defined to be the least k for which there is an ordering $\prec$ of the nodes of the graph such that for every $v \in V$, the number of adjacent nodes $w \prec v$ is less than α.

Computing the k-core decomposition of a graph can be done through a simple process that is based on the following property: to extract the k-core subgraph, all nodes with degree less than k and their adjacent edges should be recursively deleted [Sei83]. That way, beginning with $k = 0$, the algorithm removes all the nodes (and the incident edges) with degree equal or less than k, until no such nodes have been remained in the graph. Also notice that, removing edges that are incident to a node may cause reductions to the degree of neighboring nodes; the degree of some nodes may become at most k, and thus, they should also be removed at this step of the algorithm. When all remaining nodes have degree $d_v > k$, k is increased by one and the process is repeated until no more remaining nodes have left in the graph. Since each node and edge is removed exactly once, the running time of the algorithm is $\mathcal{O}(|V| + |E|)$ [MB83]. Batagelj and Zaveršnik [BZ03] proposed an $\mathcal{O}(|E|)$

Algorithm 2.1 k-core decomposition

Input: Undirected graph $G = (V, E)$
Output: Vector of core numbers $c_i, \forall i \in V$

```
Compute the degrees of each node d_v, ∀v ∈ V
Order the set of nodes v ∈ V in increasing order of their degrees d_v
for each v ∈ V do
    c_v ← d_v
    for each w ∈ Nb(v) do
        if d_w > d_v then
            d_w ← d_w − 1
            Reorder node set V accordingly
        end if
    end for
end for
return Core numbers c_i, ∀i ∈ V
```

algorithm for k-core decomposition, which is presented in Algorithm 2.1. Note that, maintaining the k-core decomposition of a graph is equivalent of keeping the core number $c_i, \forall i \in V$.

LARGE SCALE k-CORE DECOMPOSITION. As we will present shortly, k-core decomposition is a tool that has been widely applied in many network analytics tasks that involve large scale data. Due to its success, several extensions have been proposed for large scale k-core decomposition under various computation frameworks. In Ref. [MPM13], a distributed k-core decomposition algorithm was proposed, while in Ref. [PKT14] an implementation for the MAPREDUCE framework [DG04]. The authors of Ref. [Sar+13] proposed an efficient incremental k-core decomposition algorithm for streaming graph data. That way, as new nodes and edges are inserted into or removed from the graph in an online manner, the algorithm is able to efficiently update the decomposition, without recomputing it from the beginning. Cheng et al. [Che+11] proposed an external-memory algorithm for k-core decomposition of disk-resident massive graphs. Finally, in Ref. [OS14], a local method for estimating the core number of each node was proposed, using only information about the neighborhood of the node. That way, by tuning a parameter that controls the size of the region of interest around a query node,

we are able to balance between accuracy and computational complexity on the computation.

EXTENSIONS TO OTHER TYPES OF GRAPHS. The k-core decomposition described above considers that graphs are unweighted and undirected. However, many real-world networks carry rich semantics, as expressed by more complex graph types. To that end, there is research effort towards meaningful extensions of the k-core decomposition to other types of graphs. In Refs. [GTV11b; GSH12; EA13], extensions of the decomposition were proposed for the case of *weighted graphs*. Giatsidis et al. [GTV11a] introduced D-cores, an extension of the k-core structure to *directed graphs*. In this case, the notion of (k, ℓ)-core is used to represent subgraphs in which all nodes have in-degree at least k and out-degree at least ℓ respectively. In Ref. [Gia+14b], an extension of the k-core decomposition for *signed networks* was proposed. Signed networks [KLB09; Kun+10] are used to capture the notion of positive and negative interactions among the nodes of a graph (e.g., trust/distrust, friend/foe relationships). Examples of such networks include the trust networks that can be produced by product review websites like Epinions (`www.epinions.com`) and the voting election network between the administrators of Wikipedia (`www.wikipedia.org`). Finally, Bonchi et al. [Bon+14] proposed a core decomposition of *uncertain* or *probabilistic* graphs. Those kind of graphs are used to capture uncertainty, which can be introduced under several conditions, such as for privacy-preserving reasons. In this case, every edge in the graph is associated with the probability of existence.

GENERALIZED CORES. The k-core decomposition was initially introduced for the degree property of the nodes in a graph. Batagelj and Zaveršnik proposed the notion of *generalized cores* [BZ02], which extends cores from degree to other node properties.

Definition 2.8 (Generalized Cores or p-cores)**.** *Let $G = (V, E)$ be a graph and let $w : E \to \mathbb{R}$ be a function assigning values (or weights) to the edges of the graph. A node property function p that assigns real values on graph G, is defined as $p(v, C)$, where $v \in V$ and $C \subseteq V$. Then, a subgraph $H = (V_H, E_H)$ induced by the set*

$V_H \subseteq V$ is called a p-core at level $t \in \mathbb{R}$ if and only if (i) $\forall v \in V_H : t \leq p(v, V_H)$ and (ii) V_H is a maximal set.

Recall that, a function p is called *monotone* if and only if:

$$C_1 \subseteq C_2 \Rightarrow p(v, C_1) \leq p(v, C_2), \forall v \in V. \tag{2.1}$$

In Ref. [BZ02] it was shown that for a monotone function p, the p-core at level t of the decomposition can be found be successively removing nodes with value of p less than t – as has been already described for the k-core decomposition. Furthermore, the subgraphs correspond to the cores are nested, i.e., $t_1 < t_2 \Rightarrow H_{t_2} \subseteq H_{t_2}$. In fact, if we consider that function p corresponds to the degree of a node, i.e., $p(v, C) = d_v^C$, where d_v^C is the degree of node v in subgraph C, this function is monotone. Also, many other functions on the nodes v of the graph including the in-degree and out-degree, the weighted degree (i.e., sum of weights of the adjacent edges) and the number of cycles of length k that pass through node v, have been proven to be monotone; thus, the same procedure can be used to extract the corresponding p-cores.

EXTENSIONS OF THE DECOMPOSITION. In the related literature, there is recent research effort to introduce decompositions similar to those performed by the k-core one, but more meaningful in real applications. An example of such extension is the *K-truss decomposition* [Coh08; WC12] and triangle k-core decomposition [ZP12] (in fact, both decompositions are equivalent). The main idea is to define the notion of cores on the edges of a graph, by taking into account the number of triangles that each edge participates to – thus being able to extract more coherent subgraphs. This decomposition will be used in Chapter 5. Tatti and Gionis [TG15] introduced the notion of *density-friendly graph decompositions*, motivated by the fact that the maximal k-core subgraph is not necessarily a good approximation of the densest one. Since such decompositions are typically used as heuristics to detect dense subgraphs, being able to have guarantees about the density of the extracted subgraph is crucial in many real applications. Finally, Sariyüce et al. [Sar+15] proposed the *nucleus decomposition* – a generalization of the

k-core and *K*-truss decompositions. It represents the graph as a forest of nuclei, i.e., hierarchical structures in which smaller cliques are present into larger ones with the edge density being improved as we move towards the leaf nuclei.

2.3.2 *Applications*

The *k*-core decomposition and its extensions have been extensively used in several applications. Seidman [Sei83] was the first that proposed to use the tool of *k*-core decomposition in social network analysis, as an easy to compute and effective way to extract dense subgraphs. Later, other studies in large scale real networks followed [Hea+08; GTV11b], including the analysis of Microsoft Instant Messenger (MSN) [LH08] and Facebook [Uga+11] social graphs. In a similar way, the decomposition has been applied to study [AH+08], model [Car+07] and examine the evolution of the Internet graph [Zha+08]. Several theoretical studies about the structure of real networks and of the corresponding generative models have been presented from the *statistical physics* community [DGM06; HD+13]. Furthermore, the decomposition has been used as a visualization tool for large graphs [AhBV06; ZP12].

A common application of the *k*-core decomposition is the identification of dense subgraphs; Andersen and Chellapilla [AC09] were based on this to propose solutions with approximations guarantees for variants of the densest subgraph problem. In a similar spirit, variants of the community detection problem has been addressed utilizing the properties of *k*-core decomposition, including local community detection techniques [Cui+14] and the influential community search problem [Li+15] where the notion of influence is defined as the minimum weight of the nodes in that community.

In our work on community detection [Gia+14a], we built upon the properties of the decomposition to speed-up the execution time of computational intensive graph clustering algorithms, such as the one of spectral clustering. In particular, we proposed CoreCluster, an efficient graph clustering framework that can be used along with any known graph clustering algorithm. Our approach capitalizes on processing the graph in a hierarchical way provided by its *k*-core decomposition. That way, the nodes are clustered in

an incremental manner that preserve the clustering structure of the graph, while making the execution of the chosen clustering algorithm much faster due to the smaller size of the graph's partitions onto which the algorithm operates.

Recently, the concept of k-core decomposition has been also applied in information networks used to represent textual information, and more precisely, for the task of keyword extraction in documents [RV15; LCC14]. As we will present in Chapter 6, we also utilize the properties of the k-core decomposition in the text categorization domain, as a graph-based process to extract discriminative features for the classification task.

2.4 DESCRIPTION OF GRAPH DATASETS

In this section, we briefly describe the graph datasets used in this thesis (the datasets used for the graph-based text categorization task are described in Chapter 6). We have considered data from different domains, including social, collaboration, information and technological networks. In all cases, networks are treated as undirected, keeping only the largest connected component. Almost all datasets are publicly available [LK14] (we have constructed the DBLP dataset). Table 2.2 provides a summary of the network statistics.

FACEBOOK. This network was created from Facebook online social networking platform (`www.facebook.com`), focusing on the region of New Orleans, USA. The nodes of the graph correspond to users and the edges indicate friendship relationships [Vis+09].

YOUTUBE. This network was formed by the Youtube (`www.youtube.com`) video sharing platform. In the Youtube's social network, users establish friendship relationships via comments' exchange about videos [Mis+07].

SLASHDOT. Slashdot (`slashdot.org`) is a technology news website. The nodes of the social network that is created correspond to users and the edges capture friendship relationships among them. In fact, users are able to tag other users as friends or foes, forming a signed social network with positive

and negative types of edges. In our experiments, we do not take into account the type of the edges [Les+09].

EPINIONS. This is a trust-based (who-trusts-whom) online social network between the members of the Epinions.com (www.epinions.com) product review website. The nodes of the network correspond to users of the website and the edges capture trust relationships between them. Although the network is signed, in our experiments we discard this information [RAD03].

WIKI-VOTE. The graph was created from the online encyclopedia Wikipedia (www.wikipedia.org) and more precisely from the elections conducted to promote users to administrators. The nodes of the social network correspond ot Wikipedia users and an edge between users i, j denotes that user i voted for user j [LHK10b; LHK10a].

WIKI-TALK. The users that have been registered as contributors to Wikipedia, have their own talk page, that other users can edit in order to communicate and discuss about updates on Wikipedia articles. The nodes of this communication network correspond to users of Wikipedia and an edge from node i to node j indicates that user i edited a talk page of user j at least once [LHK10b; LHK10a].

EMAIL-EUALL AND EMAIL-ENRON. Both datasets correspond to graphs that have been created by email interaction between the members of large European research institution (EMAIL-EUALL) and Enron Corporation (ENRON). In both graphs, each node corresponds to an email address, and an edge is added between nodes i, j if i sent at least one email to j [LKF07; Les+09].

CA-GR-QC, CA-ASTRO-PH, CA-HEP-PH, CA-HEP-TH AND CA-COND-MAT. All these graphs represent co-authorship collaboration between researchers on the following fields of Physics: General Relativity and Quantum Cosmology, Astrophysics, High Energy Physics - Phenomenology, High Energy Physics - Theory, Condense Matter Physics. They have been constructed from the arxiv.org repository of electronic preprints (arxiv.org). The nodes of the

graphs correspond to researchers, while an edge between nodes i, j indicate that author i co-authored a paper with author j [LKF07].

DBLP. The dataset has the same semantics as the previous ones and it corresponds to co-authorship relationships in the broader field of Computer Science, extracted for the DBLP computer science bibliography website (`http://dblp.uni-trier.de`). We have created the co-authorship graph based on the xml file of the scientific publications published on the DBLP website [Dbl].

CAIDA AND OREGON. Those two technological networks capture communication information between different routers of the Internet. The graph of routers can be organized into subgraphs, called Autonomous Systems, that can exchange traffic flows between them, forming a communication network [LKF05].

Network	$\|V\|$	$\|E\|$	Description
FACEBOOK	63,392	816,886	Facebook New Orleans social network
YOUTUBE	1,134,890	2,987,624	Social network from Youtube
SLASHDOT	82,168	582,533	Slashdot social network (Feb. '09)
EPINIONS	75,877	405,739	Who trusts whom network
WIKI-VOTE	7,066	100,736	Elections of Wikipedia administrators
WIKI-TALK	2,388,953	4,656,682	User communication in Wikipedia
EMAIL-EUALL	224,832	340,795	E-mail communication network
EMAIL-ENRON	33,696	180,811	E-mail communication network
CA-GR-QC	4,158	13,428	Co-authorship network in General Relativity
CA-ASTRO-PH	17,903	197,031	Co-authorship network in Astrophysics
CA-HEP-PH	11,204	117,649	Co-authorship network in High Energy Physics
CA-HEP-TH	8,638	24,827	Co-authorship network in High Energy Physics
CA-COND-MAT	21,363	91,342	Co-authorship network in Condensed Matter Physics
DBLP	404,892	1,422,263	Co-authorship network from DBLP
CAIDA/OREGON [LKF07]	26,475/11,461	106,762/32,730	Autonomous systems graphs

Table 2.2: Networks datasets used in the thesis, along with basic statistics: nimber of nodes $|V|$; number of edges $|E|$.

3

MODELING ENGAGEMENT DYNAMICS IN SOCIAL GRAPHS

Given a large social graph, how can we model the engagement properties of nodes? Can we quantify engagement both at node level as well as at graph level? The property of engagement in social networks refers to the degree that an individual participates in the activities of a community; it is closely related to the departure dynamics of users, i.e., the tendency of individuals to leave the community. In this Chapter, building upon recent work in the field of game theory, we propose models of user engagement based on the properties of k-core decomposition of the underlying social graph. After modeling and defining the engagement dynamics at node and graph level, we examine whether they depend on structural and topological features of the graph. We perform experiments on a multitude of real graphs, observing interesting connections with other graph characteristics, as well as a clear deviation from the corresponding behavior of random graphs. Furthermore, similar to the well known results about the robustness of real graphs under random and targeted node removals, we discuss the implications of our findings on a special case of robustness – regarding random and targeted node departures based on their engagement level.

3.1 INTRODUCTION

Over the last years, there is a considerable interest on studying the properties and dynamics of social networks, arising from a plethora of online social networking and social media applications, such as Facebook, Google+ and Youtube. Typically, users become members of an online community for several reasons (e.g., create new friendship relationships and use of applications offered by a platform) and a lot of research effort has been devoted to under-

stand the dynamics of formation and evolution of those social communities. Characteristic example is the observation that individuals decide to join a community based not only on the number of friends that are already part of the community, but also on the degree of interactions among these friends [Bac+06].

However, similar to the decision of becoming member of a community, an individual may also decide to leave the network. Although in many of the popular social networking applications typically users do not explicitly leave the network, the decision of departure can be expressed by inactivity, i.e., the user do not participate in the activities of the community. Can we model and quantify the departure dynamics of individuals in a social graph?

In this Chapter, we are trying to answer the above question studying the property of *user engagement* in social interaction graphs. Typically, user engagement refers to the extend that an individual is encouraged to participate in the activities of community[1]. In the areas of sociology and economics, the problem of social engagement examines the engagement of individuals to products, services or ideas. Similarly, in the field of web mining, the property of engagement refers to the quality of the user experience, as expressed by the duration and frequency that a web application is used [BYL12]. In the context of a social graph, the property of engagement captures the *incentive* of a user (node) to remain engaged. In other words, the property of node engagement can be considered as complementary to the one of node departure.

Typically, an individual decides to remain engaged in the community (instead of depart), based on the benefit that is derived by the participation. Intuitively, the benefit of a user is based on its neighborhood, i.e., the number of friends that are also part of the community. Furthermore, as mentioned earlier, the strength of interactions among the friends of a user, is also a crucial factor for being part of the community. Therefore, it becomes clear that the decision of a user to remain engaged is affected by the structure of its neighborhood. Suppose now that a user decides to dropout, due to its low incentive of being part of the community. This decision is possible

1 Wikipedia's lemma for *Social Engagement:* http://en.wikipedia.org/wiki/Social_engagement.

to affect the engagement level of his neighbors, that potentially can depart as well. This effect can evolve in a *contagion* within the graph, leading to a *cascade* of node departures.

In this work, we model and study the engagement properties of real-world social graphs. Our approach capitalizes on recent results in the field of game theory, where the engagement property can be considered in a similar manner as a product adoption process [MJ09; Har13; Bha+11]. In the case where individuals decide simultaneously whether to remain engaged or depart from the graph, the engagement level of each node can be captured by the properties of the k-core decomposition [Sei83]. Based on this point, we propose measures for characterizing the engagement at both *node level* as well as at *graph level*. We examine in detail the properties of a large number of real graphs, trying to better understand the engagement dynamics.

The main contributions of this work can be summarized as follows:

- *Problem statement:* We study the property of engagement in social graphs and how it can be used to model the departure dynamics of nodes in the graph.
- *Measures of engagement:* Based on game theoretic models, we propose interesting measures for characterizing the engagement at both node level as well as at graph level.
- *Experiments on real graphs:* We perform a large number of experiments in several real-world graphs, examining the engagement dynamics and observing interesting properties.
- *Implications of our study:* We discuss the implications of our study regarding a new problem of robustness/vulnerability assessment in real graphs, where nodes decide to leave the graph based on their own incentives.

The rest of the Chapter is organized as follows. Section 3.2 gives the related work and Section 3.3 provides the necessary background. Then, in Section 3.4 we describe the model and the proposed engagement measures. Section 3.5 presents the experimental evaluation of our method, while in

Section 3.6 interesting implications of our study are discussed. Finally, we conclude in Section 3.7.

3.2 RELATED WORK

In this section we review the related work, regarding the engagement dynamics in social graphs, as well as other applications of the k-core decomposition. In the very recent literature, there has been presented some game-theoretic models for the problem of product adoption in networked environments [MJ09; Har13; Bha+11]. These models form the basis of our approach and are described in detail in Section 3.4. To the best of our knowledge, the only related work that provides experimental study for the problem of node departures in social networks, is the work presented in [Wu+13]. There, the authors study two real social networks and examine whether the departure dynamics show similar behavior with the arrival dynamics. In the case of node departures, the authors of [Wu+13] observed that the active users typically belong to a dense core of the graph, while inactive users are placed on the sparsely connected periphery of the graph. As we will present later, the property of node engagement can be considered complementary to the one of node departure, and our approach provides a more refined modeling of the observations made in [Wu+13]. Moreover, related to our work can be considered recent studies about the formation dynamics of communities [Bac+06; LS09], as well as studies about diffusion and contagion in social graphs [Uga+12; AKM08; RMK11; EK10].

As we will present in Section 3.4, our method builds upon the properties of the k-core decomposition in a graph (see Section 3.3 for more details). Broadly speaking, the k-core decomposition has been applied in the past for extracting the most coherent subgraphs [Sei83], graph visualization [ZP12], identification of influential spreaders [Kit+10], and for studying [AH+08] and modeling [Car+07] the Internet topology. In this work, we examine one more application domain of the k-core decomposition in the problem of node engagement in social graphs; in contrast to the previous studies that mostly focus on the nodes of the best k-core subgraph, in this work we

are interested in the hierarchy produced by the decomposition, since it can provide meaningful insights about the engagement dynamics of the graph.

3.3 PRELIMINARIES AND BACKGROUND

In this section we briefly discuss the properties of the k-core decomposition [Sei83], which is utilized by our method. For completeness in the presentation, next we repeat the definitions regarding the main concepts of the k-core decomposition used in this Chapter (see also Chapter 2, Section 2.3 for a detailed description). Let $G = (V, E)$ be an undirected graph, where $|V| = n$, $|E| = m$ and let graph H be a subgraph of G.

A subgraph H is defined to be a *k-core* of G if it is a maximal connected subgraph of G, in which all nodes have degree at least k. The *degeneracy* $\delta^*(G)$ of a graph G is defined as the maximum k for which graph G contains a non-empty k-core subgraph. A node i has *core number* $c_i = k$, if it belongs to a k-core but not to any $(k+1)$-core.

As we have already described, the k-core of a graph G can be obtained by repeatedly deleting all nodes with degree less than k. Furthermore, the k-core decomposition – which assigns a core number c_i to each node $i \in V$ – can be computed efficiently, with complexity $\mathcal{O}(m)$ proportional to the size of the graph [BZ03]. The most important point is that the k-core decomposition creates an hierarchy of the graph, where "better" k-core subgraphs (i.e., higher values of k) correspond to more cohesive parts of the graph.

3.4 PROBLEM FORMULATION AND PROPOSED METHOD

In this section, we formulate the problem of modeling and quantifying the engagement dynamics in a social interaction graph. We begin by discussing the main factors that intuitively affect the decision of nodes to remain engaged or leave the graph. Then, we present the theoretical model used to approximate and capture the engagement behavior of nodes, as well as the proposed engagement measures at both node and graph level.

3.4.1 *Problem Statement and Model Description*

Our goal is to model and study the problem of node engagement in social graphs. Informally, the property of engagement captures the incentive of individuals to remain engaged in the graph, as opposed to their decision of departure. In the context of this work, we are interested in the engagement dynamics of individuals as well as of the whole system, from a *network-wise* point of view. In other words, we consider only the underlying graph structure of a social system, and based on its properties we derive measures that characterize the behavior in terms of engagement.

Typically, each individual that participates in a social activity – as expressed by his/her participation in a social graph – derive a benefit. In most of the cases, this benefit emanates from his/her neighborhood, as captured by the node degree in the social graph. Furthermore, one additional factor that affects the benefit of each individual is the degree of interaction among its neighbors [Uga+12], in the sense that if one's friends tend to highly interact among each other, the benefit of remaining engaged in the graph could potentially be increased.

Let us now suppose that a user decides to drop out from his community due to the fact that the incentive of staying has been reduced. This decision will cause direct effects in his neighborhood, in the sense that some of his friends may also decide to depart. More precisely, a departure can become an *epidemic* (or contagion), forming a *cascade* of individual departures; nodes will decide to leave and this will also affect not only their neighbors but also the whole community. Therefore, according to the notion of *direct-benefit effects*, individuals who want to incur an explicit benefit by remaining engaged, they should align their decision with the one of their neighbors [EK10].

Next, we present our model and the proposed measures for engagement in social graphs. Each node $v \in V$ – that corresponds to an individual – can either remain engaged in the network or can decide to depart. As we mentioned earlier, it is natural that the decision of each node should be based on the decisions of its neighbors. The behavior of nodes as a system can be expressed using game-theoretic concepts, and more precisely it can be captured by the notion of *networked coordination games* [EK10]. That is, the

property of engagement can be viewed as a network model based on *direct-benefit* effects: the node's benefit of remaining engaged in the graph increases as more neighbors decide respectively to stay in the graph. This formulation has been extensively studied in the areas of game theory and economics. It is applied in situations where the nodes have to choose between two possible alternatives and the structure of the underlying social network affects the decision: for two neighborhood nodes u and v, there is an incentive to be aligned with the same decision, since that way they will both increase their benefits produced by the underlying interactions.

In a similar way, since the benefit of each node for staying in the network emanates from its neighbors, the problem can be modeled using similar concepts with the ones of coordination games. We consider that the nodes of the graph – which correspond to rational individuals/players – decide *simultaneously* whether to stay or leave. Each node $i \in V$ has the same set of possible strategies $\mathcal{X} = \{0, 1\}$, i.e., *leave* or *stay* in our case. Let $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ be the vector that denotes the decision of each node. The *payoff* (or utility) of a node i given the behavior of the rest nodes (as captured by vector $\mathbf{x}$), can be expressed as:

$$\Pi_i(\mathbf{x}) = \textbf{benefit}\Big(x_i, \sum_{j \in \mathcal{N}_i} x_j\Big) - \textbf{cost}(x_i), \tag{3.1}$$

where $\textbf{benefit}(\cdot)$ and $\textbf{cost}(\cdot)$ are the node's benefit and cost functions respectively and $\mathcal{N}_i = \{j \in V : (i, j) \in E\}$ is the neighborhood set of node i. In other words, the benefit of each node depends on its own decision x_i and the aggregate decisions of its neighbors; this captures at a large extend the problem of engagement estimation, since in many cases a user remains engaged according to the degree of interactions with its friends in the community. Furthermore, every node i incurs a cost for remaining engaged in the graph, which depends only on its own action. While the actual form of the cost function does not need to be apriori known in the model, it is clear that a node will decide to stay engaged if its cost is not higher than its benefit. Let $\textbf{cost}(x_i) = k$ be the cost value of each node $i \in V$. Then, according to Eq. (3.1), every node that will remain engaged should have non negative payoff, and therefore $\Pi_i(\mathbf{x}) = |\mathcal{N}_i^{x=1}| - k$, where $|\mathcal{N}_i^{x=1}|$ is the number of

i's neighbors that finally remain engaged, i.e., the degree of i in the graph induced by nodes with decision $x = 1$.

Thanks to some very recent results in the area of game theory, the equilibrium of this game corresponds to the core number c_i of each node, as produced by the k-core decomposition [MJo9; Har13; Bha+11].

Proposition 3.1 (Equilibrium Property, [MJo9; Har13])**.** *The best response (Nash equilibrium) of each node $i \in V$ in the model presented above corresponds to the core number c_i.*

In other words, in the case of equilibrium, every node in the induced subgraph S formed by nodes with $x_i = 1$ should have minimum degree k, satisfying the property of $c_i \geq k$. That way, no engaged node will have incentive to depart from S and no node outside S, i.e., in $V - S$, will have at least k neighbors in S in order to remain engaged after his departure. As noted by the authors in [MJo9; Har13; Bha+11], the game has multiple equilibria, but the maximum one corresponds to the "best" k-core structure of the graph (i.e., $k = \delta^*(G)$). In our case, we are interested in all nodes in the graph and not only on those that form the best equilibrium; as we will present shortly these nodes show interesting properties.

3.4.2 *Engagement Measures*

Having presented the basic theoretical model, we now proceed with the proposed measures for characterizing the engagement dynamics on graphs. We are interested in studying the engagement properties at both *node* (local) and *graph* (global) level; furthermore, we are interested in examining the behavior of specific subgraphs – as produced by the k-core decomposition – which include nodes with specific engagement level.

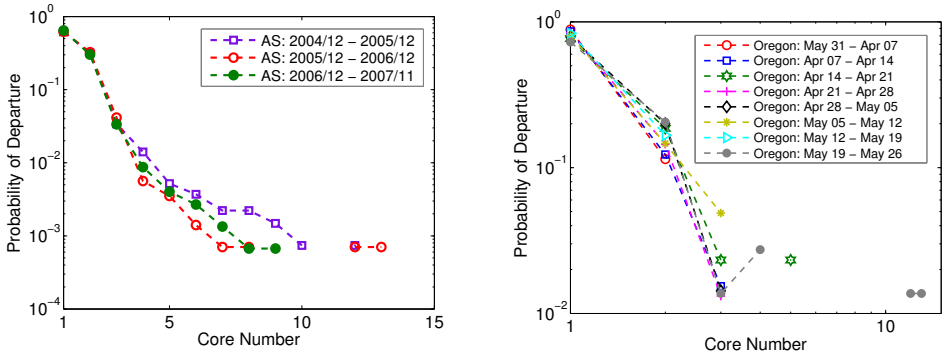
Capitalizing on Proposition 3.1, we quantify the property of node engagement using the k-core decomposition, and more specifically the core number c_i of each node $i \in V$.

Proposition 3.2 (Node Engagement)**.** *The engagement level e_i of each node $i \in V$ is defined as its core number c_i.*

Typically, nodes that belong to higher cores of the graph (higher core number), show better engagement and therefore it is less probable to depart (or, at least, the incentive to depart is lower). As we discussed in Section 3.4.1, the core number of each node is a reasonable metric (or estimator) to capture and model the engagement dynamics: nodes remain engaged if their neighbors (and the neighbors of their neighbors, etc.) also remain engaged. On the other hand, if a node decides to depart, this may affect the engagement level of its neighbors – which may decide to leave as well – forming a cascade of potential departures in the graph. This dynamic effect of cascades is naturally captured by the k-core decomposition and the core number of nodes.

Figure 3.1 provides some empirical observations regarding the departure of nodes – on data with available relevant information – that can be used to support our modeling approach. As it is difficult to access social graph data where nodes (users) explicitly define their departure time, we have examined two snapshots of the Internet topology (CAIDA and OREGON Autonomous Systems) with available dropout information. Figure 3.1 depicts the probability of departure vs. the core number c_i of a node. As it can be observed, nodes that belong to smaller cores (close to the first core) are more probable to leave the graph, thus supporting our modeling approach. Moreover, this property is persistent for several time snapshots of the graphs.

Additionally, the proposed engagement metric can be considered as a more refined modeling explanation of the departure dynamics in social graphs, as very recently observed in [Wu+13]. The authors of Ref. [Wu+13] studied the behavior of user departures in social networks and based on some inactivity criteria (e.g., in a co-authorship network, a user is considered inactive if he/she has not published a paper in a time period of more than five years), they observed that nodes which belong to the densely connected core of the graph mainly correspond to active users. On the other hand, inactive users (i.e., users that potentially have left the graph) belong to the periphery of the graph. In other words, the departure of nodes is proportional to their position in the graph, with nodes in the fringe of the graph presenting higher probability to dropout. Our modeling approach and the node engagement



(a) CAIDA (b) OREGON

Figure 3.1: Probability of departure vs. core number. The discontinuities in the plot correspond to zero (we plot only the cores from which nodes depart).

metric e – based on the properties of the k-core decomposition – quantify in a precise manner the above structural observations.

We should note that, here we examine the dynamics of engagement (and thus of departure) by a simple model and metric, that approximates real settings and observations in a concise manner. However, we do not argue that the engagement of a user is solely proportional to his core number; other external factors may affect its behavior as well. Nevertheless, in the rather realistic case where each node decides to remain engaged for maximizing its revenue by the participation in the community – thus considering the decision of its neighbors – the behavior can be modeled by the proposed metric.

Furthermore, as we will see in the experimental results, the degree of a node is not an accurate estimator of the departure dynamics: while high degree is necessary for achieving higher engagement and higher core number, the opposite is not always true. In many cases, high degree nodes have low core number because of the fact that their neighbors are not well connected among each other[2]. Therefore, the engagement should be described by a metric able to capture both the size of node's neighborhood as well as its connectivity. In Section 3.5 where the experimental results are presented, we

2 A similar behavior has been reported in [AH+08] in the context of Internet graph analysis.

also examine how other well-known structural characteristics of the graph (e.g., degree, triangle participation ratio, clustering coefficient) affect the engagement behavior.

Based on this model of user behavior, we also propose to study the characteristics of the subgraphs produced in the case of simple scenarios, where nodes with certain engagement index k (for various values of k) decide simultaneously to drop out. The subgraph that remains after such types of departures is defined as the k-engagement subgraph $\mathcal{G}_k$.

Definition 3.3 (k-Engagement Subgraph $\mathcal{G}_k$)**.** *Let k be a integer parameter such that a node remains engaged in G if at least k neighbors are engaged. The graph $\mathcal{G}_k$ which is induced by nodes $i \in V$ with engagement level $e_i \geq k$ is defined as the k-engagement subgraph.*

The k-engagement subgraphs correspond to interesting structures of the graph. Actually, for a specific value of k, subgraph $\mathcal{G}_k$ represents the remaining graph, after the cascading effect where nodes with engagement lower than k have left the graph. The properties of the remaining subgraph – as captured by $\mathcal{G}_k$ – are crucial towards a better understanding of the engagement characteristics, as well as for examining the functional operation of the remaining graph. As we will present shortly, the size distribution of $\mathcal{G}_k$ for various values of k can inform us about the overall engagement level of the graph. Furthermore, it is interested to study whether well-known structural properties – such as the degree distribution of the graph – are retained after such types of nodes' departures. We also note that, following the properties of k-core decomposition, subgraphs $\mathcal{G}_k$ form a nested hierarchy $\mathcal{G}_0 \supseteq \mathcal{G}_1 \supseteq \mathcal{G}_2 \supseteq \ldots \supseteq \mathcal{G}_k$ for the possible values of k, in the sense that subgraphs of higher k also belong to $\mathcal{G}_k$'s of lower k.

Of particular interest is the subgraph $\mathcal{G}_k$ that corresponds to the maximum value of engagement e. In terms of k-core decomposition, the nodes with the highest engagement level $e_{\max}$ are those who belong to the best k-core of the graph, i.e., $k = \delta^*(G)$, where $\delta^*(G)$ is the degeneracy of the graph [Sei83].

Proposition 3.4 (Max-Engagement Subgraph)**.** *Let $k = \delta^*(G)$ be the degeneracy of the graph, i.e., the maximum k such that there exists a k-engagement subgraph. In*

our context, we consider this value as the maximum engagement level of the graph, i.e., $e_{\max} = \delta^*(G)$ *and we denote the Max-Engagement Subgraph as* $\mathcal{G}_{e_{\max}}$.

The Max-Engagement subgraph is composed by the nodes of the graph that show the highest engagement level $e = e_{\max}$. More precisely, each node $i \in \mathcal{G}_{e_{\max}}$ has degree $d_i \geq e_{\max}$ within $\mathcal{G}_{e_{\max}}$, implying that this set of nodes has potentially the lowest incentive to depart from the graph and thus it corresponds to the best engaged nodes. As we will discuss in Section 3.6, this subgraph also contains the most influential nodes of the graph, in terms of departure dynamics.

Having defined the engagement index of each node in the graph as well as the notion of the k-engagement subgraphs, it would be interesting to summarize this information into one value capable to describe the engagement level of the whole graph. That is, each individual node contributes to the engagement of the graph – according to its engagement index $e_i, \forall i \in V$ – based on the best core c_i that the node belongs to. Ideally, in terms of engagement, it would be better to have a large fraction of the nodes of the graph belonging to largest cores, thus showing higher engagement. In the extreme case of the graph with the *best engagement properties* – which corresponds to the complete graph $K_n = (V_{K_n}, E_{K_n})$ – all nodes belong to the Max-Engagement subgraph and their engagement index is equal to $e_i = |V_{K_n}| - 1, \forall i \in V_{K_n}$. In order to capture this behavior, we consider the *area under the curve* of the cumulative distribution of the k-engagement subgraphs' sizes. However, since the graphs do not have the same maximum engagement level $e_{\max}$, we normalize this value for each graph into the interval $[0, 1]$, based on a simple normalization: Normalized $e_j = \dfrac{e_j - \min(e)}{\max(e) - \min(e)}$, where $j = 1, \ldots, e_{\max}$, $\min(e) = 1$ and $\max(e) = e_{\max}$ (for simplicity, we consider that all nodes in the graph have degree at least one, therefore the minimum engagement is 1).

Proposition 3.5 (Graph Engagement)**.** *Let* $\mathcal{F}(e) = \Pr(X \geq e)$ *be the complementary cumulative distribution function of the sizes of the k-engagement subgraphs. Then, the total engagement level of a graph G, denoted as* $\mathcal{E}_G$*, is defined as the area under the curve of* $\mathcal{F}(e)$*,* $e = [0, 1]$*, i.e.,*

$$\mathcal{E}_G = \int_0^1 \mathcal{F}(e)\, de. \tag{3.2}$$

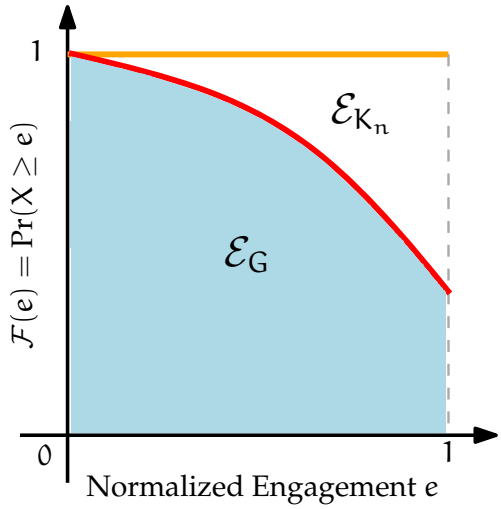


Figure 3.2: Schematic representation of the engagement index $\mathcal{E}_G$. The red curve shows an example of the complementary cumulative distribution function of the engagement level of the k-engagement subgraphs. The area under the curve (light blue region) captures the engagement properties of the graph. The orange colored curve shows the engagement level of the complete graph K_n.

Figure 3.2 depicts a schematic representation of the engagement index $\mathcal{E}_G$. The horizontal axis corresponds to the normalized engagement value e, while the vertical axis represents the probability $\Pr(X \geq e)$ that a node has (normalized) engagement level at least e (as produced by the sizes the k-engagement subgraphs). The values of $\mathcal{E}_G$ are in the range of $[0, 1]$, with higher values indicating graphs with higher total engagement level (larger area under red curve). In the case of the complete graph (orange colored curve), the probability that a node has normalized engagement at least $e, \forall e \in [0, 1]$, is $\Pr(X \geq e) = 1$. In other words, every node in the graph has engagement $e_i = |V_{K_n}| - 1$, and therefore the size of the k-engagement subgraphs, for $k = 1, \ldots, e_{\max}$, is equal to the size of the whole graph, i.e., $|V_{K_n}|$.

3.4.3 *Discussion*

Having presented the proposed engagement measures, we briefly discuss on an important point in the modeling approach followed by our method. Our approach and the proposed engagement measures are build upon the game presented in Section 3.4.1, which considers that nodes have *complete information* about the structure of the graph [EK10; MJ09]. Although this assumption may not be very accurate in many settings where individuals should take a decision (to remain engaged or to depart in our problem), we consider that in this case is valid since our goal is to model and to provide a high level study of the behavior of individual nodes and of the graph as a whole, regarding their engagement properties. Thus, our study builds upon the fact that we have knowledge of the structure of the graph. In a typical application scenario of our approach, the administrator of a social graph (e.g., FACEBOOK) – who has global knowledge of the structure of the graph – can use the proposed measures to examine the engagement dynamics of the graph and to potentially detect nodes that tend to leave, due to their low engagement.

3.5 ENGAGEMENT OF REAL GRAPHS

In this section we present detailed experimental results of the proposed engagement measures, at both local (node) and global (graph) level. The experiments were designed to address the following points:

P1: Study the characteristics of the engagement dynamics in real graphs.

P2: Examine how other graph features affect the engagement of the graph.

As we have already mentioned, we consider that the feasibility and applicability of our approach is supported by the results depicted in Fig. 3.1 and by very recent observations about the departure dynamics in social graphs [Wu+13]. Actually, here we present a more refined explanation of the departure dynamics, studying the complementary property of engagement. Furthermore, the time complexity of our approach is linear with

Network Name	Number of Nodes	Number of Edges
FACEBOOK	63,392	816,886
YOUTUBE	1,134,890	2,987,624
SLASHDOT	82,168	582,533
EPINIONS	75,877	405,739
EMAIL-EUALL	224,832	340,795
EMAIL-ENRON	33,696	180,811
CA-GR-QC	4,158	13,428
CA-ASTRO-PH	17,903	197,031
CA-HEP-PH	11,204	117,649
CA-HEP-TH	8,638	24,827
CA-COND-MAT	21,363	91,342
DBLP	404,892	1,422,263
CAIDA/OREGON	26,475/11,461	106,762/32,730

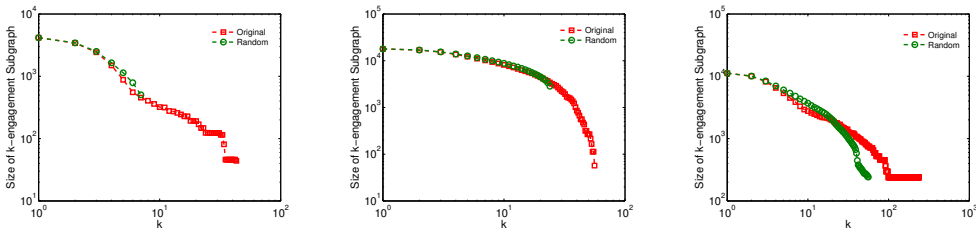
Table 3.1: Summary of real-world networks used in this study. See Section 2.4 of Chapter 2 for a detailed description of the datasets.

respect to the size of the graph, as it relies on the computation of the k-core decomposition (see Section 3.3).

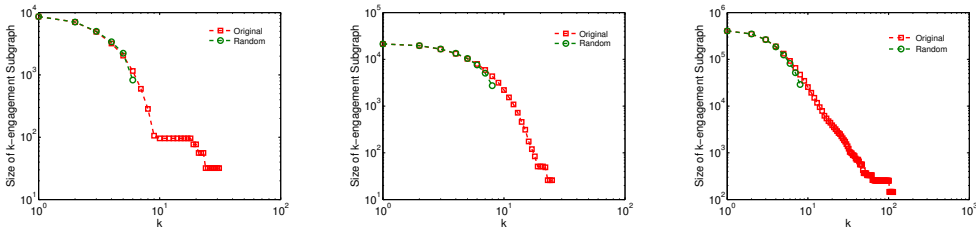
Table 3.1 presents the datasets used in our study. All of them are publicly available and correspond to well-known social and collaboration networks (except from the last datasets used to support our modeling approach). See Section 2.4 of Chapter 2 for more details about the datasets.

3.5.1 *High Level Properties of k-Engagement Subgraphs*

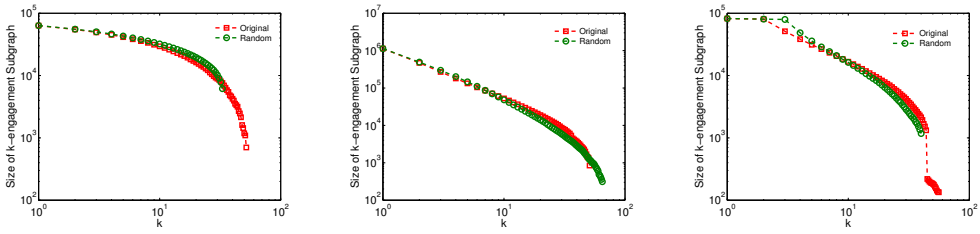
As we described in Section 3.4, a reasonable estimator for the engagement properties of a node is its core number, i.e., $e_i = c_i, \forall i \in V$. One important aspect here is to examine the size distribution of the k-engagement subgraphs, i.e., the size of the subgraphs that contain nodes with engagement e at least k. That is, for the various possible values of parameter k (that depend on the graph), we study the properties of the k-engagement subgraphs. These characteristics can help us to further understand the engagement dynamics of real graphs, both at node and at graph level. Figure 3.3 (red curve) depicts the results for the real graphs presented in Table 3.1.



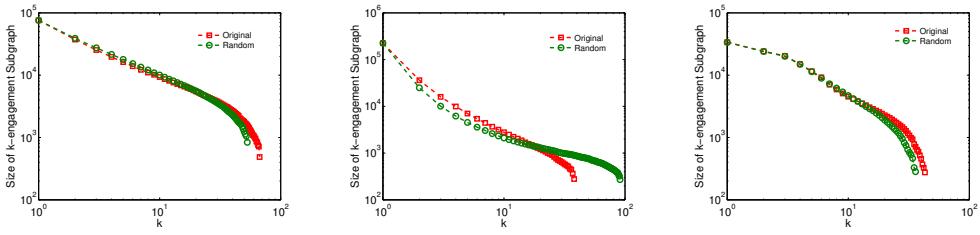
(a) CA-GR-QC (b) CA-ASTRO-PH (c) CA-HEP-PH



(d) CA-HEP-TH (e) CA-COND-MAT (f) DBLP



(g) FACEBOOK (h) YOUTUBE (i) SLASHDOT



(j) EPINIONS (k) EMAIL-EUALL (l) EMAIL-ENRON

Figure 3.3: Size distribution of the k-engagement subgraphs. Each plot depicts the size distribution of the k-engagement subgraphs vs. k, where $k = 1, \ldots, e_{\max}$. The red line corresponds to the distribution of the original graph, while the green one to the random graph with the same degree sequence as the original one.

As we can observe, for most of the datasets, the distribution of the sizes of the k-engagement subgraphs is almost skewed, meaning that the highly engagement subgraphs (for larger values of k) are relatively small in size. In other words, most of the nodes in the graph have small engagement index e, while a few nodes are highly engaged. Of course, we should note that the size distributions are not identical for all the graphs we have examined. Furthermore, the maximum engagement level $e_{\max}$ as well as the size of the Max-Engagement subgraph $\mathcal{G}_{\max}$ – that corresponds to the tail of the distribution – present different behavior for some of the examined datasets. We will discuss these points next in this Chapter.

One important question here is if these observations regarding the engagement properties of graphs, capture the behavior of a real system – and thus can be characterized as patterns of real graphs. In other words, is there any difference between the $\mathcal{G}_k$'s size distribution of real graphs and random ones? To answer this question, we have examined the engagement properties of a *configuration model*, i.e., a random graph model with the same degree distribution as the original one. As we can observe from Fig. 3.3 (green curve), a random rewiring of the original graph causes a different size distribution for the k-engagement subgraphs. More precisely, for most of the examined datasets, the random equivalent graph shows a much lower number of engagement levels, but the size of the Max-Engagement subgraph is much larger compared to the original one – indicating different behavior in terms of engagement. This observation is somewhat expected; random graphs are known to have a large core and thus Max-Engagement subgraphs of relatively large size [PSW96].

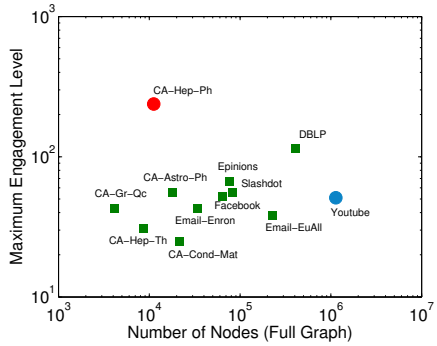
However, for a few datasets we have observed an unexpected but rather interesting behavior. For example, YOUTUBE and EMAIL-EUALL social graphs show an almost similar size distribution between the original graph and the random equivalent one. Additionally, EMAIL-EUALL has a much smaller maximum engagement index $e_{\max}$ compared to the random rewired graph. To better examine this deviation as well as for having a more refined explanation of the observed engagement properties, we have computed a set of high level characteristics of the k-engagement graphs. More specifically, we focus on the properties of the Max-Engagement subgraph $\mathcal{G}_{e_{\max}}$, trying

to understand and capture which factors potentially affect the engagement properties of the graph.

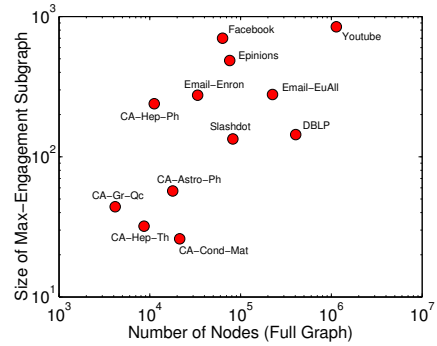
Figure 3.4 depicts the relationship between some high level characteristics of the Max-Engagement subgraphs $\mathcal{G}_{e_{\max}}$ with important global features of the graph (for the datasets of Table 3.1). We argue that examining interesting correlations between graph features and the observed engagement characteristics, we can draw meaningful conclusions about the engagement dynamics of real graphs. Initially, we consider the relationship between the size of the full graph, i.e., $|V|$ and the characteristics of the Max-Engagement subgraphs, namely the maximum engagement level $e_{\max}$ and the number of nodes in $\mathcal{G}_{e_{\max}}$. As we can see from Fig. 3.4 (a), for the majority of the examined datasets (green colored squares), the size $|V|$ of the graph shows an almost linear correlation (in log-log axis) with the maximum engagement level $e_{\max}$. However, YOUTUBE (blue colored circle) and CA-HEP-PH (red colored circle) clearly deviate from this relationship (if we ignore these two graphs, Pearson's correlation coefficient is $\rho = 0.75$). While YOUTUBE corresponds to the largest graph of our collection, its $e_{\max}$ value is relatively small. On the other hand, CA-HEP-PH has a relatively small size, while its maximum engagement level is extremely high. Thus, it seems that to achieve a higher $e_{\max}$ value, the size of the graph is not the only responsible factor. That is, as we have already mentioned, the existence of clustering structures in the graph plays a crucial role for the engagement properties.

Figure 3.4 (d) depicts the relationship between the fraction of closed triplets in the graph (triplets of nodes that form triangles) with the maximum engagement level. As we can observe, CA-HEP-PH has the largest fraction of closed triplets in our collection as well as the highest $e_{\max}$ value, although its size is relatively small. On the other hand, YOUTUBE shows an almost opposite behavior. Despite its large size, the fraction of closed triplets and the maximum engagement level are relatively small (in Section 3.5.4, we present a more detailed examination about the relationship of the engagement and the existence of clustering structures in the graph).

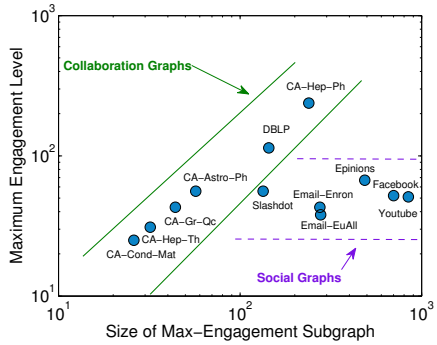
Figure 3.4 (b) depicts the number of nodes in the graph vs. the number of nodes in the Max-Engagement subgraph $\mathcal{G}_{e_{\max}}$. Here, we can observe a more



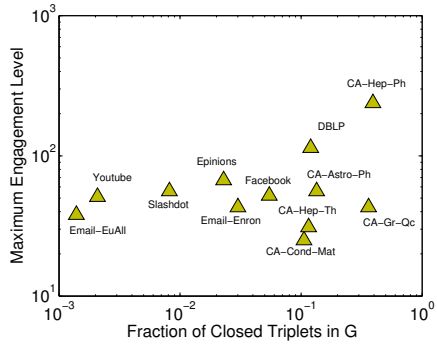
(a) $|V|$ vs. $e_{\max}$



(b) $|V|$ vs. number of nodes in $\mathcal{G}_{e_{\max}}$



(c) Number of nodes in $\mathcal{G}_{e_{\max}}$ vs. $e_{\max}$



(d) Closed triplets in G vs. $e_{\max}$

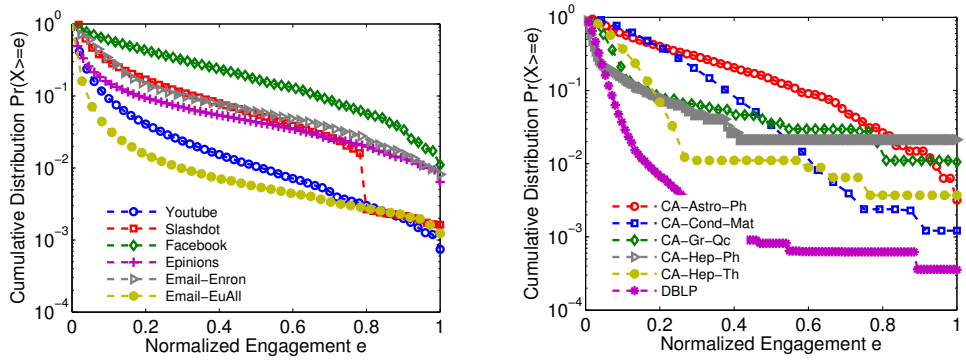
Figure 3.4: Characteristics of the $\mathcal{G}_{e_{\max}}$ subgraphs of the studied graphs (Table 3.1). (a) Maximum engagement level $e_{\max}$ vs. the size of the whole graph. (b) Number of nodes in the Max-Engagement subgraph vs. the size of the whole graph (Pearson correlation coefficient $\rho = 0.6394$). (c) Number of nodes in the Max-Engagement subgraph vs. maximum engagement level $e_{\max}$. Observe the different behavior between the collaboration (co-authorship) graphs and the social networks from social media applications. (d) Maximum engagement level $e_{\max}$ vs. fraction of closed triplets in the whole graph G.

clear correlation between $|V|$ and the size of $\mathcal{G}_{e_{\max}}$ (Pearson's correlation coefficient $\rho = 0.6394$).

Lastly, in Fig. 3.4 (c), we study the size of $\mathcal{G}_{e_{\max}}$ vs. the $e_{\max}$ values of the graphs. We can observe two different behaviors in the studied datasets. On the one hand, we have the collaboration graphs – formed by co-authorship relationships. Although they capture different scientific disciplines, we can observe an almost linear correlation in log-log scale. Furthermore, in many cases, the size of $\mathcal{G}_{e_{\max}}$ is almost equal to the maximum engagement level $e_{\max}$, indicating tightly knit communities at this portion of the graph. For example, in the case of the DBLP co-authorship graph, the Max-Engagement subgraph corresponds to a set of around 115 author that have co-authored the same paper. On the other hand, the graphs from online social networking and social media applications, follow a different behavior: the maximum engagement level is kept below a threshold of about 100 and the values are close to each other – almost constant – although the datasets are of different size, while the number of nodes in $\mathcal{G}_{e_{\max}}$ increases. This can be possibly explained by the nature of interactions in online social networking applications; although an individual can achieve a high number of friendship connections, the degree of collaboration – and similarly of engagement – among them is constrained to the threshold of around 100 nodes. We also note that, the value of $e_{\max}$ almost matches the size of the best communities (around 100 nodes) observed by Leskovec et al. [Les+09].

3.5.2 *Graph's Engagement Properties*

Having examined the properties of the k-engagement subgraphs, we proceed to the computation of the total graph engagement index $\mathcal{E}_G$. As we described in Section 3.4, the global engagement properties of the graph can be captured by the area under the curve of the normalized complementary cumulative distribution function of the sizes of the k-engagement subgraphs. That is, for each graph, we normalize the engagement level e in the interval $[0, 1]$ and we plot the cumulative distribution $\Pr(X \geq e)$, i.e., the fraction of nodes with normalized engagement at least e. Since we are not only interested in the maximum engagement level of each graph but on how individual nodes



(a) Social graphs

(b) Collaboration graphs

Figure 3.5: Normalized cumulative distribution function of the size of k-engagement subgraphs. Each curve corresponds to the probability $\Pr(X \geq e)$, i.e., the fraction of nodes with normalized engagement at least e. The area under curve captures the engagement index $\mathcal{E}_G$ for the whole graph.

are distributed within the different levels (as expressed by the k-engagement subgraphs), we are able to compare the engagement properties of different graphs. Figure 3.5 depicts the results for the collection of graphs of Table 3.1. Furthermore, Table 3.2 shows the $\mathcal{E}_G$ values that correspond to the area under curve.

In the case of social graphs (Fig. 3.5 (a)), we can observe that the graph with the maximum engagement index $\mathcal{E}_G$ is FACEBOOK. Although FACEBOOK does not have a large maximum engagement level $e_{\max}$, nodes are well distributed within levels, with a "good" fraction of nodes having high (normalized) engagement e. Looking now at the collaboration graphs (Fig. 3.5 (b)), a first observation is that the DBLP graph shows the lower engagement index $\mathcal{E}_G$, compared to the rest co-authorship graphs. One possible explanation is that DBLP covers several areas of computer science, with a significant number of relatively "new" authors. These authors, typically belong to lower cores of the graph, and thus their engagement is relatively low. On the other hand, the rest of the co-authorship networks correspond to more robust communities, where a larger fraction of authors (nodes) has higher engagement level.

Social Graph	$\mathcal{E}_G$
FACEBOOK	0.2514
YOUTUBE	0.0441
SLASHDOT	0.1221
EPINIONS	0.0755
EMAIL-EUALL	0.0277
EMAIL-ENRON	0.1245

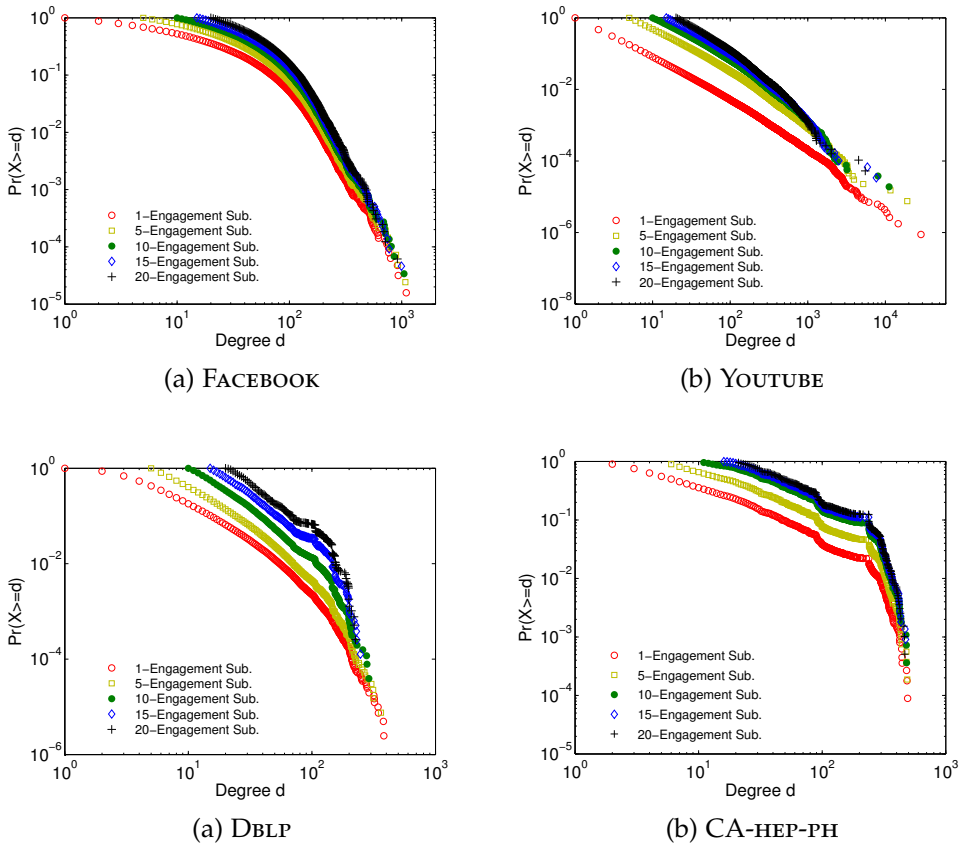
Collaboration Graph	$\mathcal{E}_G$
CA-GR-QC	0.0971
CA-ASTRO-PH	0.2293
CA-HEP-PH	0.0651
CA-HEP-TH	0.0969
CA-COND-MAT	0.1924
DBLP	0.0338

Table 3.2: Graph engagement values $\mathcal{E}_G$ for social (left table) and collaboration (right table) graphs.

3.5.3 *Near Self Similar k-Engagement Subgraphs*

In this section, we are interested to study the properties of the k-engagement subgraphs, under a simple scenario where nodes with low engagement e decide to depart. More specifically, we study the existence of self-similar properties in the k-engagement subgraphs and we focus on the simplest such property which is the existence of a skewed degree distribution. This property is crucial for the k-engagement subgraphs from several viewpoints. First of all, the degree of each node corresponds to an important structural characteristic and therefore it is interesting to examine to what extend it is preserved by the cascade of node departures. Furthermore, the existence of hubs in the k-engagement subgraphs is another crucial point, since – among other things – it is related to how fast information is disseminated in the graph and to the well known type of robustness under targeted/random node attacks [AB02]. Therefore, we are still interested to examine the major characteristics and functionalities of the graph after a cascade of dropouts.

Figure 3.6 depicts the complementary cumulative degree distribution of the k-engagement subgraphs, under different values of k (note that, $k = 1$ corresponds to the whole graph). A first observation is that the shape of the distribution is retained for the examined values of k. In other words, for the very first levels of engagement, an almost scale invariance is presented, with respect to the scenario of node departures. However, we do not argue that



(a) FACEBOOK (b) YOUTUBE

(a) DBLP (b) CA-HEP-PH

Figure 3.6: Complementary cumulative degree distribution of k-engagement subgraphs $\mathcal{G}_k$, for various values of k. Note that, the tail of the distribution also changes for different values of k.

this property is retained for all the engagement levels, i.e., $e = 1, \ldots, e_{\max}$[3]. Similar properties hold for the rest datasets.

An interesting point here is to examine the diversity of nodes – in terms of degree – that finally depart. In our scenario, nodes with low engagement decide to drop out. How is this mapped to the degree distribution? Typically, we expect that the nodes which depart, correspond to low degree ones. The crucial point here is that the produced cascades can cause the departure of high degree nodes as well, since their engagement level may be reduced. This is actually the major difference between the notion of node departures –

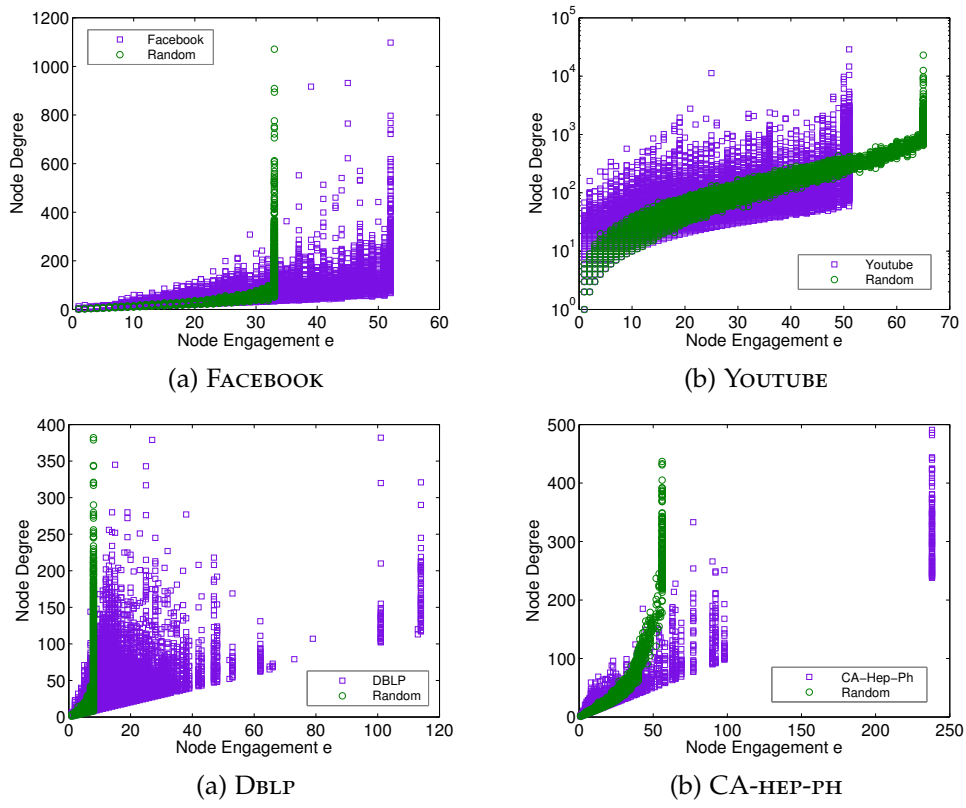
3 A similar behavior has also been noted for the Internet graph [AH+08].

based on the engagement level – studied by our work, compared to removals of nodes based on possible failures [AB02].

As we can observe from Fig. 3.6, for different values of k, the tail of the distributions – that captures high degree nodes – is also affected by the cascade of low-engaged node departures. To make this observation more precise, we compute the correlation between nodes' degree and their engagement index e. Clearly, high degree is required to achieve high engagement; however, the degree alone is not an indicator of high engagement. Figure 3.7 depicts the node engagement index e vs. the degree of each node (we focus only on four datasets of our collection). Clearly, a large fraction of high degree nodes show low engagement level, and thus it is more probable to depart. This is also an indication that the importance of some hub nodes in the graph diminishes, in terms of engagement dynamics. Lastly, we have examined this aspect in the case of random graphs with the same degree distribution as the original ones. As it can be shown from Fig. 3.7, these graphs show more smooth behavior compared to real ones; this is one more evidence about the differences in terms of engagement between real and random rewired graphs.

3.5.4 *Engagement and Clustering Structures*

As we have already discussed, it is expected that the engagement level of a node should be closely related to local clustering structures of the graph, indicating increased level of collaboration among nodes of the same neighborhood. Actually, the authors of [Wu+13] report that the probability of departure is related to the overall neighborhood activity of a node. In the more general problem of influence and product/behavior/idea adoption in a social system, the probability that a user will finally proceed with the adoption, is proportional to the size, as well as to the connectivity of the neighborhood [RMK11; Bac+06]. Furthermore, the high level characteristics presented in Fig. 3.4 (d), indicate a relationship between the fraction of closed triplets in the graph and the maximum engagement level. This is an interesting evidence, in the sense that in higher order k-engagement subgraphs (i.e., higher values of k), a higher degree of cohesiveness exists.



(a) FACEBOOK (b) YOUTUBE

(a) DBLP (b) CA-HEP-PH

Figure 3.7: Node engagement vs. node degree. Correlation between engagement and degree for each node in the graph. Purple squares correspond to real graphs and green circles to the random ones. For the real graphs, observe that high degree nodes can also have relatively small engagement.

In fact, the number of triangles that each node participates to, seems to be vital for its core number and therefore for its engagement index. Additionally, the fraction of closed triplets in a graph is closely related to the *clustering coefficient* – a measure that inform us about the tendency of nodes to cluster together, forming tightly knit groups[4]. Recently, Gleich and Seshadhri [GS12] showed that the number of cores in real-world graphs depends on the global clustering coefficient, where graphs with higher global clustering coefficient tend to have larger number of cores.

4 http://en.wikipedia.org/wiki/Clustering_coefficient.

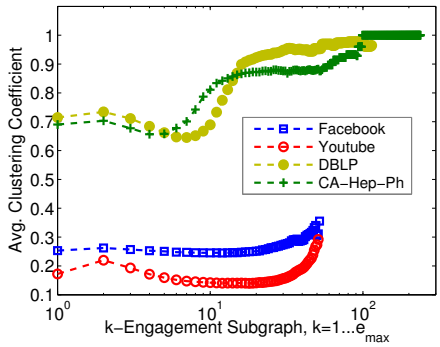
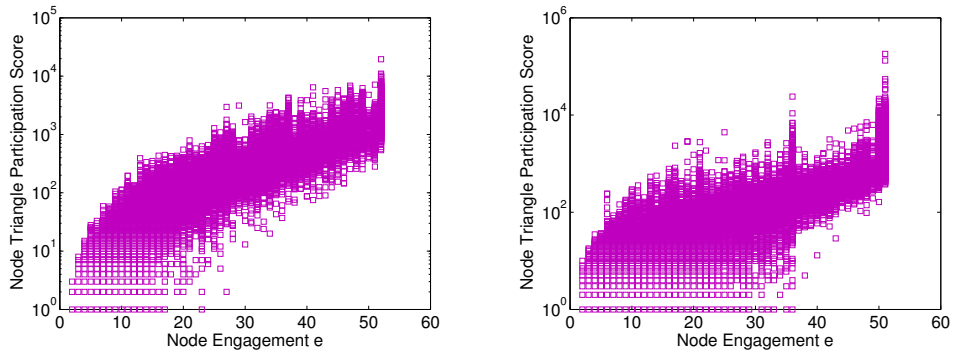


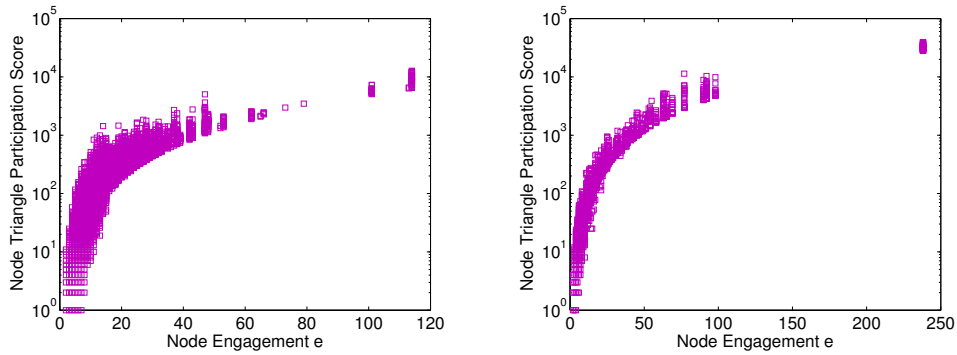
Figure 3.8: Average clustering coefficient per k-engagement subgraphs $\mathcal{G}_k$. Observe that the clustering coefficient is gradually increasing for larger values of $k = 1, \ldots, e_{\max}$.

Focusing now on the clustering properties of the k-engagement subgraphs, for different values of k ranging from $k = 1, \ldots, e_{\max}$, we examine how the clustering structure (as captured by the clustering coefficient) is affected by the departure of nodes. Figure 3.8 depicts the average clustering coefficient (CC) of each possible k-engagement subgraph, for four datasets of our collection. As we can observe, the CC increases gradually as we are moving to $\mathcal{G}_k$'s of higher engagement, indicating more cohesive subgraphs with higher degree of interactions among nodes. Actually, this captures the expected behavior of k-engagement subgraphs, in the sense that nodes which belong to higher order $\mathcal{G}_k$ (higher values of k), should demonstrate stronger degree of collaboration with their neighbors and thus higher engagement.

Lastly, we consider the clustering properties at node level and we examine how the triangle participation score of each node $i \in V$ (i.e., the number of triangles that each node participates to) is related to the engagement index e_i. Figure 3.9 presents the correlation of the engagement index e for each node vs. the triangle participation score. It seems that the triangle participation score approximates better the engagement level of a node (compared to the degree), supporting also our intuition that the existence of triangles is vital for the engagement properties.



(a) FACEBOOK (b) YOUTUBE



(a) DBLP (b) CA-HEP-PH

Figure 3.9: Node engagement vs. triangle participation score. Correlation of the node engagement index with the number of triangles that each node participates to.

3.6 DISENGAGEMENT SOCIAL CONTAGION

In this section, we briefly discuss some potential implications of our observations, regarding the property of engagement. As we presented in Section 3.4, the engagement of a node is proportional to its core number, and captures its incentive to remain in the graph. As we observed from the experimental results in Section 3.5, the size distribution of the k-engagement subgraphs is skewed, indicating that a large fraction of nodes typically show a relatively low engagement. Then, based on the size distribution, we were able to characterize the engagement properties of the whole graph. In that case, graphs in which a large portion of their nodes has high engagement level,

correspond to more robust graphs in terms of departures. In other words, in such graphs, a relatively high portion of nodes (based on the size of the full graph) do not have incentive to depart.

However, an interesting behavior can possibly occur if we consider a scenario under which nodes can also depart *independently* of their engagement level. In other words, although there is no incentive to depart, nodes decide to drop out possibly due to some external factors. Building upon the fact that such departures may affect the overall structure of the network, in Chapter 4 we propose and study a novel problem of robustness assessment in social networks based on the engagement level of each node.

3.7 CONCLUSIONS AND FUTURE WORK

In this part of the dissertation, we studied the problem of engagement estimation in a social graph. Based on a game-theoretic model, we proposed several ways to examine the engagement dynamics, at both node level as well as at graph level. The main contributions of this work concern the following points:

- *Introduce and study the notion of engagement in social graphs*. We studied the property of engagement in social graphs and we examined how it can be used to model the departure dynamics of the nodes in the graph.
- *Measures of engagement*. We proposed interesting and easy to compute measures for characterizing the engagement dynamics at both node and at graph level, based on game-theoretic and graph-theoretic concepts.
- *Experiments and observation on real graphs*. We performed several experiments on real-world graphs, observing interesting properties about the engagement dynamics.

As future work, we plan extend our study on more complex types of graphs, such as directed and signed graphs, where the engagement characteristics may behave in a different way. Furthermore, as we will present in Chapter 4, building upon the notion of engagement, we can derive a novel

concept of robustness assessment in real graphs, based on departures of individuals due to their engagement level and not by external attacks or failures which is the focus of many studies [AJB00; AB02].

4

VULNERABILITY ASSESSMENT IN SOCIAL NETWORKS UNDER CASCADE-BASED NODE DEPARTURES

Social networks constitute highly dynamic and evolving structures. Typically, new users decide to become members of a network, but also current users depart from the network, or stop being active in the activities of their community. The departure of a user may affect the engagement of its neighbors in the graph, that successively may also decide to leave, leading to a *disengagement epidemic*. In this Chapter, we propose the CasD (Cascading Departure) model to capture this cascading effect, based on recent studies about the engagement dynamics of social networks. We introduce a novel concept of *vulnerability assessment* in social networks, under cascades triggered by the departure of nodes based on their engagement level. Our results indicate that social networks are robust under cascades triggered by randomly selected nodes but highly vulnerable in cascades caused by targeted departures of nodes with high engagement level.

4.1 INTRODUCTION

Understanding the properties and dynamics of social networks is an interesting task with plenty of applications in both the Web and social sciences. Typically, the structure of social graphs is not static but is governed by an increasing level of evolution. Users decide to join in online communities for various reasons (*e.g.*, create new friendship relationships), that are mostly motivated by and express means of interaction among individuals in the social web, and these issues have become the starting points for an intense research activity in the area [New03; BBV08; EK10; Uga+12; Bac+06].

It is also expected that some users may decide to leave the network, or in general to stop being active in the activities of their community. This phenomenon is also known as churn or attrition and has been an important

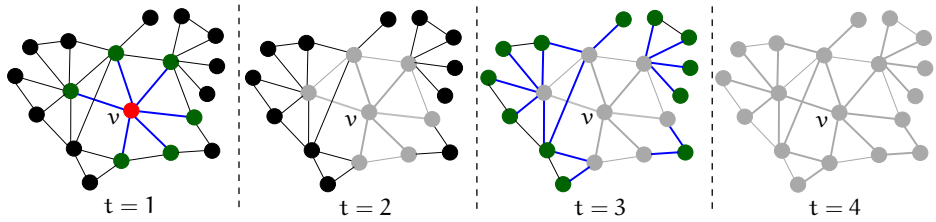


Figure 4.1: Example of cascading behavior triggered by the departure of a node. At time step $t = 1$ node v decides to depart. This will affect the decision of its neighbors (green color) that may decide to depart as well at time step $t = 2$ (gray color), forming an epidemic of disengagement in the network.

topic in the business domain [Das+08]. The key point here is that this action can affect the decision of their neighbors that, in turn, can decide to depart as well. That way, as shown in Fig. 4.1, the departure of a single node can become an *epidemic* forming a *cascade* of individual departures that potentially can lead the graph to a collapsed state. Being able to model and analyze such phenomena in real social networks is an important task, since they are related to the *vulnerability* of those social interaction systems under *node departures*.

The problem of robustness (or vulnerability) assessment in real networks has been extensively studied by different research communities, including sociology, statistical physics and computer science [CH10]. The observation of the power-law degree distribution in real networks [AB02] was the basis for several studies (e.g., Refs. [AJB00; Cal+00; Coh+01]), which showed that real networks are robust against random failures but vulnerable under attacks to high degree nodes. Later, similar observations were made for a cascading version of this robustness assessment problem, where the removal of a node can cause the redistribution of loads within the network, leading to a cascade of overload failures [ML02].

However, in the case of social networks, instead of degree-based failures and attacks, users decide to depart from or stay in the network based on their own *engagement* level. As we described in Chapter 3, recent studies about the departure dynamics in social networks suggest that the engagement level of nodes is not accurately captured by the node degree [Bha+11; MV13b;

GMS13] and therefore, well-known degree-based types of robustness assessment – such as the ones described above – do not accurately capture this feature of social networks.

Based on these points, the goal of this Chapter is to introduce and study a novel problem of vulnerability assessment in social graphs, under cascades caused by node departures. The main contributions can be summarized as follows:

- *Cascading Departure (CasD) model:* We propose the *CasD* model, a k-core decomposition-based model to capture the cascading (epidemic) disengagement effect due to the departure of a node. The model is based on recent studies about the departure dynamics in social networks, which suggest that the engagement level of nodes is captured more accurately by the core number compared to node degree [Bha+11; MV13b; GMS13].

- *Vulnerability assessment under node departures:* Combining the property of heavy-tailed core number distribution observed in social networks with the proposed Cascading Departure model, we introduce and study a new concept of vulnerability assessment in social graphs based on cascades triggered by random and targeted node departures according to their core number.

- *Experiments and finding on real social graphs:* We have performed experiments on real graphs, and our key observation is that online social networks are extremely robust under cascades started by random departures of nodes; however, they are highly vulnerable under cascades caused by targeted departures of nodes with high engagement level.

The rest of the Chapter is organized as follows. Section 4.2 briefly reviews the related work. Section 4.3 and Section 4.4 describe the proposed vulnerability assessment in social networks, as well as experimental results on real-world networks. Finally, we discuss concluding remarks in Section 4.5.

4.2 RELATED WORK

In this section we review the related work regarding the properties and dynamics of social networks, as well as main studies about the robustness assessment problem. A large number of studies about the dynamics of real-world graphs have been presented in the related literature (e.g., [New03; CF12; BBV08]). Of particular interest here are the recent theoretical and empirical works concerning the engagement and departures dynamics in social graphs [MJ09; Bha+11; Har13; MV13b; GMS13; Wu+13]. These results constitute the basis of our approach and were described in detail in Chapter 3. Close to our work are also theories and models about cascading (or epidemic) behaviors in networked environments [EK10; BBV08] and social contagion processes [Uga+12]. Moreover, related to our work can also be considered studies about the formation dynamics [Bac+06], social influence studies [Tan+09; Liu+10] and stability of social groups (i.e., the ability of a group within a social interaction network to remain stable, instead of shrinking, over time) [Pat+12; PLG13].

As we briefly presented in the Section 4.1, the cascading behavior of node departures can cause a *damage* to the network, in the sense that its structure may be affected; being able to model this effect is crucial due to the implications regarding the *robustness* of the network. The problem of robustness (or vulnerability) assessment in real networks has been extensively studied by different research communities, including sociology, statistical physics and computer science (see Ref. [CH10] for an extensive review on the problem). The observation of the power-law degree distribution in real networks [FFF99; AB02] was the basis for several works [AJB00; Cal+00; Coh+01], which shown that real networks are robust against random failures but vulnerable under attacks to high degree nodes. Later, similar observations were made for a cascading version of this robustness assessment problem, where the removal of a node can cause the redistribution of loads within the network, leading to a cascade of overload failures [ML02]. While our work follows a similar approach as in these studies, the major difference is that the proposed vulnerability assessment problem is more close to the case of social networks, where instead of degree-based failures and attacks,

Algorithm 4.1 Cascading Departure (CasD) Model

Input: Undirected graph $G = (V, E)$ and node $v \in V$
Output: Set of removed nodes R

$\mathbf{c} = [c_1, c_2, \ldots, c_{|V|}] = \texttt{k-core\_decomposition}(G)$
$\tilde{V} = V \setminus \{v\}$ /* Remove node v and the incident edges */
repeat
  /* Recompute the core number $\forall i \in \tilde{V}$ */
  $\tilde{\mathbf{c}} = \texttt{k-core\_decomposition}(G)$
  /* Normalize the core numbers $\tilde{c}_i$ into the interval $[0, 1]$ */
  $\tilde{c}_i^{\text{norm}} = \frac{\tilde{c}_i - \min(\tilde{\mathbf{c}})}{\max(\tilde{\mathbf{c}}) - \min(\tilde{\mathbf{c}})}, \forall i \in \tilde{V}$
  for all $i \in \tilde{V}$ **do**
    if $\tilde{c}_i < c_i$ **then**
      Remove node i from G with probability:
      $\Pr\left(\tilde{V} = \tilde{V} \setminus \{i\}\right) = 1 - \tilde{c}_i^{\text{norm}}$
      $R = R \cup \{i\}$
    end if
  end for
until No more nodes are removed
return Set of affected (removed) nodes R

users decide to depart or stay in the network based on their own engagement level which is not fully captured by the node degree property.

4.3 DISENGAGEMENT EPIDEMIC MODEL

In this section, we introduce a model of cascading departures in social graphs. The main idea is that the departure of a node can cause direct effects in its neighborhood, in the sense that some of the neighbors may also decide to depart. This cascading behavior is of particular significance in social networks, since it can lead to an *epidemic of disengagement*; a departure can trigger a cascade of successive departures that potentially can affect the overall structure of the graph.

Algorithm 4.1 gives the pseudocode of the proposed *Cascading Departure (CasD)* model. Next, we elaborate on its main aspects. Suppose that a node $v \in V$ decides to depart (later on we will describe how this node can be selected, namely randomly or targeted) and let $\tilde{V} = V \setminus \{v\}$ be the remaining node set. At each time step of the model, two points need to be specified: (i)

how to determine if a departure affects a neighborhood node, and (ii) how the nodes affected by the cascade decide to depart.

To address these points, we capitalize on the relationship between the engagement property and the core number, as described earlier. Let c_i and $\tilde{c}_i$ be the core numbers of a node i before and after the departure respectively. Each node $i \in V$ has an engagement level – that can be captured by the core number c_i – which expresses the incentive of the node to remain in the graph (or inversely, to depart). Thus, we consider that the nodes that are affected by a departure are those whose their core number c_i has changed after the departure of node v. We know that after the deletion of a node, the core number of each node $\tilde{c}_i$ in the graph, can either be reduced by one or remain. Thus, if $\tilde{c}_i < c_i$, node i is characterized as affected by the cascade caused by a departure (Line 7 in the pseudocode of Algorithm 4.1). Furthermore, in order to specify if an affected node $i \in \tilde{V}$ will finally depart, we consider that the probability of departure should be inversely related to the core number $\tilde{c}_i$ (nodes with lower core number are more probable to leave). Let

$$\tilde{c}_i^{\text{norm}} = \frac{\tilde{c}_i - \min(\tilde{\mathbf{c}})}{\max(\tilde{\mathbf{c}}) - \min(\tilde{\mathbf{c}})}, \forall i \in \tilde{V} \tag{4.1}$$

be the normalized core number of node i in the range $[0, 1]$, where $\tilde{\mathbf{c}} = [\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_{|\tilde{V}|}]$ (Line 5 of Algorithm 4.1) is a vector that contains the core numbers of each node, and $\min(\cdot), \max(\cdot)$ are functions that return the minimum and maximum element of a vector respectively. Then, node i is removed from the graph with probability $\Pr\left(\tilde{V} = \tilde{V} \setminus \{i\}\right) = 1 - \tilde{c}_i^{\text{norm}}$ (Line 8 of Algorithm 4.1). This process is repeated as long as nodes continue to be affected by departures during the previous time step. Finally, in order to quantify the disengagement effect of a departure, we keep track of the set of removed nodes R. As we will present shortly, the size of this set depends heavily on how the initial node v of the cascade is selected.

Figure 4.2 shows an intuitive toy example of the basic idea behind the model. In this example graph (Fig. 4.2 (a)), all nodes have core number equal to three (i.e., a 3-core). In the first case shown in Fig. 4.2 (b), suppose that node v decides to leave. Although this node can potentially affect its neighbors (blue colored edges), none of them has incentive to leave since

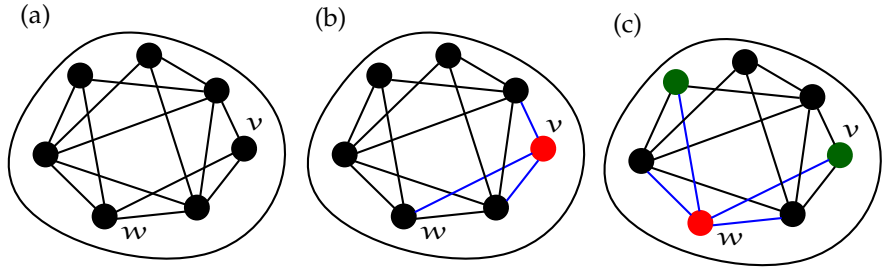


Figure 4.2: Example of two departures of nodes that can trigger (c) or not (b) a disengagement epidemic.

their core number is not affected. In the second case (Fig. 4.2 (c)), if node w leaves the network, the engagement level (as captured by the core number) for two of its neigbors will be affected (green colored nodes); that way, the core number of those nodes will be decreased to two and therefore they can decide to depart with probability inversely proportional to this value.

To the best of our knowledge, this is the first proposed approach to model the cascading behavior of node departures in social networks. Furthermore, in contrast to other degree-based models which mostly consider features of technological networks (like the Internet) that are responsible for *functional errors* (*e.g.*, Refs. [ML02; AB02]), the proposed model naturally captures the social component of an epidemic process in the social web, in the sense that the decision of individuals can potentially be affected by the decisions of other individuals within their social environment. As we will present in the next section, based on this model a new concept of vulnerability assessment in social networks can be derived.

4.4 VULNERABILITY ASSESSMENT UNDER NODE DEPARTURES

A seminal result in the area of robustness/vulnerability assessment in real graphs, was the finding that networks with a heterogeneous degree distribution (*e.g.*, power-law), tend to be highly robust against random failures but vulnerable under targeted attacks to high-degree nodes [AJB00]. In the case of social networks, instead of node failures, it is more meaningful to consider node departures; individuals decide to remain engaged or to depart based

			Cascading Departure Model	
Network Name	**Nodes**	**Edges**	*Random*	*Targeted*
EPINIONS	75,877	405,739	1.61×10^{-2}	6.31×10^{-1}
FACEBOOK	63,392	816,886	7.21×10^{-2}	7.58×10^{-1}
YOUTUBE	1,134,890	2,987,624	1.92×10^{-2}	5.74×10^{-1}
DBLP	404,892	1,422,263	2.70×10^{-3}	1.90×10^{-1}
CA-GR-QC	4,158	13,428	2.79×10^{-3}	1.12×10^{-2}
CA-COND-MAT	21,363	91,342	1.60×10^{-3}	1.84×10^{-1}

Table 4.1: Graph characteristics and fraction of removed nodes for random and targeted departures. Observe the significant difference in the fraction of affected nodes for the two strategies.

on their own engagement level – as captured by the core number. Here we will show that combining

(i) The CasD model that simulates how the disengagement epidemic is spreading, and

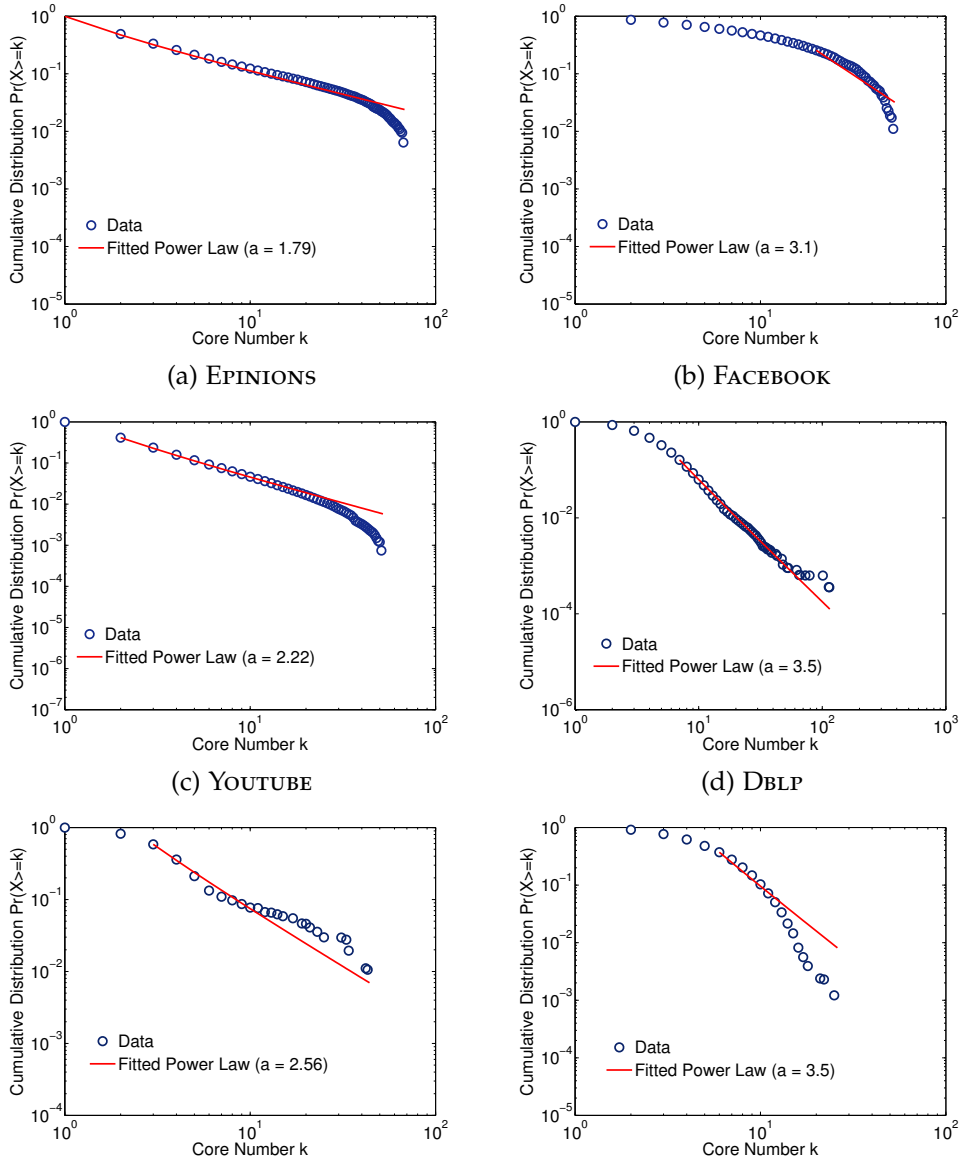
(ii) Our observations about the skewness of the core number distribution,

we can derive a novel vulnerability assessment problem under node departures that is suitable for the case of social networks.

4.4.1 *Observations on Real Graphs*

The CasD model is heavily based on the engagement level of each node, as captured by the core number. We have examined the core number distribution of a large collection of real-world social graphs presented in Table 4.1, and here we present our findings (in all cases, the examined networks are treated as undirected. In Section 2.4 of Chapter 2, we provide a detailed description of each network dataset.

Figure 4.3 depicts the complementary cumulative core number distribution function (CCDF) for each of the examined social networks. Each plot shows the cumulative fraction $\Pr(X \geq k)$ of nodes with core number $c_i = k$, in logarithmic scale on both axes. Note that, these plots depict similar



(a) EPINIONS (b) FACEBOOK

(c) YOUTUBE (d) DBLP

(e) CA-GR-QC (f) CA-COND-MAT

Figure 4.3: Complementary cumulative core number distribution function. Each plot depicts the distribution of the core numbers for the nodes of the graph on log-log scale. Observe that the distribution is heavy-tailed. The red line corresponds to the fitted power-law distribution.

information as the ones in Fig. 3.3 of Chapter 3. We can observe that the distribution is *heavy-tailed*, indicating that the vast majority of nodes have small core number, while only a few nodes in the graph have high core number (similar observations have been made for the Internet graph [AH+08; Car+07]). That is, the probability that a node has core number $c_i = k$ follows the form of $p(k) \propto \vartheta k^{-\alpha}$, for some constants $\vartheta > 0$ and $\alpha > 0$ (α is called the exponent or slope of the distribution). In terms of engagement dynamics, this observation states that a very small fraction of the nodes has high engagement (i.e., high core number), while most of the nodes show poor engagement characteristics. We have also fitted to the data a particular subclass of heavy-tailed distributions, called *power-law*. Figure 4.3 shows the fitted power-law distribution to the observed data (red colored line) as well as the estimated exponent α [CSN09]. Although the values of the exponent α vary, the distributions correspond to heavy-tailed ones.

Heavy-tailed distributions occur frequently in natural and man-made phenomena; prominent examples here are the degree distributions of the Internet topology and the Web graph [AB02]. Among other things, these seminal studies gave rise to the robustness assessment of real networks under *degree-based random and targeted* node removals (errors and attacks respectively) [AJB00; ML02; AB02]. In our case, the main message of our observations is that most of the nodes in social networks typically have low core number – and thus low engagement properties, while only a small portion of the nodes belongs to high k-cores of the graph. Therefore, if a randomly selected node decides to depart, this node is more probable to have low core number due to the skewed core number distribution (this is actually what is happening in social graphs; nodes decide to depart based on their engagement level, i.e., their core number c_i). As we will present shortly, such nodes typically cause a small scale cascade within the network, in the sense that their departure affects only a relatively small portion of the graph. This key observation constitutes the basis for a new problem of vulnerability assessment in social graphs under *random and targeted core-based* (i.e., engagement-based) node departures. Lastly, we should note that the existence of heavy-tailed core number distribution in real networks is not

necessarily correlated with the respective degree distribution; therefore, it can be considered as an inherent property of real-world networks.

4.4.2 *Social Vulnerability Assessment*

Based on above points, we define two different strategies of node departures, i.e., how to select a single node $v \in V$ that will depart first and trigger a cascade:

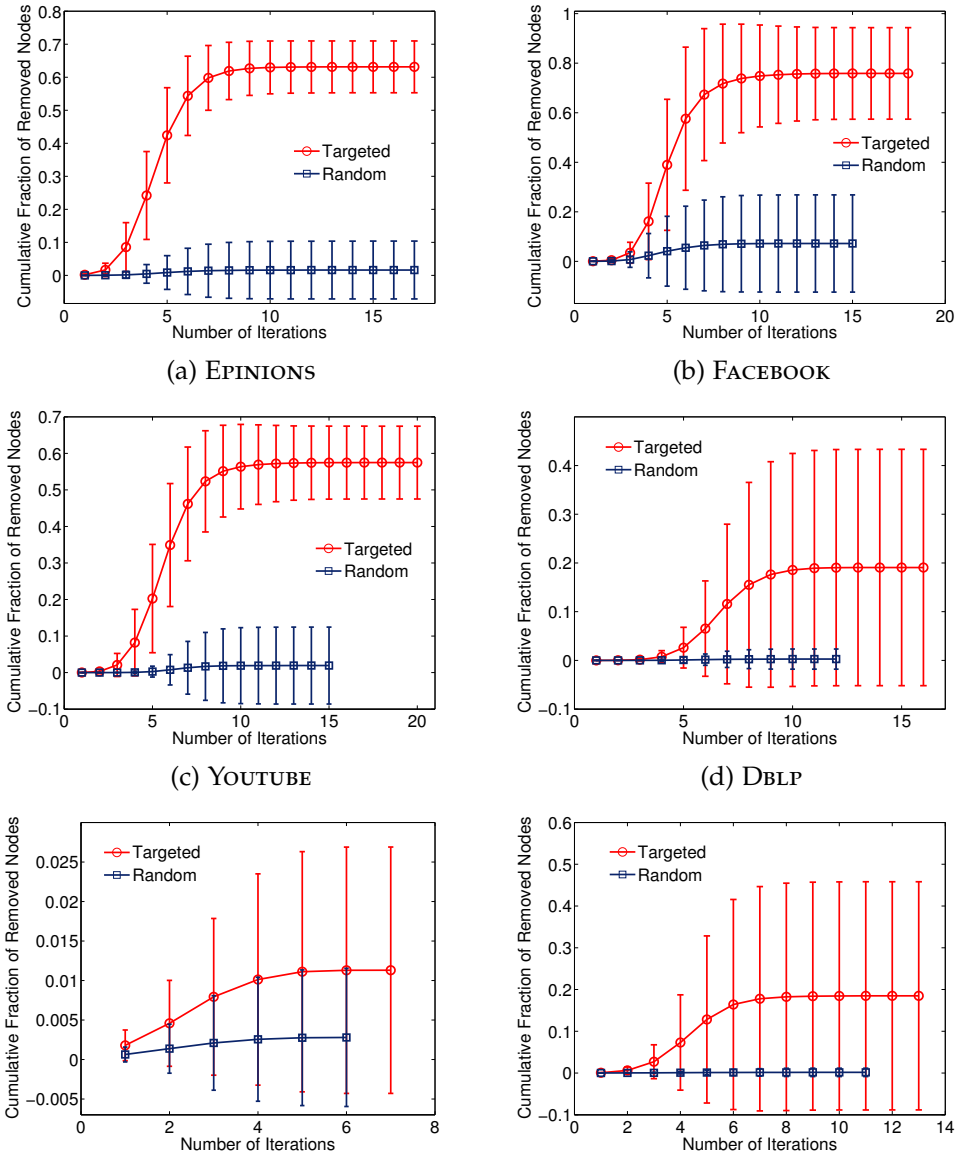
(i) *Random departure:* a randomly selected node leaves the graph.

(ii) *Targeted departure:* a node selected among the ones with the highest core number decides to depart.

The first strategy simulates what is more probable to occur in practice. The strategy of targeted departures captures the case in which a node, although it does not have incentive to leave (as expressed by a high core number), it finally departs due to external factors (such as an adversary that motivates a user to disengage from the activities of the network). To examine the dynamics of these two departure strategies, we apply the CasD model selecting accordingly the initial node v that will trigger the cascade. To assess the vulnerability of social networks under random and targeted node departures, we examine the total fraction of removed nodes during the execution of the model (i.e., for the time steps that the epidemic is spreading). What we argue here is that cascades triggered by the targeted departure of high core number nodes will affect a larger portion of the graph and will potentially cause a relatively large damage due to node disengagements. However, cascades triggered by random departures typically die out early causing negligible effects to the social structure.

4.4.3 *Experimental Results*

Figure 4.4 depicts the cumulative fraction of removed (or more generally affected) nodes per iteration of the CasD model under random and targeted



(a) EPINIONS (b) FACEBOOK

(c) YOUTUBE (d) DBLP

(e) CA-GR-QC (f) CA-COND-MAT

Figure 4.4: Cascading Departure Model triggered by random and targeted departures of nodes. Cumulative fraction of removed (affected) nodes per iteration of the model under random and targeted node departures based on their engagement level. Observe that the examined social graphs are robust in random node departures but vulnerable under targeted ones.

departures (iterations here mean the number of **repeat** calls (Line 3 of Algorithm 4.1) in the model). All the experiments were conducted 100 times (30 times for the YOUTUBE dataset due to its size) and here we report average values. As we can observe, in the case where the initial node that will depart first is selected randomly, the fraction of affected nodes is extremely small and the epidemic typically dies out early without affecting large portion of the graph. As shown in Table 4.1, for the EPINIONS, FACEBOOK and YOUTUBE graphs, the total fraction of removed nodes is in the order of 10^{-2}. The behavior of the collaboration networks DBLP, CA-GR-QC and CA-COND-MAT slightly deviates; the total number of affected nodes is in the order of 10^{-3} (i.e., roughly negligible), that is actually even less intense compared to the examined social networks. These empirical evidences suggest that real social graphs tend to be extremely *robust against cascades triggered by random departures of nodes*, which is actually the typical behavior in social graphs as we have already discussed.

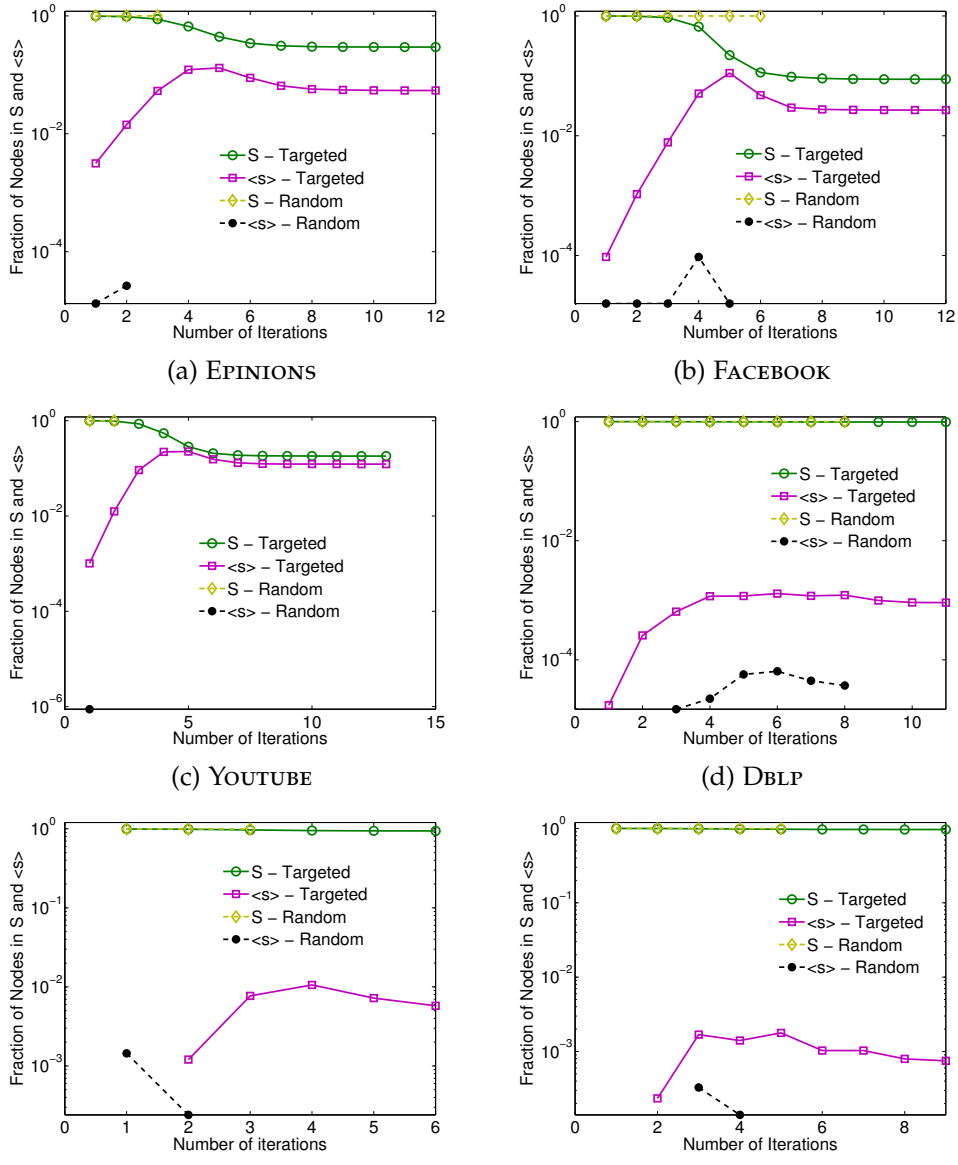
What is happening in the case of a targeted departure, where an adversary motivates a highly engaged user to depart? As we can observe from Fig. 4.4, when the epidemic starts by the departure of a node randomly selected among the ones with the highest core number (targeted selection), the behavior is completely different from the one in the random case. We have noticed that for all datasets, during the first iterations, the number of affected nodes increases exponentially. As we can see from Table 4.1, targeted departures have the potential to affect a large portion of the graph (in some cases more than 50% of the nodes), and this behavior is persistent for all datasets. Therefore, we argue that real social graphs are *vulnerable under cascades triggered by targeted departures of high core nodes*. The good spreading properties of high core number nodes have already been discussed by Kitsak et al. [Kit+10], under different settings compared to the ones described here.

Additionally, the vulnerability of social networks under node departures can be assessed by examining the fragmentation of the graph, as captured by the fraction of nodes belonging to the largest connected component (i.e., the giant connected component) S and the fractional size of the remaining isolated components $\langle s \rangle$. Figure 4.5 depicts the graph fragmentation for an individual execution of the CasD model. As we can observe, in the case

of a cascade produced by the departure of a random node, the fraction of nodes that belong to S and $\langle s \rangle$ is slightly affected during the iterations of the CasD model. Since the initial graphs into which the epidemic process takes place are connected, this property is also retained during the execution of the model and at the end, almost all nodes of the graph are still part of the largest connected component S. On the other hand, in the case of a cascade triggered by a targeted departure, the fragmentation of the network is more intense. As it is shown in Fig. 4.5, the fraction of nodes that belong to the largest component S decreases, while the size of the remaining smaller components increases exponentially fast. Moreover, in the case of social networks (EPINIONS, FACEBOOK and YOUTUBE), after the first few iteration steps of the model, the size of both S and $\langle s \rangle$ almost stabilizes until the last iteration (this property was observed on a single execution of the model). Also, notice that the fragmentation is not so intense for the collaboration networks DBLP, CA-GR-QC and CA-COND-MAT. Summarizing, all the above findings suggest an additional *robust-yet-fragile* property of networks (e.g., [Wato2]) with heterogeneous structural properties.

4.5 CONCLUSIONS AND DISCUSSION

In this Chapter, we studied a new problem of vulnerability assessment that seems to be of particular importance in the case of social networks. Unlike previous studies that mostly perform degree-based robustness assessment, we built upon recent studies that propose the core number of a node as a reasonable indicator about the engagement and departure dynamics in social networks, and we introduced the CasD (Cascading Departure) model to describe the epidemic effect triggered by the departure of a node. Then, combining the observation about the heavy-tailed core number distribution with the proposed CasD model, we introduced a new concept of vulnerability assessment in social networks based on random and targeted core number based node departures. We performed simulation experiments on real graphs and our empirical findings indicate that social networks are robust against cascades triggered by random departures of nodes, but highly vulnerable under cascades caused by targeted departures of high core nodes.



(a) EPINIONS (b) FACEBOOK

(c) YOUTUBE (d) DBLP

(e) CA-GR-QC (f) CA-COND-MAT]

Figure 4.5: Graph fragmentation under random and targeted departures for an individual run of the CasD model. Fraction of nodes (logarithmic scale) in the largest connected component S and in the rest isolated connected components $\langle s \rangle$ per iteration of the CasD model.

We consider that the proposed concept of vulnerability assessment is more close to what really occurs in social networks and the social web, and suggests several directions for future work. One possible direction could be to further validate the predictive cascade capabilities of the model by examining departure (or inactivity) traces of real networks. The CasD model can be also thought of as an epidemic process [BBV08] and therefore a more thorough theoretical analysis of its properties is also an interesting future direction. Lastly, it would be interesting to investigate how additional features that may contribute to the decision of a node to depart can be incorporated to the model or how to model more complex types of interactions that arise in social networks (e.g., positive/negative interactions among individuals modeled by signed networks).

5

LOCATING INFLUENTIAL SPREADERS IN SOCIAL NETWORKS

Understanding and controlling spreading processes in networks is an important topic with many diverse applications. In the core of this task lies the problem of identifying influential nodes in complex networks, which is the focus of this Chapter. Our goal is to locate individual influential nodes, which are able to perform fast and wide epidemic spreading. To that end, we capitalize on the properties of the K-truss decomposition, a triangle-based extension of the core decomposition of graphs. Our analysis on real networks indicates that the nodes belonging to the maximal K-truss subgraph show better spreading behavior compared to previously used importance criteria; more nodes get infected during the outbreak of the epidemic, and also the total number of infected nodes at the end of the process is higher.

5.1 INTRODUCTION

Spreading processes in complex networks have gained great attention from the research community due to the plethora of applications that they occur, ranging from the spread of news and ideas to the diffusion of influence and social movements and from the outbreak of a disease to the promotion of commercial products. Being able to understand the underlying mechanisms that govern such processes is a crucial task with direct applications in various fields, including epidemiology, viral marketing and collective dynamics.

Typically, the interactions among individuals are responsible for the formation of information pathways in the network and to this extend, their position and topological properties have direct effect to the spreading phenomena occur in the network. That way, a fundamental aspect on understanding and controlling the spreading dynamics is the identification of *influential spreaders*

that can diffuse information to a large portion of the network. For example, in the case of virus propagation, such as influenza, the transmission of the disease mainly depends on the extend of contacts of the infected person to the susceptible population; thus, being able to locate and vaccinate individuals with good spreading properties can prevent from a potential outbreak of the disease, leading to efficient strategies of epidemic control. In a similar way, suppose that our goal is to promote a product or an idea in order to be adopted by a large fraction of individuals in the network. A key idea behind viral marketing is the *word-of-mouth* effect [TBP09]; individuals that have already adopted the product, recommend it to their friends who in turn do the same to their own social circle, forming a cascade of recommendations [DR01]. The basic question here is how to target a few initial individuals (e.g., by giving them free samples of the product or explaining them the idea), that can maximize the spread of influence in the network, leading to a successful promotion campaign.

The problem of identifying nodes with good spreading properties in networks, can be further split in two subtopics:

(i) Identification of *individual influential nodes* with good spreading properties.

(ii) Identification of a *group of nodes* that, by acting all together, are able to *maximize* the total spread of influence.

For example, in disease spreading, the process is typically triggered by a single individual node in the network. On the other hand, in the case of viral marketing, the goal is to convince a small subset of individuals to adopt a new product, in such a way that, at the end of the process, a large number of individuals will be influenced [DR01; RD02]. The latter problem is known as *influence maximization* [KKT03; KKT05; KKT15] and is briefly discussed in Section 5.2.

In this work, we focus on the problem of identifying single influential spreaders in networks. The process that is typically followed to locate individual nodes with good spreading properties, consists of two steps: (i) initially, the spreading capability of nodes is quantified based on network

topological criteria; (ii) the nodes are ranked according to the chosen criterion and the top-ranked ones as selected as the most influential. These steps are accompanied by models that simulate how the process is spreading over the network (e.g., the SIR model discussed in Section 5.2.2).

A straightforward approach towards finding effective spreading predictors in networks, is to consider *node centrality criteria* and in particular the one of *degree centrality*. In fact, several studies have examined how the existence of heavy-tailed degree distribution in real-world networks [FFF99] is related to cascading effects concerning the robustness of such complex systems [AB02; AJB00; PSV01; Coh+01]. Nevertheless, there exist cases where a node can have arbitrarily high degree, while its neighbors are not well-connected, making degree a not very accurate predictor of the spreading properties. For example, this can occur when a high degree node is located to the periphery of the network. In fact, the spreading properties of a node are strongly related to the ones of its neighbors in the graph, and thus, global centrality criteria seem to be more appropriate for this task.

Of particular importance is the work by Kitsak et al. [Kit+10], which stressed out that highly connected nodes or those having high betweenness and closeness centralities, have little effect on the range of the spreading process. The main finding of their work was that, less connected but strategically placed nodes in the core of the network, are able to disseminate information to a larger part of the population. To quantify the core-periphery structure of networks, they applied the k-core decomposition algorithm [BZ03] – a pruning process that removes nodes which do not satisfy a particular degree-based threshold. Their results indicated that nodes belonging to the maximal k-core subgraph are able to infect a larger portion of the network, compared to node degree or betweenness centrality, making the k-core number of a node a more accurate spreading predictor. Furthermore, extracting the k-core subgraph is a more efficient task compared to the heavy computation required by some centrality criteria (e.g., betweenness). Nevertheless, the resolution of k-core decomposition is quite coarse; depending on the structure of the network, many nodes will be assigned the same k-core number at the end of the process, even if their spreading capability differs from each other.

Our proposed approach moves on a similar axis as the one by Kitsak et al. [Kit+10]; we argue that the topological properties of the nodes play a crucial role towards understanding their spreading capabilities. In particular, we consider that only a relatively small fraction of the nodes extracted by the k-core decomposition method corresponds to highly influential nodes. To that end, we study the spreading properties of the K-truss decomposition [Coho8; WC12; ZP12], a triangle-based extension of the k-core decomposition, showing that it can serve as an even better criterion to identify privileged spreaders. The algorithm is able to extract a more refined and even more dense subgraph of the initial graph – compared the k-core decomposition – as the K-truss is structurally more close to a clique. The main contributions of this work can be summarized as follows:

- *K-truss decomposition for locating influential nodes:* We propose the K-truss decomposition algorithm, as a graph-theoretic mechanism to identify nodes with good spreading properties in the network.

- *Experimental evaluation on real graphs:* We perform experiments on large scale real-world graphs, showing that the nodes belonging to the maximal K-truss subgraph of the network show better spreading behavior under the SIR epidemic model – compared to previously used importance criteria – leading to faster and wider epidemic spreading. Furthermore, the extracted nodes dominate the small set of nodes that achieve the optimal spreading in the network.

The rest of the Chapter is organized as follows. Section 5.2 presents the background concepts that are used throughout the Chapter and Section 5.3 reviews the related literature on the problem of identifying influential nodes in complex networks. Then, in Section 5.4 we present the proposed method for locating influential spreaders. Section 5.5 presents a detailed experimental evaluation of our method. Finally, in Section 5.6 we present concluding remarks.

Symbol	Definition
$G = (V, E)$	Undirected graph G
V, E	Node and edge set of graph G
$n = \|V\|, m = \|E\|$	Number of node and edges of G
$d(v)$	Degree of node $v \in V$
$Nb(v)$	Set of neighbors of node v
$\triangle_{uvw}$	Triangle subgraph defined by nodes u, v, w
T_K	K-truss subgraph
C_k	k-core subgraph
$t_{edge}(e)$	Truss number of edge $e \in E$
$t_{node}(v)$	Truss number of node $v \in V$
$\mathcal{C}$	Set of nodes with maximum core number value c
$\mathcal{T}$	Set of nodes with maximum t_{node} value
M_v	Average infection size caused by node v
τ	Epidemic threshold

Table 5.1: List of symbols and their definitions.

5.2 PRELIMINARIES AND BACKGROUND

In this Section, we discuss the preliminary concepts upon which our approach for finding influential nodes is built. Initially, we describe in detail the concept of K-truss decomposition in graphs. Then, we define epidemic propagation models that are used to simulate information diffusion processes on networks. Table 5.1 provides a list of symbols used in this Chapter, along with their definitions.

5.2.1 *K-truss Decomposition*

Before introducing the K-truss decomposition which constitutes the basis of our approach, we briefly recall to the notion of k-core decomposition in networks (a detailed description can be found in Chapter 2, Section 2.3). Let $G = (V, E)$ be an undirected graph. C_k is defined to be the k-core subgraph of G if it is a maximal connected subgraph in which all nodes have degree at least k. Then, each node $v \in V$ has a core number $c(v) = k$, if it belongs to a k-core but not to a $(k+1)$-core. In this Chapter, we denote as C the set of

nodes with the maximum core number $k_{\max}$, i.e., the nodes of the maximal k-core subgraph of G that corresponds to the maximum value of k.

The K-truss decomposition extends the notion of k-core using triangles [Tso08; Tso+09; Par+14; Bec+10], i.e., cycle subgraphs of length 3 [Coh08; WC12; ZP12].

Definition 5.1 (Triangle subgraph). *Let $G = (V, E)$ be an undirected graph. We define as a triangle $\triangle_{uvw}$ a cycle subgraph of nodes $u, v, w \in V$. Additionally, the set of triangles of G is denoted by $\triangle_G$.*

Definition 5.2 (Edge support). *The support of an edge $e = (u, v) \in E$ is defined as $sup(e, G) = |\{\triangle_{uvw} : \triangle_{uvw} \in \triangle_G\}|$ and expresses the number of triangles that contain edge e.*

Definition 5.3 (K-truss subgraph). *Given an undirected graph $G = (E, V)$, the K-truss, $K \geq 2$, denoted by $T_K = (V_{T_K}, E_{T_K})$, is defined as the largest subgraph of G, where every edge is contained in at least $K - 2$ triangles within the subgraph, i.e., $\forall e \in E_{T_K}, sup(e, T_K) \geq K - 2$.*

Definition 5.4 (Edge truss number $t_{edge}(e)$). *The truss number of an edge $e \in G$ is defined as $t_{edge}(e) = \max\{K : e \in E_{T_K}\}$. Thus, if $t_{edge}(e) = K$, then $e \in E_{T_K}$ but $e \notin E_{T_{K+1}}$. We use $K_{\max}$ to denote the maximum truss number of any edge $e \in E$.*

Definition 5.5 (K-class). *The K-class of a graph $G = (V, E)$ is defined as $\Phi_K = \{e : e \in E, t_{edge}(e) = K\}$.*

Based on the above definitions, we can now introduce the concept of K-truss decomposition.

Definition 5.6 (K-truss decomposition). *Given a graph $G = (V, E)$, the K-truss decomposition is defined as the task of finding the K-truss subgraphs of G, for all $2 \leq K \leq K_{\max}$. That is, the K-truss can be obtained by the union of all edges that have truss number at least K, i.e., $E_{T_K} = \bigcup_{j \geq K} \Phi_j$.*

The computation of the K-truss subgraph, for a specific value of $K \geq 2$, can be done based on the following procedure: remove all edges $e = (u, v) \in E$ if they do not participate to at least $K - 2$ triangles, i.e., $|Nb(u), Nb(v)| \leq K - 2$. Algorithm 5.1 describes the pseudocode for performing K-truss decomposition and computing the truss number $t_{edge}(e), \forall e \in E$ [WC12].

Algorithm 5.1 *K*-truss decomposition

Input: Undirected graph $G = (V, E)$
Output: K-truss subgraphs, for $3 \leq K \leq K_{\max}$
1: $K \leftarrow 2$
2: **for** each $e = (u, v) \in E$ **do**
3: $sup(e) = |Nb(u), Nb(v)|$
4: **end for**
5: **repeat**
6: **while** $(\exists e = (u, v) : sup(e) < K - 2)$ **do**
7: $W \leftarrow Nb(u) \cap Nb(v)$
8: **for** each $e' = (u, w)$ or $e' = (v, w)$, where $w \in W$ **do**
9: $sup(e') \leftarrow sup(e') - 1$
10: **end for**
11: $t_{edge}(e) = K$ /* Truss number of edge e */
12: $E = E \setminus \{e\}$ /* Remove e from G */
13: **end while**
14: $T_K \leftarrow G$ /* Output G as the K-truss subgraph */
15: $K \leftarrow K + 1$
16: **until** $|E| = \emptyset$ /* All edges in G have been removed */
17: **return** T_K for $3 \leq K \leq K_{\max}$

ANALYSIS OF THE COMPLEXITY. The complexity of Algorithm 5.1 is polynomial. In general, the computation of the support values in Step 3, i.e., the number of triangles that each edge participates to, can be done in $\mathcal{O}(d_{\max}^2)$ time, where $d_{\max}$ is the maximum degree in G. Step 7 of the algorithm requires time $\mathcal{O}(d(u) + d(v))$ for each edge $e = (u, v) \in E$, giving total time complexity proportional to $\mathcal{O}\Big(\sum_{e=(u,v)\in E} \; d(u) + d(v)\Big) = \mathcal{O}\Big(\sum_{v\in V} \; d^2(v)\Big)$. Also, the space complexity of the algorithm is $\mathcal{O}(m + n)$ [Coho8; WC12]. Wang and Cheng [WC12] proposed an improved algorithm for K-truss decomposition, based on fast triangle counting algorithms [Lato8; Scho7]. The time complexity of the improved algorithm is $\mathcal{O}(m^{1.5})$ and the space complexity $\mathcal{O}(m + n)$.

Next, we provide interesting properties of the K-truss subgraphs that will be later used in our analysis.

Proposition 5.7 ([Coho8]). *Every node v in a K-truss subgraph T_K has degree $d(V) \geq K - 1$.*

Proof. Let v be a node of T_K. Since a K-truss subgraph do not contain isolated nodes, v should be incident to an edge $e = (v, w) \in E_{T_K}$. By the definition

of T_K, nodes v and w must share at least $K-2$ additional neighbors (except from w and v respectively). Then, there should be at least $K-1$ nodes adjacent to v, i.e., $d(v) \geq K-1$. □

Proposition 5.8. *The K-truss subgraph T_K is contained within the $(K-1)$-core subgraph.*

Proof. According to Proposition 5.7, each node of T_K has degree at least $K-1$ within the K-truss subgraph. Thus, T_K is part of a $(K-1)$-core subgraph. □

Although relatively new, the concept of K-truss decomposition has been applied in many application domains, including network vizualization [ZP12], online community search for a given query node [Hua+14], coloring of networks [RA14] and anomaly detection in financial transaction data [RHC12]. Furthermore, there is an intense recent effort towards scalable algorithms for the K-truss decomposition method capable to deal with large scale networks, including MapReduce [Coh09; CCC14], Pregel [QWH12] and parallel implementations [Ros14].

5.2.2 *Epidemic Models*

Modeling the epidemic and contagious processes in networks, is an active research topic with plenty of applications in several disciplines. Researchers from various fields, including epidemiology, computer science and social science, share similar models to study spreading phenomena; typically, they rely on similar methodological concepts to describe and analyze how viruses, influence, ideas and innovation spread over a network [BBV08]. As we have already mentioned, our approach relies on epidemic models for determining the spreading effect of specific nodes in the network; in this section, we describe the basic concepts behind the model used by our study.

One of the mostly studied epidemic models is the *Susceptible-Infected-Revovered* (SIR) [KM27; BBV08; Hetoo]. The model assumes a population of N individuals, divided on three states:

- *Susceptible (S):* The individual is not yet infected, thus being susceptible to the epidemic.

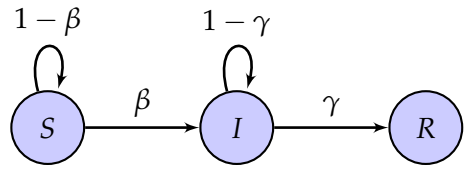


Figure 5.1: State diagram of the SIR model.

- *Infected (I):* The individual has been infected with the disease and it is capable of spreading the disease to the susceptible population.

- *Recovered (R):* After an individual has experienced the infectious period, it is considered as removed from the disease and it is not able to be infected again or to transmit the disease to others (immune to further infection or death).

The infected individuals are able to contact with randomly chosen individuals at average rate β per unit time, called *infection rate*. Furthermore, an infected individual can recover at an average rate γ per unit time, called *recovery rate*. Figure 5.1 depicts the state diagram of each individual at the SIR model.

Let $S(t)$, $I(t)$ and $R(t)$ be the number of susceptible, infected and recovered individuals at time t. Then, the model can be described by the following differential equations:

$$\begin{aligned} \frac{dS}{dt} &= -\frac{\beta SI}{N} \\ \frac{dI}{dt} &= \frac{\beta SI}{N} - \gamma I \\ \frac{dR}{dt} &= \gamma I. \end{aligned} \tag{5.1}$$

The last equation can be considered as redundant, since $S(t) + I(t) + R(t) = N$. In the limit of large population size N, from these equations analytical solutions for several quantities can be derived, such as the size of the outbreak.

The model described above and the derived equations, assumes that the population is *fully mixed*, meaning that an individual that is infected can equally infect any other member of the population or subpopulation to which it belongs to. To overcome this assumption and come up with a more realistic modeling approach, we can consider the underlying network of connections among individuals. That way, any susceptible node can only be infected by an infected neighbor on the graph and several studies in network science have focused on understanding the spreading dynamics of the SIR model (and other variants) in networks with special properties [PSV01; MPSV02; New02; BBV08].

As we will present in Section 5.4, in our implementation of the model (Algorithm 5.2), initially all the nodes of the network are set at the susceptible state S, except from the one that we are interested to examine its performance which is set at the infected state I. Then, at each time step t of the process, every node that is on the I state can infect its susceptible neighbors with probability β and afterwards it can recover with probability γ. Note that, a node cannot directly pass from state I to state R during the same time step t.

5.3 RELATED WORK

In this section, we review the related work for the problem of identification of influential spreaders in complex networks. Initially we present methods for the task of identifying single spreaders in networks – being also the goal of the proposed work. Then, we briefly refer to the orthogonal problem of selecting a set of nodes for influence maximization.

5.3.1 *Identifying Individual Spreaders in Networks*

As we discussed in the Introduction, in order to find effective predictors for the spreading capabilities of nodes, several centrality criteria can ne applied. Lu et al. [Lü+11] proposed LeaderRank, a random walk-based algorithm that is able to outperform PageRank [BP98] on identifying influential users in social networks. Later, Li et al. [Li+14] extended LeaderRank to properly

detect influential nodes in weighted networks. Chen et al. [Che+12] proposed a semi-local centrality measure which serves as a tradeoff between degree and other computationally complex measures (betweeness and closeness centrality). The authors of [Che+13a] proposed ClusterRank, a local ranking method that takes into account the clustering coefficient of a node. In Ref. [Che+13b], the diversity of the paths that emanate from a node was considered. The main idea was that the spreading ability of a node may be reduced if its propagation depends only on a few paths, while the rest ones lead to dead ends.

Building upon the fact that the k-core decomposition is an effective (and efficient) measure to capture the spreading properties of nodes, as introduced by Kitsak et al. [Kit+10] (also presented in the Introduction), several extensions have been proposed. The authors of Ref. [ZZ13] introduced a modified version of the k-core decomposition in which the nodes are ranked taking into account their connections to the remaining nodes of the graph as well as to the removed nodes at previous steps of the process. They showed that the proposed node ranking method is able to identify nodes with better spreading properties compared to the traditional k-core decomposition. Bae et al. [BK14] extended the metric of k-core number of each node by considering the core number of its neighbors. That way, the ranking produced by the method is more fine-grained in the sense that the effect of assigning the same score (i.e., k-core number) to many nodes is eliminated. Basaras et al. [BKT13] proposed to rank the nodes according to a criterion that combines the degree and the k-core number of a node within an μ-hop neighborhood. In Ref. [HYL12] the authors introduced a criterion that combine three previously examined measures, namely degree, betweenness centrality and core number. The intuition was that, most of the widely used centrality criteria produce highly correlated rankings of nodes; combining them in a proper way, we are able to achieve a more accurate indicator of influential nodes. Zhang et al. [Zha+13] proposed a method to locate influential nodes taking into account the existence of community structure in networks. In Ref. [BHRM12], the authors considered real social media data, in order to examine to what extend the structural position of a user in the network allows us to characterize the ability of an individual to spread rumors effectively. Their results indicate

that although the most appropriate feature is the degree of a node, only a few such highly-connected individuals exist; however, by considering the k-core number metric, we are able to locate a larger set of individuals that are likely to trigger large cascades. For a detailed review in the area, we refer to the paper by Pei and Makse [PM13].

5.3.2 *Influence Propagation Models and Influence Maximization in Networks*

Let $G = (V, E)$ be a graph and let A be the initial seed set of nodes that are set to active and from which the diffusion starts. Similar to the SIR epidemic model described in Section 5.2.2, in influence propagation the diffusion starts from the set A of activated nodes and spreads over the network. Let $\sigma(A)$ denotes the expected number of influenced (i.e., active) nodes at the end of the process. Next we briefly describe two widely used models for influence propagation.

INDEPENDENT CASCADE MODEL. This model can be considered as a special case of the SIR epidemic model. Let A_t be the set of active nodes at step t. Then, at step $t+1$, each node $v \in A_t$ can activate its inactive neighbors $w \in Nb(v)$ with probability $p(v, w)$. If the activation is succeed, node w will become active at step $t+1$ and it will remain activated until the end of the process. Furthermore, each activated node has a single chance to activate its neighbors. The process is repeated until no more activations are performed and the total number of activated nodes (i.e., influenced) is $\sigma(A)$.

LINEAR THRESHOLD MODEL. In this model, each node v can be influenced by each neighborhood node $w \in Nb(v)$ based on a weight $b_{v,w}$ such that $\sum_{w \in Nb(v)} b_{v,w} \leq 1$. Each node v has a threshold $\theta_v \in [0, 1]$, chosen uniformly at random. Node v becomes active if $\sum_{w \in X(v)} b_{v,w} \geq \theta_v$, where $X(v) \subseteq Nb(v)$ is the set of active neighbors of node v. In other words, threshold θ_v represents the fraction of neighbors of v that should be active in order for v to become active. Similarly to the previous model, the process is repeated until no more nodes become active, and we report the total number of activated nodes $\sigma(A)$.

The *Influence Maximization* problem is defined as follows: for a given parameter k, find a set A of k nodes that maximizes the expected number of influence $\sigma(A)$. Kempe, Kleinberg and Tardos [KT05] proved that this optimization problem is NP-hard for both the Independent Cascade and the Linear Threshold models. However, the optimal solution can be approximated well under both models; in particular, Kempe, Kleinberg and Tardos [KKT03] gave a polynomial time greedy algorithm that can achieve $\left(1 - \frac{1}{e}\right)$ approximation of the optimal solution, where e is the base of the natural logarithm. This algorithm provides theoretical guarantees that its performance can be close to 63% of the optimal one. We refer to the paper by Kempe, Kleinberg and Tardos [KKT15] for more details about the problem of influence maximization.

5.4 K-TRUSS DECOMPOSITION FOR IDENTIFYING INFLUENTIAL NODES

In this section, we present the proposed approach for the identification of individual influential spreaders in networks. Our method is based on the concept of K-truss, a type of cohesive subgraph extracted by the K-truss decomposition [Coho8; WC12; ZP12] presented in Section 5.2.1.

As we described earlier in this Chapter, a K-truss subgraph T_k of G, is defined as the largest subgraph where all edges belong to at least $K-2$ triangles. Respectively, an edge $e \in E$ has truss number $t_{edge}(e) = K$ if it belongs to T_K but not to T_{K+1}. Since the K-truss subgraph is defined on a per edge basis, in the following we extend the definition to the nodes of the graph.

Definition 5.9 (Node truss number $t_{node}(v)$)**.** *Let $G = (V, E)$ be an undirected graph and let $t_{edge}(e), \forall e \in E$ be the truss number of each edge in the graph. We define as truss number of a node $v \in V$, denoted by $t_{node}(v)$, the maximum truss number of its incident edges, i.e., $t_{node}(v) = \max\{t_{edge}(e),\ e = (v,u) \forall\ u \in Nb(v)\}$.*

Let $\mathcal{T}$ denotes the set of nodes with the maximum node truss number t_{node}. In fact, these nodes correspond to the nodes of the maximal K-truss subgraph

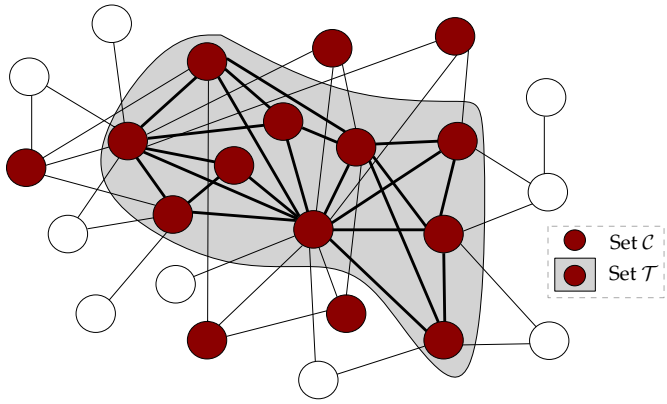


Figure 5.2: Schematic representation of the maximal k-core and K-truss subgraphs of a graph. The red colored nodes correspond to the 3-core subgraph of the graph (set $\mathcal{C}$); the gray shadowed region indicate the 4-truss subgraph (set $\mathcal{T}$).

of the graph. In this work, we argue that this set contains *highly influential nodes* with good spreading properties.

It has been shown that the maximal k-core and K-truss subgraphs (i.e., maximum values for k, K) overlap, with the latter being a subgraph of the former; in fact, K-truss represents the *core* of a k-core that filters out less important information (see Fig. 5.2 for an example). Building upon the fact that the nodes belonging to the maximal k-core of the graph perform good spreading properties [Kit+10], here we further refine this set of the most influential nodes, showing that the nodes having maximum node truss number (i.e., set $\mathcal{T}$ defined above) perform even better, leading to faster and wider epidemic spreading.

To study the spreading process and evaluate the performance of the nodes extracted by the K-truss decomposition method, we apply the SIR model defined in Section 5.2.2. Algorithm 5.2 presents the steps of the proposed framework for (i) selecting the initial node that will trigger the epidemic (cascade) and (ii) evaluate the impact of this individual node with respect to the epidemic spreading under the SIR model.

Initially, we set one node to be in the infected state I. This node corresponds to our single spreader, that is chosen by the K-truss decomposition method (in general, the initial node can be any node of the graph; as we

Algorithm 5.2 Identify nodes and evaluate spreading process

Input: Undirected graph $G = (V, E)$, parameters β, γ
Output: Size of infected population M_v for cascade triggered by node v

Select node $v \in \mathcal{T}$
$State(v) \leftarrow I$, $State(V \setminus v) \leftarrow S$ /* Initialization steps */
$I(0) \leftarrow \{v\}$, $S(0) \leftarrow V \setminus v$, $R(0) \leftarrow \emptyset$
$t \leftarrow 0$
repeat
  $t \leftarrow t + 1$
  $I(t) \leftarrow \emptyset$, $R(t) \leftarrow \emptyset$
  for each node $w \in V$ **do**
    /* Infected (I) nodes can infect susceptible neighbors */
    if $State(w) = I$ **then**
      for each node $z \in \{Nb(w) : State(z) = S\}$ **do**
        $\Pr(State(z) \leftarrow I) = \beta$ (also $I(t) \leftarrow I(t) \cup \{z\}$)
      end for
    end if
    /* Nodes that got infected at previous time steps can recover (R) */
    if $State(w) = I$ and $w \notin I(t)$ **then**
      $\Pr(State(w) \leftarrow R) = \gamma$ (also $R(t) \leftarrow R(t) \cup \{w\}$)
    end if
  end for
until $I(t) = \emptyset$ /* No more infected nodes left */
return $M_v \leftarrow I(1) \cup I(2) \cup \ldots \cup I(t)$

will present later in the experimental evaluation, we perform the same procedure for the baseline methods). The rest of the nodes are assigned to the susceptible state S. At each time step, the infected nodes can infect their susceptible neighbors with probability β (i.e., infection rate). Furthermore, the nodes that have been previously infected can recover from the disease with probability γ (i.e., recovery rate). The process is repeated until no more new nodes get infected. That way, the size of the infected population M_v for the cascade triggered by node $v \in \mathcal{T}$, is the union of all the infected nodes from the beginning of the process.

COMPUTATIONAL COMPLEXITY. As we described in Section 5.2.1, the time complexity of the K-truss decomposition is proportional to $\mathcal{O}(m^{1.5})$, since it requires the computation of the number of triangles that each node participates to. This is actually the main weak point of this method, compared

to the widely used k-core decomposition. However, in this work, we are mainly interested in the nodes that belong to the maximal K-truss subgraph. By taking into account Proposition 5.8 which states that a K-truss subgraph is contained within a $(K-1)$-core subgraph, we can speedup the computation by firstly reducing the graph to its maximal core and then performing further refinements to extract the K-truss subgraph [Coho8].

PARAMETER SETTINGS. In the SIR model used to simulate the epidemic spreading, one has to set values for parameters β and γ. As we described in Section 5.2.2, parameter β concerns the probability that an already infected node will infect a susceptible neighbor, while parameter γ describes the probability of an infected node to enter the recovered state R where the nodes are immunized and cannot be infected again in the future. Parameter β directly controls the spreading process; setting high β values, a relatively large fraction of the nodes will be infected and thus the role of individual nodes in the spreading process is diminished. In other words, large infecting probability will trigger a cascade that is able to cover most part of the network, independently of the source node that initiated the process. In fact, parameters β and γ define the epidemic threshold, a quantity that determines whether the epidemic will be spread to the network or will die out early [BBVo8].

Definition 5.10 (Epidemic threshold τ)**.** *The epidemic threshold τ is defined as a value such that*

$$\frac{\beta}{\gamma} < \tau \Rightarrow \quad \textit{infection dies out over time,}$$
$$\frac{\beta}{\gamma} > \tau \Rightarrow \quad \textit{infection becomes an epidemic.}$$

The epidemic threshold depends on the spreading model that is under consideration as well as on the properties of the underlying graph. In the related literature, several epidemic threshold conditions have been proposed for graphs of various properties [Cha+o8; Wan+o3; PSVo2; BnPSo2; CPS10;

BBV08]. In our work, we adopt the estimation proposed by Chakrabarti, Wang et al. [Cha+08; Wan+03] and Prakash et al. [Pra+11; Pra+12], in which the epidemic threshold is

$$\tau = \frac{1}{\lambda_1}, \tag{5.2}$$

where λ_1 is the largest eigenvalue of the adjacency matrix $\mathbf{A}$ of the graph. Since the graphs that we consider here are undirected, all the eigenvalues will be real and $\lambda_1 > 0$ [Chu97; Mie11]. The leading eigenvalue is also known as *spectral radius* [BH12] and is related to structural and connectivity properties of the graph. That way, we set parameter β close to the epidemic threshold τ of the graph, in order to reduce the effect of the spreading model on the number of infected nodes. Parameter γ is set to a value close to one ($\gamma = 0.8$ in our experimental results). As we will present in Section 5.5, we have performed experiments with several values of β and γ and the results are persistent concerning the comparison of the proposed method to other baselines.

5.5 EXPERIMENTAL EVALUATION

In this Section, we present experimental results concerning the performance of the proposed method for the identification of influential nodes.

5.5.1 *Datasets and Baseline Methods*

DATASETS. We have performed experiments with several real-world networks. Table 5.2 presents the datasets used in our study, along with properties that will be described later. A detailed description of the datasets is presented in Section 2.4 of Chapter 2.

BASELINE METHODS. In the experimental results that follow, we are comparing the spreading performance of the nodes belonging to the set $\mathcal{T}$ (**truss** method), to those belonging to the set $\mathcal{C} - \mathcal{T}$ (**core** method), i.e., the nodes belonging to the maximal k-core excluding those that belong to the

Network Name	Nodes	Edges	k_{max}	K_{max}	$\lvert\mathcal{C}\rvert - \lvert\mathcal{T}\rvert$	$\lvert\mathcal{T}\rvert$	τ
EMAIL-ENRON	33,696	180,811	43	22	230	45	0.00840
EPINIONS	75,877	405,739	67	33	425	61	0.00540
WIKI-VOTE	7,066	100,736	53	23	286	50	0.00720
EMAIL-EUALL	224,832	340,795	37	20	230	62	0.00970
SLASHDOT	82,168	582,533	55	36	38	96	0.00074
WIKI-TALK	2,388,953	4,656,682	131	53	463	237	0.00870

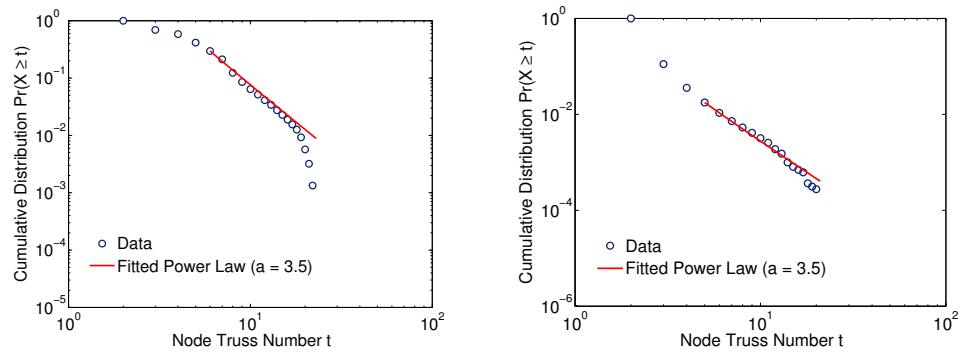
Table 5.2: Properties of the real-world graphs used in this study (Table 5.1 provides definitions of the symbols). $k_{\max}$ and $K_{\max}$ denote the maximum k-core and K-truss numbers respectively (as produced by the decompositions); $|\mathcal{T}|$ represents the number of nodes belonging to set $\mathcal{T}$; $|\mathcal{C}| - |\mathcal{T}|$ represents the number of the nodes belonging to set $\mathcal{C}$, excluding the nodes that belong to set $\mathcal{T}$; τ is the epidemic threshold of the graph.

maximal K-truss of the graph – since $\mathcal{T}$ is subset of $\mathcal{C}$, as discussed in Section 5.2.1. The **core** method constitutes the basic baseline approach, since it has been shown that outperforms other well known node importance criteria such as betweenness centrality [Kit+10].

For completeness in the presentation, we also compare the spreading capabilities of the nodes that belong to the maximal K-truss subgraph to those belonging to the set $\mathcal{D}$ that contains the highest degree nodes in the graph (**top degree** method); we choose $|\mathcal{C}| - |\mathcal{T}|$ high degree nodes to achieve fair comparison between the different methods.

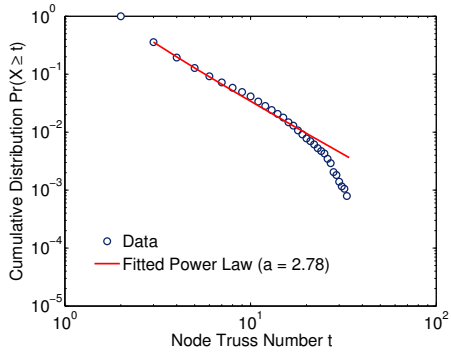
5.5.2 *Characteristics of K-truss Subgraphs*

Before presenting the results about the spreading properties of the proposed **truss** method, we present high-level characteristics of the K-truss decomposition applied on the graphs of Table 5.2. Initially, we have examined the distribution of the node truss numbers t_{node} of the graphs presented in Table 5.2 and the results are depicted in Fig. 5.3. Each plot shows the complementary cumulative distribution function (CCDF) of the nodes' truss number in log-log scale. As we can observe, in most of the cases the distribution is skewed, indicating that very few nodes have high truss number; the majority

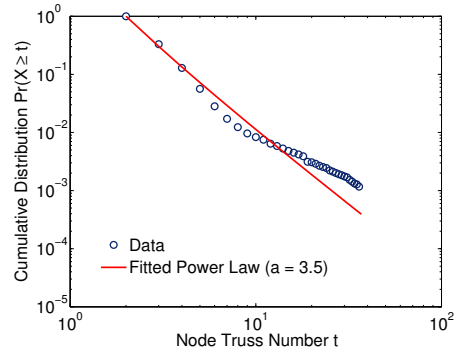


(a) EMAIL-ENRON

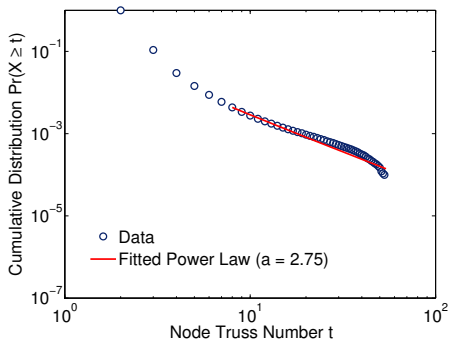
(b) EMAIL-EUALL



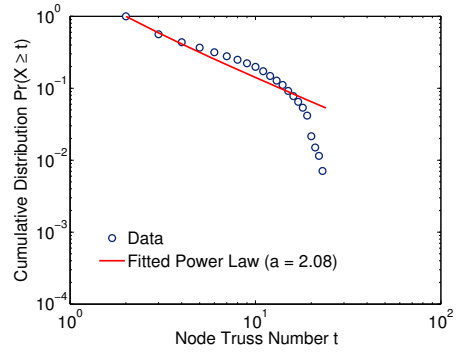
(c) EPINIONS



(d) SLASHDOT



(e) WIKI-TALK



(f) WIKI-VOTE

Figure 5.3: Complementary cumulative truss number distribution function. Each plot depicts the distribution of the truss numbers for the nodes of the graph on log-log scale. The red line corresponds to the fitted power-law distribution.

of the nodes belong to "low" K-truss subgraphs, i.e., small values of parameter K of the decomposition. We have also fitted a power-law distribution [CSN09] to the data (red colored line) and the exponent is shown in Fig. 5.3. Here, we do not claim that the truss number distribution is fully captured by a power-law; nevertheless, it corresponds to heavy-tailed distribution and this fact can help us to better understand the underlying properties of the data. In our case, this means that we can reduce the graph into a subgraph with exponentially smaller size and try to locate influential spreaders within this subgraph. Note that, a similar property has also been observed for the core numbers produced by the k-core decomposition (Fig. 4.3 in Chapter 4).

In addition to that, we have examined the maximum level of the K-truss decomposition, i.e., value $K_{\max}$, for the various graphs. As we can observe from Table 5.2, $K_{\max}$ values vary from dataset to dataset, but compared to the $k_{\max}$ values of the k-core decomposition, they tend to be much smaller. This is rather expected since the K-truss decomposition relies on triangle participation, which is a more strict criterion compared to node degree. This last point is also a justification for the differences on the number of nodes belonging to sets $\mathcal{T}$ and $\mathcal{C}$. Although these sets are overlapping, the one that corresponds to K-truss has significantly smaller size compared to the maximal k-core subgraph. This was also one of the motivations of the work presented in this Chapter; since the nodes of the maximal k-core subgraph perform well in information spreading, can we further refine this set by selecting a small subset that achieves even better spreading properties?

5.5.3 *Evaluating the Spreading Performance*

Next, we describe the experimental results concerning the performance of the proposed technique. To evaluate the spreading efficiency of the methods, we perform the SIR simulation starting from a single node each time, as described in Algorithm 5.2, and we focus on the following quantities:

(i) The number of nodes that become infected at each time step of the process and the corresponding cumulative one.

(ii) The total number of infected nodes at the end of the epidemic.

(iii) The time step where the epidemic fades out.

For each node, we repeat the simulation 100 times (10 times for the WIKI-TALK graph due to its large size) and report the average behavior. In each case, we repeat the above for all the respective nodes and calculate the average behavior for the nodes of each set (**truss** method versus the two baselines **core** and **top degree**).

The experimental results are shown in Tables 5.3 and 5.4. For this experiment, we set parameter β of the SIR model close to the epidemic threshold of each graph, as it is shown in Table 5.2 and parameter $\gamma = 0.8$. Table 5.3 shows the number of the newly infected nodes for the first ten time steps of the spreading process, which we consider as the outbreak of the epidemic, while Table 5.4 depicts the cumulative number of infected nodes per step. We also report the total number of nodes that were infected at the end of the process (*Final step*) and the time step where the epidemic dies out (*Max step*).

As we can observe, the **truss** method achieves significantly higher infection rate during the first steps of the epidemic. Furthermore, in almost all cases, the total number of infected nodes at the end of the process (*Final step*) is larger, while the fade out occurs earlier (*Max step*). Lastly, as we discussed above, the number of nodes in the truss set $\mathcal{T}$ is much smaller compared to the set $\mathcal{C} - \mathcal{T}$ (Table 5.2). By refining significantly the set of influential nodes in truss set $\mathcal{T}$, the "weaker" spreaders of $\mathcal{C}$ are left in core set $\mathcal{C} - \mathcal{T}$, explaining the inferior behavior of the **core** method compared to **top degree**.

Some small deviations from this behavior are observed in the SLASHDOT and WIKI-TALK graphs. In the SLASHDOT graph, the best performance is achieved by the **top degree** method, which from the very first steps is able to infect a larger amount of nodes. In the case of the WIKI-TALK graph, although the total number of infected nodes at the end (*Final step*) of the epidemic is almost the same for all methods, the proposed **truss** method performs quite effectively at the first steps of the process. In fact, it significantly outperforms both baseline methods achieving an increase of almost 23% on the cumulative number of infected nodes compared to both **core** and **top degree** methods, at the sixth step of the process.

		Time Step											
	Method	2	3	4	5	6	7	8	9	10	*Final step*	σ	*Max step*
Email-Enron	**truss**	8.44	18.58	46.66	104.11	204.08	328.39	418.77	425.06	355.84	2,596.52	136.7	33
	core	4.78	12.82	31.97	73.77	152.55	264.36	367.28	403.98	364.13	2,465.60	199.6	37
	top degree	6.89	13.87	34.13	76.67	155.48	264.13	360.89	394.37	357.08	2,471.67	354.8	36
Epinions	**truss**	4.17	9.25	19.70	39.56	75.04	130.48	204.14	278.69	329.08	2,567.69	227.8	37
	core	3.45	7.18	14.72	29.11	55.27	98.11	158.56	226.17	280.03	2,325.37	327.2	43
	top degree	4.22	7.94	16.03	31.32	58.84	103.91	166.23	234.96	289.49	2,414.99	331.7	47
Wiki-Vote	**truss**	2.92	4.37	6.92	10.43	15.27	21.63	28.73	35.93	42.46	560.66	114.9	52
	core	1.92	3.07	4.78	7.22	10.65	15.18	20.66	26.70	32.40	466.01	104.5	57
	top degree	2.43	3.53	5.46	8.17	12.05	17.04	23.05	29.49	35.55	502.88	104.5	62
Email-EuAll	**truss**	11.62	28.04	62.25	127.79	240.97	405.53	584.87	705.89	725.42	5,018.52	487.94	36
	core	9.85	18.69	40.82	82.28	158.72	279.41	433.81	574.97	644.76	4,579.84	498.71	38
	top degree	17.96	16.74	39.93	73.66	144.69	384.07	503.18	565.06	548.25	4,137.56	1,174.84	39
Slashdot	**truss**	5.36	20.57	66.21	188.52	461.35	917.2	1,390.52	1,571.97	1,359.99	8,207.46	368.37	32
	core	6.48	19.68	61.13	168.36	410.19	820.77	1,272.29	1,486.5	1,344.33	8,002.76	518.43	32
	top degree	13.95	27.88	83.29	204.60	483.95	940.49	1,426.81	1,616.55	1,403.80	8,489.45	59.01	32
Wiki-Talk	**truss**	64.21	435.79	3,259.05	16,227.25	34,543.23	23,818.06	9,853.84	3,487.65	1,186.41	93,491.81	476.22	21
	core	41.77	269.96	2,027.69	11,169.2	31,223.21	28,732.06	13,055.45	4,805.11	1,664.52	93,496.50	767.35	23
	top degree	88.84	324.11	2,475.01	11,718.28	29,694.45	27,009.05	13,720.15	5,396.45	1,937.89	93,411.18	1,166.77	24

Table 5.3: Average number of infected nodes per step of the SIR model using β close to the epidemic threshold of each graph and $\gamma = 0.8$. At the *Final step* column, we show the total number of infected nodes at the end of the process (*Max step*), with standard deviation σ.

		Time Step											
	Method	2	3	4	5	6	7	8	9	10	*Final step*	σ	*Max step*
Email-Enron	**truss**	9.44	28.03	74.69	178.80	382.88	711.27	1,130.05	1,555.11	1,910.95	2,596.52	136.7	33
	core	5.78	18.60	50.57	124.35	276.90	541.26	908.54	1,312.52	1,676.65	2,465.60	199.6	37
	top degree	7.89	21.76	55.90	132.57	288.05	552.18	913.07	1,307.45	1,664.53	2,471.67	354.8	36
Epinions	**truss**	5.17	14.42	34.13	73.69	148.74	279.23	483.37	762.06	1,091.14	2,567.69	227.8	37
	core	4.45	11.64	26.36	55.48	110.75	208.87	367.43	593.59	873.62	2,325.37	327.2	43
	top degree	5.22	13.16	29.20	60.52	119.36	223.27	389.49	624.46	913.95	2,414.99	331.7	47
Wiki-Vote	**truss**	3.92	8.30	15.23	25.66	40.94	62.57	91.31	127.25	169.71	560.66	114.9	52
	core	2.92	5.99	10.78	18.01	28.66	43.85	64.50	91.20	123.60	466.01	104.5	57
	top degree	3.43	6.96	12.43	20.61	32.66	49.70	72.75	102.25	137.81	502.88	104.5	62
Email-EuAll	**truss**	12.62	40.66	102.92	203.72	471.69	877.22	1,462.10	2,168.00	2,893.43	5,018.52	487.94	36
	core	10.85	29.55	70.37	152.65	311.38	590.79	1,024.60	1,599.57	2,244.34	4,579.84	498.71	38
	top degree	18.96	35.71	75.64	149.30	294.00	543.45	927.52	1,430.70	1,995.77	4,137.56	1,174.84	39
Slashdot	**truss**	6.36	26.93	93.14	281.67	743.03	1,660.23	3,050.75	4,622.73	5,982.73	8,207.46	368.37	32
	core	7.48	27.17	88.31	256.67	666.86	1,487.64	2,759.93	4,246.43	5,590.76	8,002.76	518.43	32
	top degree	14.95	42.84	126.13	330.74	814.69	1,755.18	3,181.99	4,798.55	6,202.35	8,489.45	59.01	32
Wiki-Talk	**truss**	65.21	501.00	3,760.06	19,987.31	54,530.55	78,348.62	88,202.46	91,690.11	92,876.53	93,491.81	476.22	21
	core	42.77	312.74	2,340.43	13,509.64	44,732.85	73,104.92	86,160.38	90,965.49	92,630.01	93,496.5	767.35	23
	top degree	89.84	413.95	2,888.96	14,607.24	44,301.69	71,310.74	85,030.90	90,427.35	92,365.25	93,411.18	1,166.77	24

Table 5.4: Cumulative number of infected nodes per step of the SIR model using β close to the epidemic threshold of each graph and $\gamma = 0.8$. At the *Final step* column, we show the total number of infected nodes at the end of the process (*Max step*), with standard deviation σ.

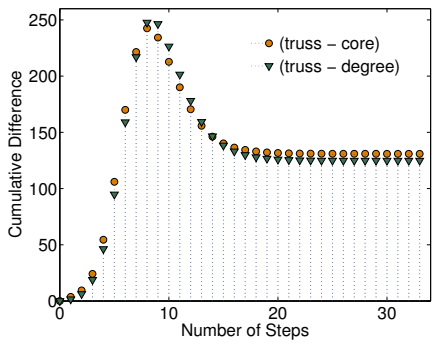
We have also computed the cumulative difference of the number of infected nodes per step achieved by the methods. Let $I_t^{\textbf{truss}}$ be the number of infected nodes at step t achieved by the **truss** method (similar for **core** and **top degree**). We define the cumulative difference for the **truss** and **core** methods at step t as

$$D_t^{\textbf{truss-core}} = \underset{z=1\ldots t}{\text{cumsum}}(I_z^{\textbf{truss}} - I_z^{\textbf{core}}). \tag{5.3}$$

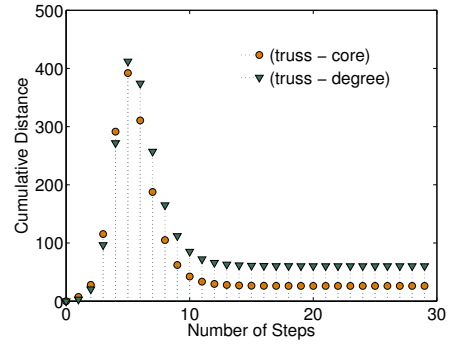
Similarly, we can define the same quantity for the **truss** vs. **top degree** methods. The results are shown in Figures 5.4 and 5.5. For each graph, we have performed experiments for two values of parameter β and $\gamma = 0.8$. We observe that the cumulative difference of the number of nodes that are being infected at every step is always larger between **truss** and **core** than between **truss** and **top degree**. Both differences increase during the outbreak of the epidemic until they stabilize to the number of nodes which is actually the final difference of the number of nodes that got infected (i.e., entered state I) during the epidemic process of the two compared methods. Clearly, as in almost all cases the differences are always above zero, one can conclude to the effectiveness of information diffusion when the spreading is triggered by the nodes that belong to the maximal K-truss subgraph.

5.5.4 *Comparison to the Optimal Spreading*

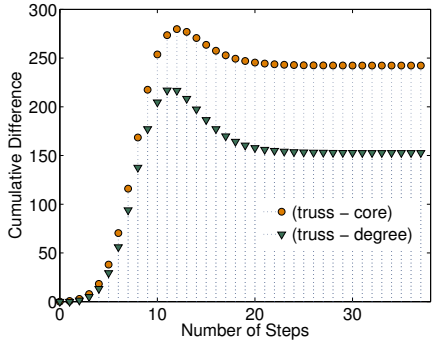
Since we lack ground-truth information about the best spreaders in the network, to further study the performance of the proposed K-truss decomposition method, we have examined the spreading achieved by each node of the graph. More precisely, we set each node $v \in V$ at the state I and simulate the spreading capabilities of this node using the SIR model, as described earlier. Let M_v, $v \in V$ be the size of the population that is infected by the epidemic triggered by node v (average value over multiple executions of the model). Figure 5.6 depicts the distribution of the nodes with respect to the infection size M, for the Email-Enron and Wiki-Vote graphs (parameter β of the SIR model was set to $\beta = 0.01$ for this experiment). In both cases, the axes of the plot have been set to logarithmic scale. As we can observe,



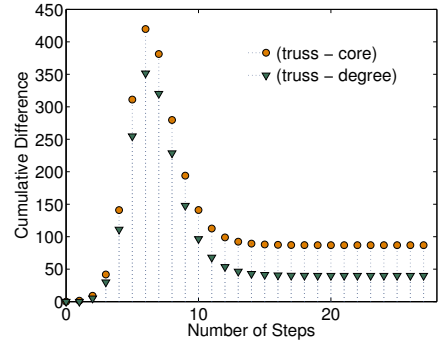
(a) EMAIL-ENRON: $\beta = 0.01$



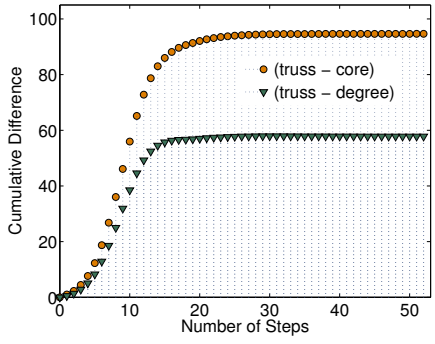
(b) EMAIL-ENRON: $\beta = 0.03$



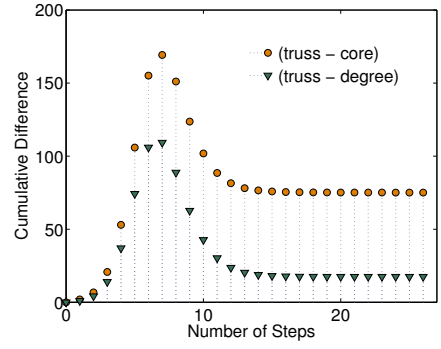
(c) EPINIONS: $\beta = 0.007$



(d) EPINIONS: $\beta = 0.01$

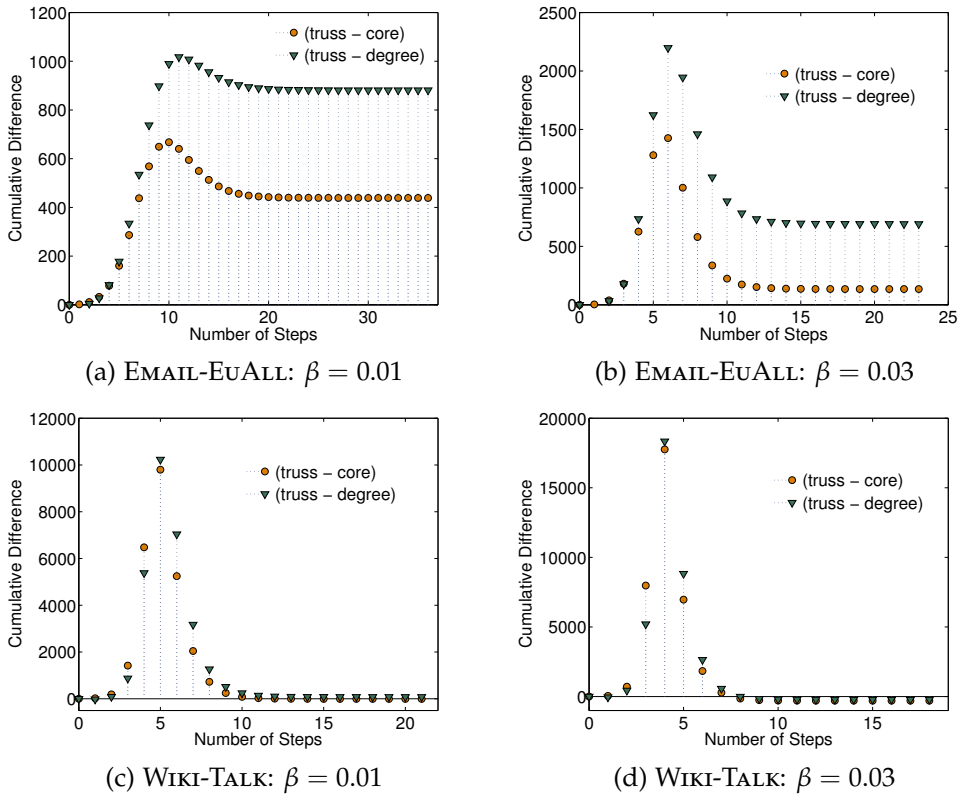


(e) WIKI-VOTE: $\beta = 0.009$



(f) WIKI-VOTE: $\beta = 0.01$

Figure 5.4: Cumulative difference of the infected nodes per step achieved by the **truss** method vs. the **core** (truss - core) and **top degree** (truss - degree) methods. Parameter γ of the SIR models is set to $\gamma = 0.8$. Continued in Fig. 5.5.



(a) EMAIL-EUALL: $\beta = 0.01$

(b) EMAIL-EUALL: $\beta = 0.03$

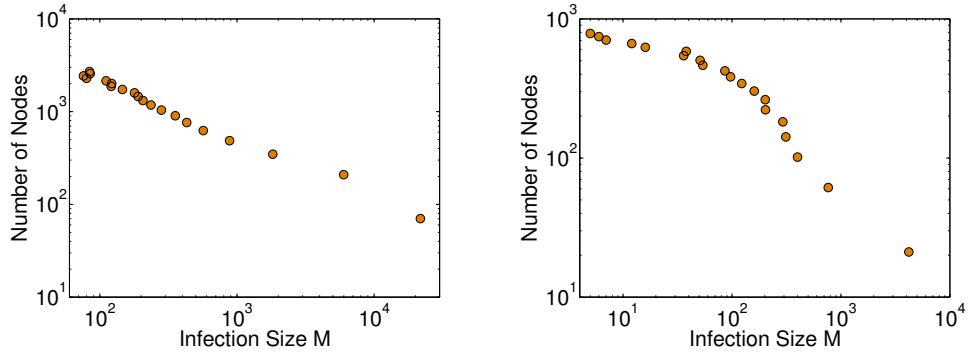
(c) WIKI-TALK: $\beta = 0.01$

(d) WIKI-TALK: $\beta = 0.03$

Figure 5.5: Cumulative difference of the infected nodes per step achieved by the **truss** method vs. the **core** (truss - core) and **top degree** (truss - degree) methods. Parameter γ of the SIR model is set to $\gamma = 0.8$.

the distribution of the infection size M is skewed; only a small percentage of nodes are highly influential, while the majority of the nodes are able to infect only a small portion of the graph (small values of infection size M). Thus, our goal is to examine how the nodes detected by the K-truss decomposition are distributed on this small subset of spreading-efficient nodes. Note that, similar observations have been made for the rest graphs of Table 5.2.

To that end, we rank the nodes $v \in V$ of the graph, according to the infection size M_v. Let



(a) EMAIL-ENRON (b) WIKI-VOTE

Figure 5.6: Spreading distribution of the nodes in the network, in log-log scale. The horizontal axis corresponds to the infection size M achieved by each node in the graph, after a binning process. The vertical axis captures the number of nodes that fall on each bin. Observe that only a small percentage of nodes achieves high spreading. In both cases, we have set $\beta = 0.01$ in the SIR model.

$$OPT_1 \stackrel{\Delta}{=} \underset{v \in V}{\text{argmax}} \; M_v \tag{5.4}$$

be the node that achieves that highest infection size M among all nodes in the graph, i.e., $OPT_1 \geq OPT_2 \geq \ldots \geq OPT_{|V|}$. In order to examine how the nodes detected by the K-truss decomposition are distributed among the most efficient (optimal) spreaders, we consider a variable size window W over the ranked nodes, as shown in Fig. 5.7, and define $P_W^{\mathcal{T}}$ to be the fraction of nodes of set $\mathcal{T}$ that can be found within W as follows:

$$P_W^{\mathcal{T}} = \frac{|T_W|/|\mathcal{T}|}{|W|/|V|}, \tag{5.5}$$

where T_W is the set of nodes $v \in \mathcal{T}$ that are located in the window W of size $|W|$ (in a similar way, we can define $P_W^{\mathcal{C}}$ for the nodes of the maximal k-core subgraph). We are interested to examine how the quantities $P_W^{\mathcal{T}}$ and $P_W^{\mathcal{C}}$ behave with respect to the size of the window W.

Figure 5.8 depicts the distribution of the top-truss $P_W^{\mathcal{T}}$ and top-core $P_W^{\mathcal{C}}$ nodes, for various sizes of window W (i.e., fractions of the most efficient

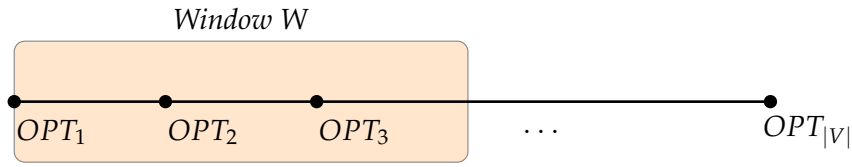


Figure 5.7: Schematic representation of the ranking of nodes based on the spreading that they achieve.

spreaders). As we can observe, for almost all datasets, $P_W^{\mathcal{T}}$ reaches the maximum value (i.e., 100%) relatively early and for small window sizes, compared to $P_W^{\mathcal{C}}$. The maximum value of $P_W^{\mathcal{T}}$ indicates that we have found all the nodes belonging to set $\mathcal{T}$ in the window of fractional size W. An early and intense upward trend of the curve implies that a large fraction of the nodes belonging to the set of interest ($\mathcal{T}$ or $\mathcal{C}$), corresponds to nodes with the best spreading properties on the graph. For example, in the Email-EuAll graph, the maximum for the nodes of set $\mathcal{T}$ is reached in window $W = 1.7\%$, while in the case of set $\mathcal{C}$ in window $W = 2.8\%$. Thus, the nodes detected by the K-truss decomposition method (set $\mathcal{T}$) are better distributed among the most efficient spreaders, compared to those located by the k-core decomposition (set $\mathcal{C}$). A slightly different behavior is observed in the Wiki-Talk and Slashdot graphs; in both graphs, the values of $P_W^{\mathcal{T}}$ and $P_W^{\mathcal{C}}$ are very close to each other for almost all choices of window W, indicating that both sets have almost the same overlap with the set of optimal spreaders. Nevertheless, as we have already presented in Tables 5.3 and 5.4, for those two datasets the spreading performance of the truss nodes achieved during the first steps of the epidemic is much better.

Lastly, we are interested to study the distribution of the nodes' truss number t_{node} with respect to window W. Similar to what described above, we consider a fraction of the best spreaders in the graph (as specified by W) and we examine the distribution of all truss numbers (and not only the maximum one) within it. Since nodes with high truss number are of particular importance here, we have considered groups of nodes as follows:

- Individual groups for each of the top five truss numbers, i.e., $K_{\max}$ to $K_{\max-4}$. That way, the first group contains nodes with truss number

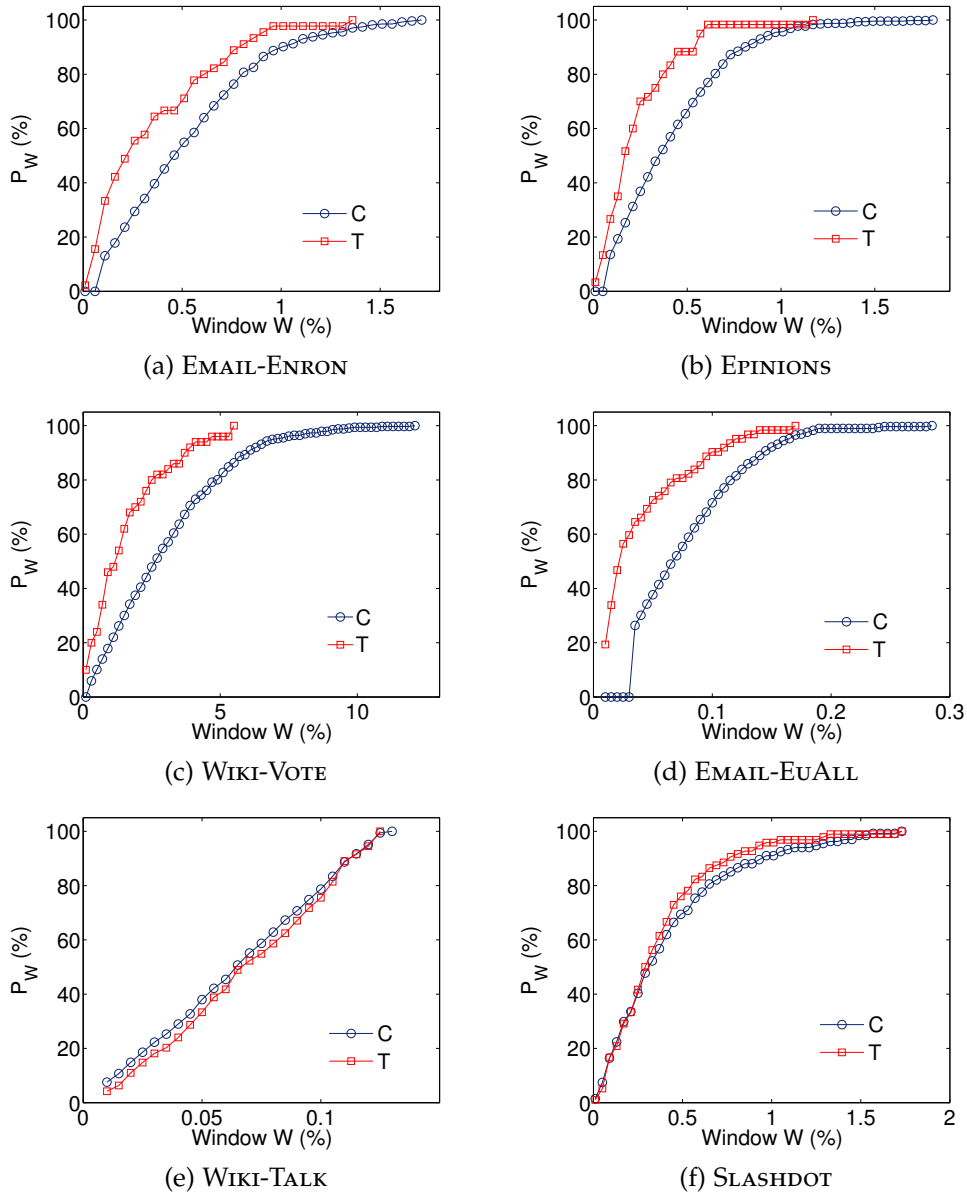


Figure 5.8: Distribution of the top-truss $P_W^{\mathcal{T}}$ and top-core $P_W^{\mathcal{C}}$ nodes among the nodes with optimal spreading properties under a window of size W. Observe that for small values of window size W (i.e., closer to the optimal spreading), the number of top-truss nodes is always higher compared to the number of top-core nodes.

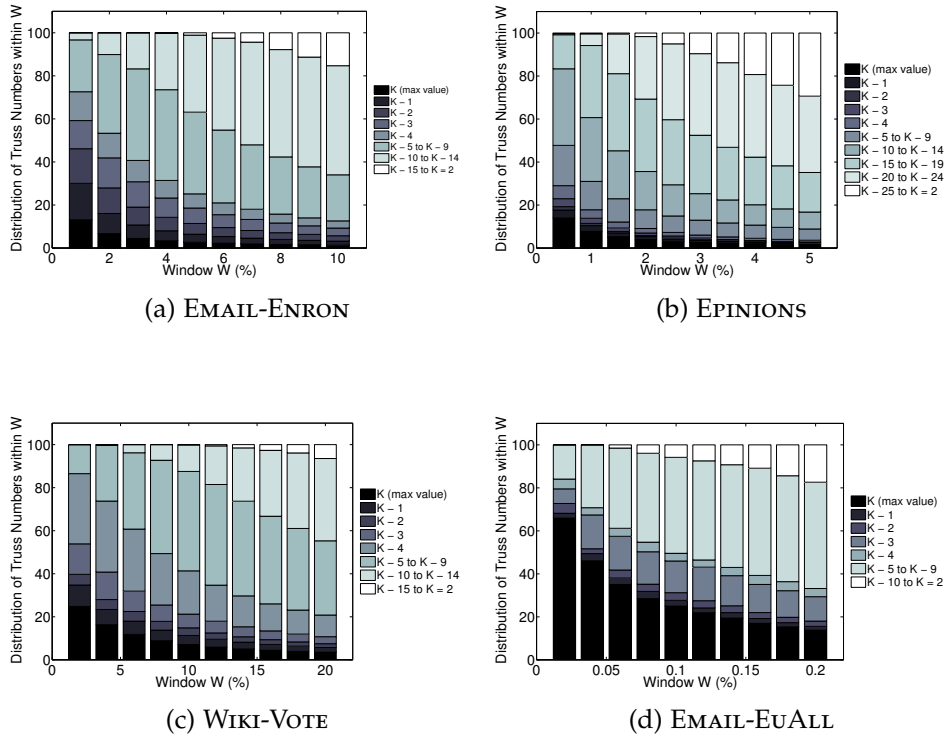
equal to $K_{\max}$, the second group nodes with truss number $K_{\max-1}$ and so on.

- The rest groups concern truss numbers in the range $K_{\max-5}$ to $K = 2$, grouping together five consecutive truss numbers each time. For example, the sixth group contains nodes with truss number in the range $K_{\max-5}$ to $K_{\max-9}$. Note that, the last group may contain less than five truss numbers.

Figure 5.9 depicts the distribution of truss numbers for various values of window W. The colors on each bar correspond to the groups of truss number (darker colors for truss numbers closer to the maximum one). As we can observe in most of the datasets, for small values of window W, a large number of the nodes belong to the first group, i.e., their truss number is maximum one. Since in most of the cases only a tiny fraction of the nodes of the graph belong to the very first groups (i.e., close to $K_{\max}$), even for small window sizes we also observe nodes from groups that correspond to smaller truss numbers. As the window W increases, i.e., deviate from the optimal spreading behavior, groups of smaller truss numbers start to evolve. From these results, it is evident that the truss number is related to the spreading capabilities of the nodes. Until now, we had only examined the effect of the nodes that belong to the maximal K-truss subgraph. However, from this experiment we can conclude that, in general, nodes with high truss number tend to have good spreading properties – with the truss number being highly related to the spreading effect.

5.5.5 *Impact of Infection and Recovery Rate on the Spreading Process*

In the experiments that we have presented so far, parameters β and γ of the SIR model have been set to some constant values; the infection rate β is typically set close to the epidemic threshold of the graph (see also Section 5.4), while the recovery rate is considered constant and always set to $\gamma = 0.8$. In this section, we are interested to examine the impact of the infection and recovery rate on the epidemic spreading achieved by the proposed method (**truss**) and the two baseline methods (**core** and **top degree**). To that end,



(a) EMAIL-ENRON (b) EPINIONS

(c) WIKI-VOTE (d) EMAIL-EUALL

Figure 5.9: Distribution of node's truss number with respect to the ranking of the nodes under their spreading properties. The nodes are classified in groups (different colors) depending on their truss number; for each window size W, we plot the distribution of truss numbers observed within it. Observe that, for small window sizes a large number of the nodes belong to the first group, i.e., their truss number is $K_{\max}$. When the window is enlarged, the groups of lower truss numbers involve a large percentage of the considered nodes.

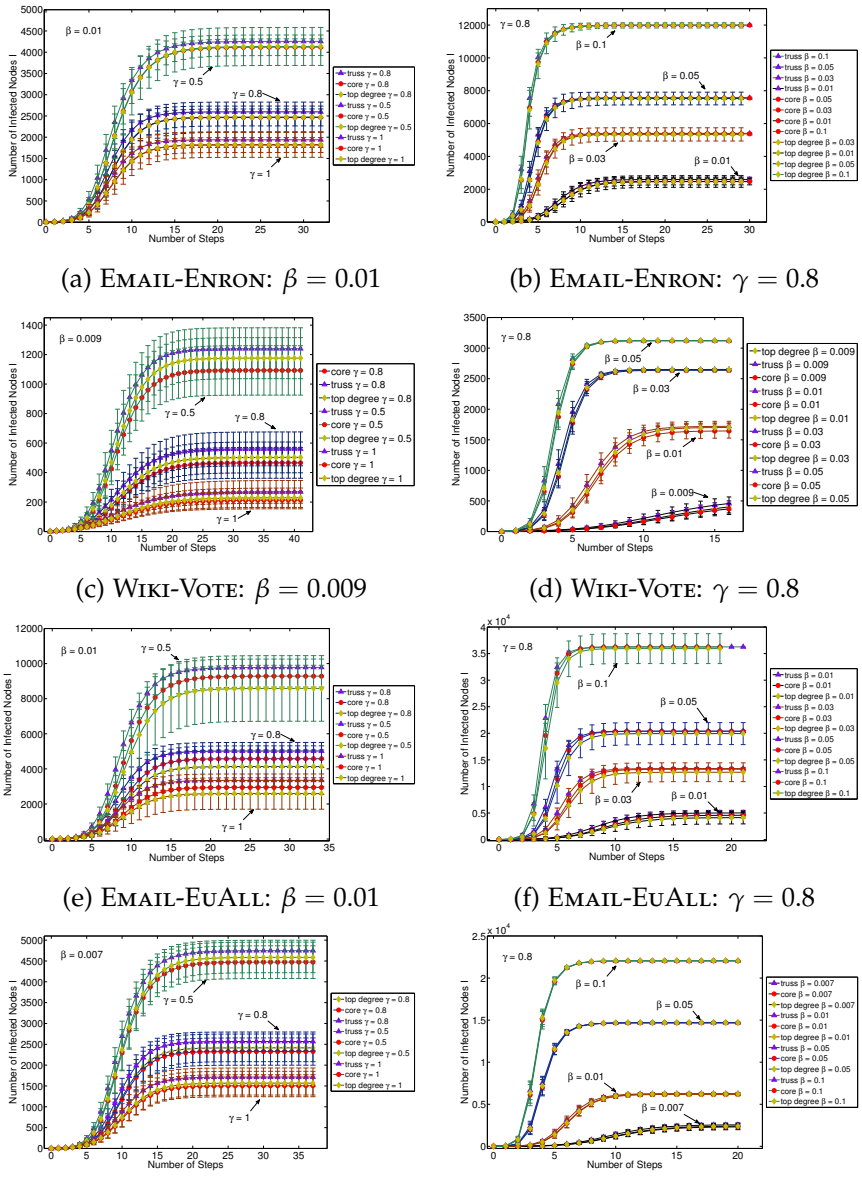
we simulate the spreading process for each of the above methods, setting parameters β and γ as follows:

(i) Parameter β is set close to the epidemic threshold of the graph, while varying parameter $\gamma \in \{0.5, 0.8, 1\}$. Parameter $\gamma = 1$ implies that each infected node moves to the *recovered* (R) state with probability one, in the next step of the model.

(ii) The recovery rate is set to $\gamma = 0.8$, while considering different values of parameter β, always above the epidemic threshold of the graph. As we discussed in Section 5.4, if we consider high values of the infection rate β, a relatively high fraction of nodes will be infected, and thus, the spreading capabilities of individual nodes is diminished.

Figure 5.10 shows the results. In all cases, we have computed the cumulative fraction of infected nodes I_t per step of the process, for each of the three methods, along with the standard deviation (depicted as error bars in the plot). As we can observe, while the recovery probability γ decreases, the number of infected nodes increases both during the first time steps of the process, as well as at the of the epidemic. This behavior is expected since, as we discussed above, with high recovery rate γ most of the nodes will move to the R state, thus being inactive in subsequent iterations of the model. Regarding the performance of the methods, it is evident that the proposed **truss** outperforms both baselines for all different settings of parameter γ.

In the second case where the recovery rate γ is constant, while the infection probability is increasing, the number of infected nodes naturally increases. However, for higher values of β, the total number of infected nodes is almost the same for all methods. This behavior is rather expected; by increasing the infection rate, the importance of individual nodes in the epidemic process is reduced. For these values of β, the difference between the methods can be observed during the outbreak of the epidemic (i.e., first steps of the process), where the **truss** method performs qualitatively better.



(a) EMAIL-ENRON: $\beta = 0.01$

(b) EMAIL-ENRON: $\gamma = 0.8$

(c) WIKI-VOTE: $\beta = 0.009$

(d) WIKI-VOTE: $\gamma = 0.8$

(e) EMAIL-EUALL: $\beta = 0.01$

(f) EMAIL-EUALL: $\gamma = 0.8$

(g) EPINIONS: $\beta = 0.007$

(i) EPINIONS: $\gamma = 0.8$

Figure 5.10: Impact of infection and recovery probabilities of the SIR model on the spreading process: (i) parameter β is set close to the epidemic threshold of the graph, while varying parameter $\gamma \in \{0.5, 0.8, 1\}$; (ii) setting parameter $\gamma = 0.8$ and considering different values of parameter β (always above the epidemic threshold of the graph).

5.6 CONCLUSIONS AND FUTURE WORK

Understanding and controlling the mechanisms that govern spreading processes in complex networks is a fundamental task in various domains, including viral marketing and disease propagation; central to these tasks, is the problem of identification of influential nodes with good spreading properties, that are able to diffuse information to a large part of the network. In this work, we showed that the K-truss decomposition of a network can help towards identifying single influential spreaders. The K-truss subgraph, being a subset of the k-core of the network, contributes in the reduction of the set of privileged spreaders for information diffusion. Using the SIR epidemic model, we have shown that such spreaders will influence a greater part of the network during the first steps of the process, but will also cover a larger portion of it at the end of the epidemic. Additionally, these nodes are well distributed among those that are achieving the optimal spreading in the graph.

As future work, we are interested to study the behavior of our method on time-varying networks, where the network connectivity patterns may change over time. Therefore, it is interesting to examine how these changes affect the influential nodes that have been selected by the K-truss decomposition method. Furthermore, our method is designed to detect single influential spreaders and, as expected, it does not perform well when multiple initial nodes should be selected (e.g., as described in the influence maximization problem in Section 5.3). This is mainly happening because the nodes that belong to the maximal K-truss subgraph share many common neighborhood nodes. Thus, it is of particular importance to examine how to extend the method in order to be able to detect multiple spreaders. Lastly, here we examined the spreading properties of the nodes detected by the K-truss decomposition under the SIR epidemic model. It would be interesting to study the performance of this method under different spreading models, such as the Independent Cascade and Linear Threshold models presented in Section 5.3.

6

A GRAPH-BASED FRAMEWORK FOR TEXT CATEGORIZATION

The focus of the dissertation until now, was mostly on mining and analyzing graphs that correspond to social and collaboration networks, or more generally, data that have an inherent graph structure. However, graphs as powerful modeling tool, can also be used to represent dependencies of terms within a document – leading to a graph-based representation of textual content. The goal of this Chapter is to examine how graph mining techniques can be used to enhance text analytics problems, proposing a graph-based framework for text categorization. In particular, we treat the term weighting task as a node ranking problem in the feature space defined by the graph; the importance of a term to a document is determined based on node centrality criteria. We also introduce novel graph-based global weighting schemes at the document collection level, in order to penalize the importance of commonly used terms. Furthermore, we propose an unsupervised feature selection approach at the graph level, through a trimming process of the less important features based on the properties of the k-core decomposition. Our results indicate that the proposed weighting mechanisms produce more discriminative feature weights for text categorization, outperforming existing frequency-based criteria. Additionally, the term selection process can reduce significantly the feature space of the problem without affecting the accuracy.

6.1 INTRODUCTION

With the rapid growth of the social media and networking platforms, the available textual resources have been increased. Being able to automatically analyze and extract useful information from textual data is an important task with many applications. *Text categorization* or classification (TC) refers

to the supervised learning task of assigning a document to a set of two or more predefined categories (or classes) [Seb02]. TC can be applied in several domains. A well-known application is the one of opinion mining (also known as sentiment analysis), where the goal is to identify subjective information (i.e., positive or negative opinions) from text corpora. Other very well-known applications of TC include spam detection [And+00] and news classification.

The pipeline of the TC problem is similar to any other supervised learning task for text mining and analysis. Each document is modeled using the so-called *Vector Space Model* [BYRN99], i.e., it is represented as a vector in the space defined by the different terms; if a term occurs in the document, the corresponding value in the vector is non-zero. Then main issue here is how to find appropriate weights regarding the importance of each term in a document. Typically, the *Bag-of-Words* model is applied [BYRN99] and a document is represented as a multiset of its terms, disregarding dependencies between the terms; using this model, the importance of a term in a document is mainly determined by the frequency of the term. Although several variants and extensions of this modeling approach have been proposed (e.g., the n-gram model [BYRN99]), the main weakness comes from the underlying term independence assumption. The order of the terms within a document is completely disregarded and any relationship between terms is not taken into account in the categorization task.

In this work, we explore term weighting criteria for TC that go beyond the term independence assumption. The notion of dependencies between terms is introduced via a graph-based document representation model. Under this model, each term is represented as a node in the graph and the edges capture co-occurrence relationships of terms with a specified distance in the document. That way, the term weighting process for the TC task is transformed to a ranking problem in the interconnected feature space defined by the graph. The basic advantage of our approach is that we are able to augment the unigram feature space of the learning task with weights that implicitly consider information about n-grams in the document – as expressed by paths in the graph – without increasing the dimensionality of the problem.

The main contributions of the work can be summarized as follows:

- *Graph-based term weighting schemes*: we adopt a graph-based representation of documents and derive novel term weighting schemes for TC, through a ranking process in the interconnected feature space defined by the graph.
- *Inverse Collection Weight (ICW)*: we propose a novel graph-based, term weighting criterion to penalize the importance of terms across the document collection level.
- *Unsupervised graph-based feature selection*: capitalizing on the graph representation of each document in the collection, we apply the k-core decomposition to extract discriminative terms in an unsupervised manner.
- *Large scale empirical study*: we perform experiments in well-known datasets for document categorization (e.g., Web pages classification and sentiment analysis). Our results indicate that the proposed weighting schemes are able to outperform existing frequency-based ones.

The rest of the Chapter is organized as follows. Section 6.2 reviews the related work on term weighting schemes, text categorization and graph-based text categorization methods as well as graph-based models in text mining, natural language processing (NLP) and information retrieval (IR). Section 6.3 presents preliminary and background concepts that will be used throughout this Chapter. Then, in Section 6.4 we introduce the proposed graph-based framework for TC. Section 6.5 presents the experimental results and finally, in Section 6.6 we conclude and provide further directions for the TC problem based on graph representation of documents.

6.2 RELATED WORK

In this section we review the related work, which can be placed into four main categories: term weighting schemes for document representation, text categorization, graph-based text classification and graph-based methods in

text mining, natural language processing (NLP) and information retrieval (IR).

TERM WEIGHTING SCHEMES. A core aspect in the Vector Space Model for document representation, is how to determine the importance of a term within a document. This is central, and still active, research topic that goes back to the origins of IR; since then, many criteria have been introduced with the most prominent ones being TF, TF-IDF [SB88; Rob04; BYRN99; MRS08] and Okapi BM25 [Rob+96], while some recent ones include N-gram IDF [SHN15]. With the advances on learning techniques for textual data, many of these weighting schemes were considered or extended in order to deal with supervised and unsupervised tasks. Some examples include the Term Frequency - Inverse Corpus Frequency (TF-ICF) [Ree+06] method proposed for document clustering and the TF-RF scheme [Lan+09] used in text categorization. Lan et al. [Lan+] conducted a comparative study of frequency-based term weighting criteria for text categorization; one of their outcomes was that, in many cases, the IDF factor is not significant for the categorization task, leading to no improvement of the performance. It is interesting to point out here that, more specialized approaches have been proposed for specific classification tasks, such as the Delta TF-IDF method that constitutes an extension of TF-IDF for sentiment analysis [MF09]. However, most of the previously proposed frequency-based weights consider the document as a Bag-of-Words; that way, any structural information about the ordering or in general, syntactic relationship of the terms is ignored by the weighting process.

TEXT CATEGORIZATION. TC is one of the most fundamental and well studied tasks in text analytics and a number of diverse approaches have been proposed [Lew92; Seb02; Joa98; Joa99; Lod+02; SL01; Nig+00; Kim+06; Lan+09; DGT02; MN98; Maa+11]. The first step of TC concerns the feature extraction task, i.e., which features will be used to represent the textual content. Typically, the straightforward *Bag-of-Words* approach is adopted, where every document is represented by a feature vector that contains boolean or weighted representation of unigrams or n-grams in general [Für98]. In the

case of weighted feature vectors, various term weighting schemes have been used, with the most well-known ones being TF (Term Frequency), TF-IDF (Term Frequency - Inverse Document Frequency) and several variants of them, as described in the previous paragraph. Although these weighting schemes were initially introduced in the NLP and IR fields, they have also been applied in the TC task. Paltoglou and Thelwall [PT10] reported that, in the case of sentiment analysis, extensions of the TF-IDF weighting schemes introduced in the IR field, can further improve the classification accuracy. A comprehensive review of this area is offered in the article by Sebastiani [Seb02].

GRAPH-BASED TEXT CATEGORIZATION. Close to our work are several text categorization methods that also capitalize on a graph representation of the document. Some techniques rely on graph mining algorithms that are applied to extract frequent subgraphs, which are then used to produce feature vectors for classification [Jia+10; MLK07; Aro+10; Des+05; RKV15]. The basic shortcoming of those methods stems from the computational complexity of the frequent subgraph mining algorithm. Furthermore, most of these methods require from the user to set the *support* parameter, which concerns the frequency of appearance of a subgraph. Wang et al. [WDL05] introduced a term graph model that, contrary to our approach, captures the relationships among terms using frequent itemset mining algorithms. Aery and Chakravarthy [AC05] proposed InfoSift, a graph-matching based method for document classification. Close to our work is the approach followed by Hassan et al. [HMB07], where they proposed to use random walks in order to determine the importance of a term. In this work, we show that we can rely on simpler and easier to compute graph-based criteria – such as the degree of a node – to achieve even better classification results.

GRAPH-BASED TEXT MINING, NLP AND IR. Representing documents as graphs is a well-know approach in many text analytics tasks. Dhillon [Dhi01] proposed to deal with the document clustering problem using a bipartite graph model. The first partition of the graph corresponds to the terms of the collection, while the other to the documents; then, the existence

of an edge indicates the appearance of the term to the document. Then, graph partitioning algorithms can be used to solve the document clustering problem. Still, one question here is how to weight the edges of the graph and in most of the related work, the TF-IDF scheme is adopted. Close to our work are methods proposed for keyword extraction and ad hoc IR. TextRank algorithm, proposed by Mihalcea and Tarau [MT04], was among the first works that considered a random walk model similar to PageRank, over a graph representation of the document, in order to extract representative keywords and sentences. Later, several methods for these tasks were followed [ER04; RV15; Bou13; BBD13; BHTM11; LL08]. Another domain where graph-based term weighting schemes have been applied is the one of ad hoc Information Retrieval [BL12; BL07; RV13]. As we will present later, our approach moves on a similar axis as these techniques, but the application is on the TC task. The interesting reading can refer to [BL12] for a detailed description of graph-based methods in the text domain.

6.3 PRELIMINARIES AND BACKGROUND

In this section, we present preliminary and background concepts that will be used throughout the Chapter. Initially, we briefly discuss the basic formulation of the TC problem, as well as the frequency-based weighting criteria that are derived from the traditional Bag-of-Words model, namely TF and TF-IDF. Then, we introduce the graph-theoretic concepts upon which our framework for TC is built.

6.3.1 *Text Categorization Pipeline and Term Weighting in the Bag-of-Words Model*

Let $\mathcal{D} = \{d_1, d_2, \ldots, d_m\}$ be a collection of documents and let $\mathcal{C} = \{c_1, c_2, \ldots, c_{|\mathcal{C}|}\}$ be the set of predefined categories. Text categorization is considered the task of assigning a boolean value to each pair $(d_i, c_i) \in \mathcal{D} \times \mathcal{C}$, i.e., assigning each document to one or more categories [Seb02]. In this work, we deal with the problem of multi-class, single-label text categorization, where each document is assigned exactly to one category.

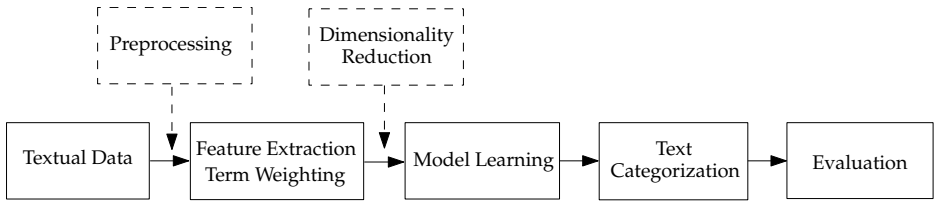


Figure 6.1: Basic pipeline of the Text Categorization task.

The basic pipeline of the TC task is shown in Fig. 6.1. The pipeline that typically is followed to deal with the problem is similar to the one applied in any classification problem; the goal is to learn the parameters of a classifier from a collection of training documents (with known class information) and then to predict the class of unlabeled documents [Bis06].

The first step in the TC process is to transform documents, which typically are strings of characters, into a representation suitable for the learning algorithm and the classification task. Here, we consider the widely used *Vector Space Model*, i.e., the spatial representation in which each document is represented by a vector in the n-dimensional space defined by the terms $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ of the overall vocabulary of the collection [BYRN99]. That way, each document $d_i \in \mathcal{D}$ is represented by a vector of weights $d_i = \{w_{i,1}, w_{i,2}, \ldots, w_{i,n}\}$, where $w_{i,k}$ is the weight of term k in document d_i. The main point here is how to find appropriate *weights* for the terms within a document. In the traditional *Bag-of-Words* model, each document is represented as the bag (multiset) of its words, disregarding the ordering or in general, any potential dependencies between the terms of the document. Under this model, the importance of a term in a document is mainly determined by the frequency of the term. That is, the weight of a term $t \in \mathcal{T}$ within a document $d \in \mathcal{D}$ is based on the *frequency* $tf(t, d)$ of the term in the document (TF weighting scheme). Furthermore, terms that occur frequently in one document but rarely in the rest of the documents, are more likely to be relevant to the topic of the document. This is known as the *inverse document frequency* (IDF) factor, and is computed at the collection level. It is obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient, as follows:

$$idf(t, \mathcal{D}) = \log\left(\frac{m+1}{|\{d \in \mathcal{D} : t \in d\}|}\right), \tag{6.1}$$

where m is the total number of documents in collection $\mathcal{D}$, and the denominator captures the number of documents that term t appears. Then, the TF-IDF scheme is produced by the multiplication of the TF and IDF factors. In this work, we will use the TF-IDF weighting scheme proposed in [SBM96; Sin+99], also called pivoted normalization weighting:

$$tf\text{-}idf(t, d) = \frac{1 + \ln\left(1 + \ln(tf(t, d))\right)}{1 - b + b \times \frac{|d|}{\ell}} \times idf(t, \mathcal{D}), \tag{6.2}$$

where d is the length of the document, ℓ is the average document length and parameter b is set by default to 0.20 (as suggested in [SBM96]). The TF-IDF scoring function captures the intuitions that (i) the frequency of a term in a document is proportional of how representative it is for its content, and (ii) the higher the number of documents a term occurs in, the less discriminating it is (term specificity)[1]. This weighting scheme (and its variants) has been widely used in the TC task; as we will present shortly, the TF and TF-IDF weighting mechanisms will be the main baseline approaches for our experimental evaluation.

6.3.2 *Graph-Theoretic Concepts*

Next, we describe two graph-theoretic concepts upon which our framework is built.

Node Centrality Criteria

Centrality[1] represents a central notion in graph theory and network analysis in general; it constitutes a measure that captures the relative importance of the node in the graph based on specific criteria [New10; EK10]. One important characteristic of the centrality measures is that they consider

1 Several variants of the TF-IDF score have been proposed. See also the description given in Ref. [Seb02].

1 Wikipedia's lemma for *network centrality*: http://en.wikipedia.org/wiki/Centrality.

either *local* information of the graph (e.g., degree centrality, in-degree/out-degree centrality in directed networks, weighted degree in weighted graphs, clustering coefficient) [EK10], or more *global* information, in the sense that the importance of a node is determined by the properties of the node globally in the graph (e.g., PageRank centrality, eigenvector centrality, betweenness centrality, closeness centrality). Let $G = (V, E)$ be a graph (directed or undirected), and let $|V|$, $|E|$ be the number of nodes and edges respectively. Next, we define the basic centrality measures that will be used in the rest of the Chapter [New10].

DEGREE CENTRALITY. The degree centrality is one of the simplest local node importance criteria, which captures the number of neighbors that each node has. Let $\mathcal{N}(i)$ be the set of nodes connected to node i. Then, the degree centrality can be derived based on the following formula:

$$\text{degree\_centrality}(i) = \frac{|\mathcal{N}(i)|}{|V| - 1}. \tag{6.3}$$

IN-DEGREE AND OUT-DEGREE CENTRALITY. Those two centrality measures constitute extensions of the degree centrality in directed networks, where we treat independently the in-degree (number of incoming edges) and out-degree (number of outgoing edges) of each node.

CLOSENESS CENTRALITY. In general, closeness centrality measures how close a node is to all other nodes in the graph. Let $dist(i, j)$ be the shortest path distance between nodes i and j. The closeness centrality of a node i is defined as the inverse of the average shortest path distance from the node to any other node in the graph [OCL08]:

$$\text{closeness}(i) = \frac{|V| - 1}{\sum_{j \in V} dist(i, j)}. \tag{6.4}$$

Contrary to degree centrality, the closeness score is a global metric, in the sense that it combines information from all the nodes of the graph. Here we compute the closeness centrality in the undirected graph.

EIGENVECTOR CENTRALITY. Let $\mathbf{A}$ be the adjacency matrix of an undirected graph G. The eigenvector centrality of node i is defined as

$$u(i) = \frac{1}{\lambda} \sum_j A_{ij} u_j, \tag{6.5}$$

where λ is a constant. Vector $\mathbf{u}$ whose elements are $u_i, \forall i \in V$, is derived from $\mathbf{Au} = \lambda \mathbf{u}$, i.e., the vector that contains the centralities of the nodes of the graph is an eigenvector of the adjacency matrix. Since the centralities should be non-negative, the Peron-Frobenius theorem [GL96; Str88] suggests that $\mathbf{u}$ is the eigenvector that corresponds to the largest eigenvalue λ of $\mathbf{A}$.

k-core Decomposition

The k-core decomposition [Sei83] constitutes an hierarchical decomposition of the graph into nested subgraphs of increased coherence and connectivity properties. The basic idea is to set a threshold on the node degree, say k; nodes that do not satisfy the threshold are removed from the graph. More formally, let $G = (V, E)$ be an undirected graph. A subgraph H of G, denoted by $H \subseteq G$, is defined as the graph that can be obtained from G after removing edges or nodes. Then, a subgraph H of G is defined to be a k-core of G, if it is a maximal connected subgraph of G, in which all nodes have degree at least k within H. The largest value $k_{\max}$ of k for which such a maximal subgraph exists, defines the *maximal k-core subgraph* G_k of graph G.

k-core decomposition has been widely used in many application domains, including community detection and graph visualization. One important point here, is that the maximal k-core subgraph G_k can be computed in $\mathcal{O}(|E|)$ time, i.e., linear to the number of edges of G [BZ03]. Extracting now G_k, can be considered as an *unsupervised* way to extract a dense subgraph of the original graph. More details about the k-core decomposition and its applications can be found in Chapter 2, Section 2.3.

6.4 PROPOSED FRAMEWORK FOR TEXT CATEGORIZATION

In this section, we present the basic parts of the proposed graph-based framework for TC. We adopt the Graph-of-Words document representation, where documents are represented as graphs that capture term co-occurrence relationships within a fixed-size sliding window. Then, we show (i) how to derive meaningful term weighting schemes for TC – both at the document and collection level – and (ii) how to reduce the unigram feature space of the problem by an efficient graph trimming process that excludes terms with low discriminative power.

6.4.1 *Graph Construction*

We model documents as graphs that capture dependencies between terms. More precisely, each document $d \in \mathcal{D}$ is represented by a graph $G_d = (V_d, E_d)$, where the nodes correspond to the terms t of the document and the edges capture co-occurrence relationships between terms within a fixed-size sliding window of size w. That is, for all the terms that co-occur within the window, we add edges between the corresponding nodes of the graph. Note that, the windows are overlapping starting from the first term of the document; at each step, we simply remove the first term of the window and add the new one from the document. In the experiments that have been conducted, we have removed stopwords from the datasets. Thus, the window can expand over different sentences. As graphs constitute rich modeling structures, several parameters about the construction phase need to be specified.

DIRECTED VS. UNDIRECTED GRAPH. One parameter of the model is if the graph representation of the document will be directed or undirected [New10; EK10]. Directed graphs are able to preserve actual flow on a text, while in undirected ones, an edge captures co-occurrence of two terms whatever the respective order between them is. We have tested both choices, observing that undirected graphs perform significantly better; it seems that the actual ordering of terms is not a discriminative factor in TC.

WEIGHTED VS. UNWEIGHTED GRAPH. One approach is to consider weighted graphs [New10; EK10]. That is, the higher the number of co-occurrences of two terms in the document, the higher the weight of the corresponding edge (i.e., the weight of each edge will be equal to the number of co-occurrences of its endpoints). The second and more simple option, is to consider unweighted graphs. Again, in the majority of the cases that we examined, unweighted graphs were a better modeling choice for TC.

SIZE w OF THE SLIDING WINDOW. In the construction of the graph, we add edges between the terms of the document that co-occur within a sliding window of size w. Although by increasing the size of the window we are able to capture co-occurrence relationships between not necessarily nearby terms (similar to the notion of (long) n-grams), the produced graph becomes relatively dense. From our experimental results, we have observed that window of size $w = \{2, 3\}$ give persistently better classification results.

Figure 6.2 shows an example of a graph-based representation of a textual document. Note that, the above procedure, as has been applied in our framework, concerns unigrams; we consider that a similar approach can potentially be applied to build a graph using n-gram features of documents. To summarize, the key point of the graph-based representation for TC is the fact that it deals with the term independence assumption. Even if we consider the n-gram model, still information about the relationship between two different n-grams is not fully captured – as happens in the case of graphs. This has also been noted in other application domains (e.g., IR [BL12; RV13]).

6.4.2 *Term Weighting*

As we have already presented, when the document is represented by the Bag-of-Words model, the term frequency (TF) criterion (or TF-IDF) constitutes the basis for weighting the terms of each document. How this can be done in the graph-based representation? The answer is given by utilizing node centrality criteria of the graph [New10; EK10]. That way, the importance of a term in a document can be inferred by the importance of the corresponding node in the graph. In Section 6.3, we presented well-known centrality criteria that

Data Science is the extraction of knowledge from large volumes of data that are structured or unstructured which is a continuation of the field of data mining and predictive analytics, also known as knowledge discovery and data mining.

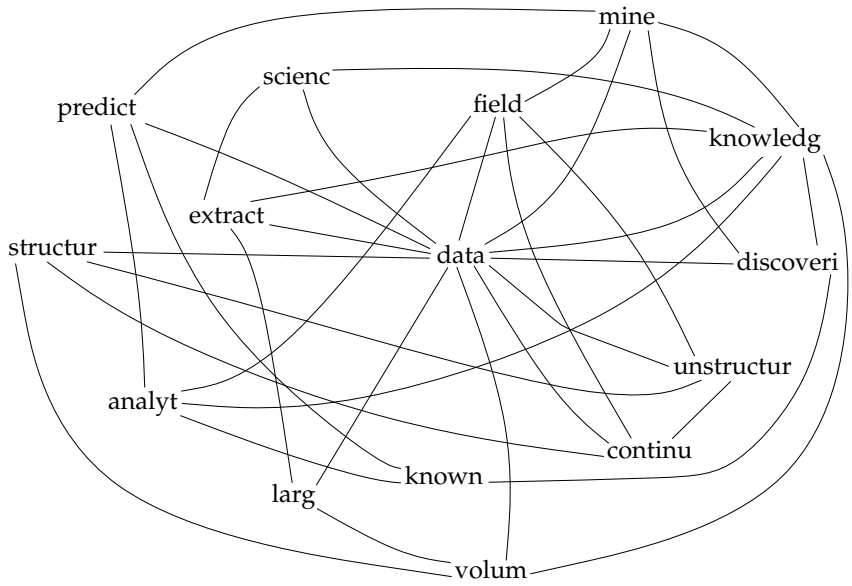


Figure 6.2: Example of graph-based representation of text after applying stemming and stopwords removal (undirected graph, window size $w = 3$).

have been widely used for graph mining and network analysis purposes; here, we propose that those criteria can also be used for weighting terms in the TC task. Although the semantics behind some of them are simple, still they can further improve the classification performance, when applied as term ranking criteria. Let $\tilde{tw}(t,d)$ be the centrality score of term (node) t in the graph representation G_d of document d. Then, similar to the TF term of Eq. (6.2), we can define the basic weighting scheme, denoted as Term Weight (TW), as follows:

$$tw(t,d) = \frac{\tilde{tw}(t,d)}{1 - b + b \times \frac{|d|}{\ell}}. \tag{6.6}$$

Parameter b is a normalization factor concerning the length of the document; we set parameter $b = 0.003$ as suggested by our experimental results (also reported in [RV13]). The interesting point here is that TW can be used along with any centrality criterion in the graph, local or global.

Furthermore, we can extend this weighting scheme by considering information about the inverse document frequency (IDF factor) of the term t in the collection $\mathcal{D}$. That way, we can derive the TW-IDF model as follows:

$$tw\text{-}idf(t,d) = tw(t,d) \times idf(t,\mathcal{D}). \tag{6.7}$$

In fact, TW and TW-IDF constitute suites for graph-based term-weighting schemes and thus, they can be applied in any text analytics task. Some of them have already been explored in graph-based Information Retrieval [RV13; BL12] and keyword extraction [MT04; LCC14; RV15].

A natural question here is what is the additive value of the TW and TW-IDF, compared to the widely used TF and TF-IDF. As we have already discussed, the graph-based representation and the corresponding weighting functions, question the term independence assumption that is imposed by the Bag-of-Words model and is inherited to the frequency-based schemes. The proposed weights are inferred from the interconnection of features (i.e., terms) – as suggested by the graph – and therefore information about n-grams is implicitly captured. That way, the feature space of the learning problem is kept to the one defined by the unique unigrams of our collection (instead of using simultaneously as features all the possible unigrams, bi-

grams, 3-grams, etc.), but the produced term weights incorporate n-gram information through the graph-based representation. In other words, the importance and discriminative power of a term is determined by a ranking process on the corresponding centrality metrics, in the interconnected feature space defined by the graph.

6.4.3 *Inverse Collection Weight (ICW)*

In the case of TF-IDF scheme, the frequency of each term in the document (TF factor) is penalized by the number of documents into which it appears (IDF factor). As we presented above, the same appearance-based penalization mechanism can also be applied to the TW graph-based weight, and according to this the TW-IDF scheme of Eq. (6.7) has been derived. In this section, we introduce the concept of *inverse collection weight* (ICW) – a graph-based criterion to penalize the weight of terms that are "important" across the whole collection of documents. The main concept behind ICW is the *collection level* graph $\mathcal{G}$ – an extension of Graph-of-Words in the collection of documents $\mathcal{D}$.

Definition 6.1 (Collection Level Graph $\mathcal{G}$)**.** *Let $\{G_1, G_2, \ldots, G_d\}_{|\mathcal{D}|}$ be the set of graphs that correspond to all documents $d \in \mathcal{D}$. The collection level graph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ is defined as the union of graphs $G_1 \cup G_2 \cup \ldots \cup G_d$ over all documents in the collection.*

The union of two graphs $H = (V_H, E_H)$ and $W = (V_W, E_W)$ is defined as the union of their node and edge sets, i.e., $H \cup W = (V_H \cup V_W, E_H \cup E_W)$. The number of nodes $|V_\mathcal{G}|$ in graph $\mathcal{G}$ is equal to the number of unique terms in the collection, while the number of edges $|E_\mathcal{G}|$ is equal to the number of unique edges over all document-level graphs (we do not consider weights over the edges of $\mathcal{G}$).

This graph captures the overall dependencies between the terms of the collection; the relative overall importance of a term in the collection will be proportional to the importance of the corresponding node in $\mathcal{G}$. Following similar methodological arguments as used for IDF [Rob04], we define a probability distribution over the nodes of $\mathcal{G}$ (or equivalently, the unique

terms of $\mathcal{D}$), with respect to a centrality (term-weighting in our case) criterion; then, the probability of node (term) t will be

$$\Pr(t) = \frac{tw(t, \mathcal{D})}{\sum_{v \in \mathcal{D}} tw(v, \mathcal{D})}. \tag{6.8}$$

Note that, in Eq. (6.8), we use $\mathcal{D}$ instead of $\mathcal{G}$; we consider that the space defined by the document collection $\mathcal{D}$ is equivalent to the one defined by graph $\mathcal{G}$ with respect to the unique terms of the collection. That way, the notion of $tw(t, \mathcal{D})$ used here is consistent with what described earlier. Based on this, we define the ICW measure as

$$icw(t, \mathcal{D}) = \log \left(\frac{\sum_{v \in \mathcal{D}} tw(v, \mathcal{D})}{tw(t, \mathcal{D})} \right). \tag{6.9}$$

The ICW measure shares common intuition with the *inverse total term frequency* described in [Rob04]. In fact, it can be considered as an extension of the total collection frequency of a term to the graph-based document representation. Furthermore, similar to TW, it can be used along with any node centrality criterion.

Using ICW as a graph-based collection-level term penalization factor, we introduce two new classes of term-to-document weighting mechanisms, namely TW-ICW and TF-ICW. These weighting schemes are derived combining different local (i.e., document-level) and global (i.e., collection-level) criteria as follows:

$$tw\text{-}icw(t, d) = tw(t, d) \times icw(t, \mathcal{D}) \tag{6.10}$$

$$tf\text{-}icw(t, d) = tf(t, d) \times icw(t, \mathcal{D}). \tag{6.11}$$

As we have already discussed, in the case of TW and ICW, any centrality criterion can be applied. However, the computational complexity is a crucial factor that should be taken into account (see Section 6.4.5). Nevertheless, as we have noticed from the experimental evaluation, even using simple and easy-to-compute criteria, we can have good classification performance.

Algorithm 6.1 Unsupervised Graph-Based Feature Selection

Input: List of graphs $\{G_1, G_2, \ldots, G_d\}, d = |\mathcal{D}|$ corresponding to the documents of the collection
Output: Reduced feature set $\widetilde{\mathcal{T}}$
1: $\widetilde{\mathcal{T}} \leftarrow \emptyset$
2: **for** each graph $G \in \{G_1, G_2, \ldots, G_d\}$ **do**
3: $\quad G_k = (V_k, E_k) \leftarrow k$-`core_decomposition`$(G)$
4: $\quad \widetilde{\mathcal{T}} \leftarrow \widetilde{\mathcal{T}} \cup V_k$
5: **end for**

6.4.4 *Unsupervised Feature Selection*

In this section, we describe a method to reduce the number of features (i.e., terms) from the collection, that will be used by the classification algorithm. The idea is to capitalize on the graph representation of each document, in order to extract meaningful and discriminative terms from each one. To this end, we apply the k-core decomposition process on each graph (i.e., document), retaining only the maximal k-core subgraph. More formally, let $\{G_1, G_2, \ldots, G_d\}, d = |\mathcal{D}|$ be the graphs corresponding to the documents of the collection $\mathcal{D}$. Algorithm 6.1 presents the proposed method for feature selection.

The new feature set $\widetilde{\mathcal{T}}$ is obtained as the union of the nodes belonging to each G_k. The main intuition behind this approach is that we can extract representative *keywords* from each document, using them as discriminative features for the classifier. In the past, graph-based keyword selection methods (e.g., [MT04; Bou13]) have been proved quite successful, with the nodes being on the maximal k-core subgraph performing rather well [RV15].

In the related literature, several ways have been proposed to perform feature selection in machine learning tasks. In the case of text categorization, well-known feature selection criteria can be employed, including Information Gain, Mutual Information and the χ^2 statistic [YP97]. However, all these widely used criteria are *supervised* and in many cases computationally intensive. Other, more simple ones, rely on statistical metrics over the terms, such as the Document Frequency (DF) [YP97], and exclude or retain terms accordingly. The main issue here is that we should set a threshold on DF and

keep only those terms that satisfy the threshold. However, this is typically considered as an *ad hoc* approach to reduce the feature space, without any justification of why the terms that are removed are noisy. Furthermore, this comes to conflict with a fundamental assumption that is widely used in IR; terms with low DF are rather informative, and therefore should be utilized in the learning process.

Contrary to those methods, the proposed core decomposition-based one is purely unsupervised. The maximum value k of the k-core decomposition is not an input of the algorithm, but depends on the properties of the graph. As we will present shortly in the experimental results, this process can reduce the feature space from 3% to almost 30% – depending on the window size and the properties of the graphs – without sacrificing the accuracy. Concluding, the proposed term selection technique is able reduce the feature space in an unsupervised manner with cost linear to the size of the document; this cost is comparable or even smaller to other text preprocessing or more complicated feature selection techniques.

6.4.5 *Computational Complexity*

As described earlier, some formulations of the TW-based schemes consider centrality criteria that are computationally intensive (e.g., closeness centrality). Nevertheless, in the case of TW and TW-IDF, those criteria are applied on a per document basis, where the corresponding graphs are sparse and of very small size, and thus are not prohibitive (see also Table 6.1 for the properties of the graphs). As we have observed from the experiments, the density of the graph – and therefore the running time – increases with the size of the window w; however, for very small window sizes ($w = \{2, 3\}$), we can achieve a very good trade-off between accuracy and execution time. We stress out here that the local and computationally efficient degree centrality criterion – with complexity on same order as the one of term frequency – performs quite well in most of the cases. In any case, even the most intensive criteria considered here (such as the closeness centrality), can be efficiently approximated quite well [EW04].

For the collection level graph $\mathcal{G}$ and the ICW-based schemes, the basic point is that the weighting of each term will be computed *only once* during the training phase of the model; thus, in the testing phase, heavy computations are not performed. In a similar way as in the document-based (local) graphs, the overall execution time can be improved, relying on approximation techniques of the corresponding measures. Lastly, by applying the easy-to-compute feature selection method presented earlier, the feature space of the problem is reduced, improving the training time of the classifier.

6.4.6 *Classification Algorithms*

Part of the TC pipeline, is the choice of the classification model [Bis06]. Since the goal of this work is to introduce new term weighting schemes, we rely on widely used classification algorithms. Specifically, in our experimental evaluation, we have used linear SVMs, due to their superior performance in TC [Joa98]. Furthermore, as discussed in [LK02], the choice of the kernel function of SVM is not very crucial, compared to the significance of the term weighting schemes.

6.5 EXPERIMENTAL EVALUATION

In this section, we present the experimental evaluation of the proposed TC framework.

6.5.1 *Description of the Datasets*

We have evaluated our method on four freely available standard TC datasets: three datasets for multi-class document categorization[2] (news articles: 20NG and REUTERS; web pages: WEBKB) and one for sentiment analysis[3] (IMDB):

2 The datasets can be found in: http://web.ist.utl.pt/acardoso/datasets/.
3 The dataset can be found in: http://ai.stanford.edu/~amaas/data/sentiment/.

- 20NG: the 20 Newsgroups dataset constitutes a collection of 18,821 newsgroup documents belonging to 20 categories. The dataset is split into 11,293 documents for training and 7,528 for test.

- REUTERS: this dataset is formed by 8 categories of Reuters-21578 dataset. It constitutes a collection of news articles that appeared on the Reuters newswire in 1987. The dataset is split into 5,485 documents for training and 2,189 for testing.

- WEBKB: the dataset is composed by the 4 most frequent categories of webpages from Computer Science departments. It is split into 2,803 training documents and 1,396 for test.

- IMDB: it constitutes a movie review dataset [Maa+11] from IMDB, where each document contains a positive or negative movie review. The dataset is split into 25,000 reviews for training and 25,000 for testing.

6.5.2 *Experimental Set-up and Baselines*

We have implemented the proposed TC framework in Python. We have used the `NetworX` library[4] for the graph-based representation of documents, the implementation of the term weighting schemes and the term selection method, and the `scikit-learn` machine learning library[5] for text preprocessing and classification. In the experiments, we have used linear SVMs.

We compare the proposed weighting schemes to several baseline methods as follows:

(i) The TW weighting scheme to (a) the term frequency with binary n-gram features (denoted by TF binary) and (b) the traditional TF weights (denoted by TF).

(ii) The TW-IDF scheme to the well-known TF-IDF of Eq. (6.2).

4 `NetworkX` Python software package: `https://networkx.github.io/`.
5 `scikit-learn` Python software package: `http://scikit-learn.org/`.

(iii) The TW-ICW and TF-ICW schemes to the term *frequency - inverse collection frequency* method (denoted by TF-ICF), where the penalization factor (ICF) counts the total frequency of appearance of a term.

We consider that comparing models of similar degree of complexity is more meaningful. Given the fact that the TF and TF-IDF baselines perform well in general, we are also interested to have a more broad comparison of the performances, examining the best scheme for each dataset. We compare the different approaches using the macro-average F1 score and the classification accuracy on the test sets; that way, we deal with the skewed class size distribution of some datasets [Sebo2]. For the notation of the proposed schemes, we use TW (centrality measure) (e.g., TW (degree)) to indicate the centrality and TW-ICW (centrality at G, centrality at $\mathcal{G}$) (e.g., TW-ICW (degree, degree)) for the local and collection-level graphs respectively.

6.5.3 *Experimental Results*

As we have already discussed, the size of the window w considered to create the graphs is one of the parameters of the model. From the extensive experimental evaluation that we have performed, we have concluded that window sizes w equal to 2 and 3 give the most persistent results across various datasets and weighting schemes. Table 6.1 shows the basic properties of the graphs that correspond to the documents of each collection. As we can observe, the average number of nodes and edges of the document level graphs G_d for both window sizes, is quite small. For the collection level graph $\mathcal{G}$, the number of nodes $|V_{\mathcal{G}}|$ corresponds to the number of unique terms in the collection. Additionally, as expected, the number of edges $|E_{\mathcal{G}}|$ almost doubles from $w = 2$ to $w = 3$.

Table 6.2 presents the results concerning the categorization performance of the proposed schemes for the four datasets. For completeness in the presentation, we report results for both cases of window size. Also, since for the frequency based baseline methods (TF, TF binary, TF-IDF and TF-ICF) there is no notion of window size, the results for $w = 2$ and $w = 3$ are the same (we simply fill all cells of the table). We have also examined

Dataset	Document level graphs G_d			Collection level graph $\mathcal{G}$		
	$\text{avg}(\lvert V_d \rvert)$	$\text{avg}(\lvert E_d \rvert)$		$\lvert V_{\mathcal{G}} \rvert$	$\lvert E_{\mathcal{G}} \rvert$	
		$w = 2$	$w = 3$		$w = 2$	$w = 3$
20NG	101.14	142.29	159.65	62,436	771,517	1,449,601
REUTERS	40.60	56.51	109.76	14,575	144,202	269,998
WEBKB	75.55	110.48	216.05	7,287	177,160	329,288
IMDB	96.69	119.13	233.87	50,983	1,377,742	2,521,640

Table 6.1: Properties of the graphs that correspond to the documents of each collection. For each dataset and for window size $w = \{2, 3\}$, we report: (i) the average number of nodes $\text{avg}(\lvert V_d \rvert)$ and edges $\text{avg}(\lvert E_d \rvert)$ over all document level graphs $G_d = (V_d, E_d), \forall d \in \mathcal{D}$; (ii) the number of nodes $\lvert V_{\mathcal{G}} \rvert$ and edges $\lvert E_{\mathcal{G}} \rvert$ of the collection level graph $\mathcal{G} = (V_{\mathcal{G}}, E_{\mathcal{G}})$.

several centrality criteria (from undirected and directed graphs) and the best performance among them was achieved by the degree, closeness and eigenvector centrality. More precisely, comparing the baseline TF to the graph-based ones, namely TW (degree) and TW (closeness), in almost all cases TW gives higher F1 and accuracy results. For example, in the case of the IMDB dataset, the TW (closeness) weighting scheme achieves 4.4% and 4.5% increase over TF on the accuracy and F1 score respectively.

Similar observations can be made in the case where IDF penalization is applied. In most of the datasets, the TW-IDF (degree) scheme performs quite well; in fact, in the 20NG and REUTERS dataset, consistently has the best reported performance compared to all the other weighting mechanisms. The interesting point here, which is confirmed by the related literature [Lan+], is that TF-IDF is in general inferior of TF in TC. However, when the IDF penalization factor is applied on the TW term-to-document weighting, a powerful mechanism is derived.

In the case of purely graph-based schemes (e.g., TW-ICW), we can observe that some of them produce very good classification results. In almost all cases, the two models considered here, namely TW-ICW (degree, degree) and TW-ICW (degree, eigenvector) are superior of their frequency-based equivalent (TF-ICF). Furthermore, in most of the datasets, they also outperform TW,

Weighting Schemes	Reuters				WebKB			
	$w = 2$		$w = 3$		$w = 2$		$w = 3$	
	F1	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**
TF	0.9139	0.9666	0.9139	0.9666	0.8837	0.8954	0.8837	0.8954
TF binary	0.8691	0.9566	0.8691	0.9566	0.8686	0.8818	0.8686	0.8818
TW (degree)	0.9130	0.9671	**0.9409**	**0.9707**	0.8647	0.8839	**0.8906**	**0.8989**
TW (closeness)	0.8979	0.9602	0.9105	0.9634	0.8935	0.9025	0.8876	0.8982
TF-IDF	0.8950	0.9639	0.8950	0.9639	0.8665	0.8789	0.8665	0.8789
TW-IDF (degree)	**0.9410**	**0.9748**	0.9396	0.9716	0.8480	0.8753	0.8789	0.8875
TW-IDF (closeness)	0.8930	0.9597	0.8940	0.9584	0.8871	0.8946	0.8729	0.8832
TF-ICF	0.8848	0.9570	0.8848	0.9570	0.8661	0.8825	0.8661	0.8825
TW-ICW (degree, degree)	0.9335	0.9721	0.9194	0.9661	**0.8976**	**0.9040**	0.8684	0.8839
TW-ICW (degree, eigenvector)	0.9313	0.9712	0.9201	0.9661	0.8916	0.8982	0.8633	0.8782
TF-ICW (degree)	0.8797	0.9479	0.8797	0.9479	0.8651	0.8775	0.8681	0.8796

(a) Reuters and WebKB

Weighting Schemes	20ng				Imdb			
	$w = 2$		$w = 3$		$w = 2$		$w = 3$	
	F1	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**
TF	0.7985	0.8040	0.7985	0.8040	0.8361	0.8376	0.8361	0.8376
TF binary	0.8173	0.8226	0.8173	0.8226	**0.8822**	**0.8813**	**0.8822**	**0.8813**
TW (degree)	0.7660	0.7827	0.7954	0.8078	0.8620	0.8605	0.8728	0.8726
TW (closeness)	0.8090	0.8171	0.7984	0.8065	0.8740	0.8744	0.8645	0.8649
TF-IDF	0.8222	0.8275	0.8222	0.8275	0.8322	0.8344	0.8322	0.8344
TW-IDF (degree)	0.8261	0.8377	**0.8362**	**0.8454**	0.8774	0.8780	0.8661	0.8680
TW-IDF (closeness)	0.8241	0.8321	0.8257	0.8333	0.8480	0.8495	0.8367	0.8389
TF-ICF	0.8328	0.8291	0.8328	0.8291	0.8318	0.8340	0.8318	0.8340
TW-ICW (degree, degree)	0.8291	0.8374	0.8231	0.8301	0.8745	0.8755	0.8539	0.8566
TW-ICW (degree, eigenvector)	**0.8354**	**0.8426**	0.8203	0.8279	0.8741	0.8757	0.8552	0.8583
TF-ICW (degree)	0.7862	0.7898	0.7856	0.7892	0.8264	0.8292	0.8262	0.8290

(b) 20ng and Imdb

Table 6.2: Macro-average F1 score and accuracy on the (a) Reuters, WebKB datasets and (b) 20ng, Imdb, for window size $w = \{2, 3\}$. Bold font indicates the best performance on each column and blue the best overall performance on each dataset.

having results very close to the ones of TW-IDF. Here, we consider two variants of this model. In the TW-ICW (degree, degree) a local centrality criterion is used both at the document level and at the collection level; in the TW-ICW (degree, eigenvector), we set a local criterion at the document level and a global one for the collection graph. As the results indicate, the first choice has led to the best reported result for the WEBKB dataset. We also note here that, the TF-ICW scheme gave results inferior from the rest models.

From the above results, it is evident that the graph-based criteria can further improve the classification task. We consider that the discriminative nature of the features is derived by the underlying graph and by the fact that we treat the term weighting process as a ranking task in the interconnected feature space defined by the graph. That way, we are able to augment the unigram feature space of the learning task with weights that implicitly consider information of n-grams (short and long ones) in the document – as expressed by paths in the graph – without increasing the dimensionality of the problem. In other words, the feature space is the one defined by the unigrams of our collection, but the weights capture information beyond them.

To see this effect more clearly, we have examined the TF n-gram binary scheme (TF binary), i.e., all the possible n-grams of the collection with binary weights (up to 6-grams in our experiments). This has the effect that the feature space of the problem explodes, having the following number of features: 20NG: 6,225,130 , REUTERS: 1,304,720 , WEBKB: 1,508,622 and IMDB: 13,401,619 – with direct implication on the efficiency of the method. For comparison reasons, the size of the unigram feature space considered by our framework is equal to the unique terms in the collections and much smaller compared to the $n-$grams one (also shown in Table 6.1): 20NG: 62,436 , REUTERS: 14,575 , WEBKB: 7,287 and IMDB: 50,983. Moreover, as we can see from Table 6.2, in most of the cases the graph-based weighting mechanisms are able to outperform TF binary (except from the IMDB dataset; however, in that case the feature space of TF binary is orders of magnitude larger).

Category	Precision		Recall		F1 score	
	TF	TW	TF	TW	TF	TW
project	0.8250	0.8701	0.7857	0.7976	0.8049	0.8323
course	0.9511	0.9668	0.9419	0.9387	0.9465	0.9525
faculty	0.8770	0.8908	0.8770	0.8503	0.8770	0.8700
student	0.8973	0.8767	0.9154	0.9412	0.9063	0.9078

Table 6.3: Comparison of TF vs. TW (degree), per category of the WEBKB dataset (window size $w = 3$).

Category	Precision		Recall		F1 score	
	TF-IDF	TW-IDF	TF-IDF	TW-IDF	TF-IDF	TW-IDF
project	0.8089	0.8767	0.7560	0.7619	0.7815	0.8153
course	0.9316	0.9519	0.9226	0.9581	0.9271	0.9550
faculty	0.8740	0.8564	0.8717	0.8449	0.8728	0.8506
student	0.8730	0.8787	0.8971	0.9191	0.8849	0.8985

Table 6.4: Comparison of TF-IDF vs. TW-IDF (degree), per category of the WEBKB dataset (window size $w = 3$).

We have also examined how TW (degree) and TW-IDF (degree) behave on each category of the WEBKB dataset, compared to TF and TF-IDF respectively. Note here that, the size of the categories is skewed, with "project" and "course" being the smallest ones. As we can observe from Tables 6.3 and 6.4, the macro-average precision, recall and F1 score for those categories is higher compared to TF and TF-IDF. This is an indication that the proposed TW-based weights perform relatively better – compared to frequency-based – even when few training examples are available.

To further understand the discriminative power of the graph-based weighting schemes compared to the frequency-based ones, as well as deviations among them, we examine correlations between the ranking of nodes achieved from the graph-based representation vs. the one produced by term frequency. More precisely, for each document of a dataset, we compute the Pearson

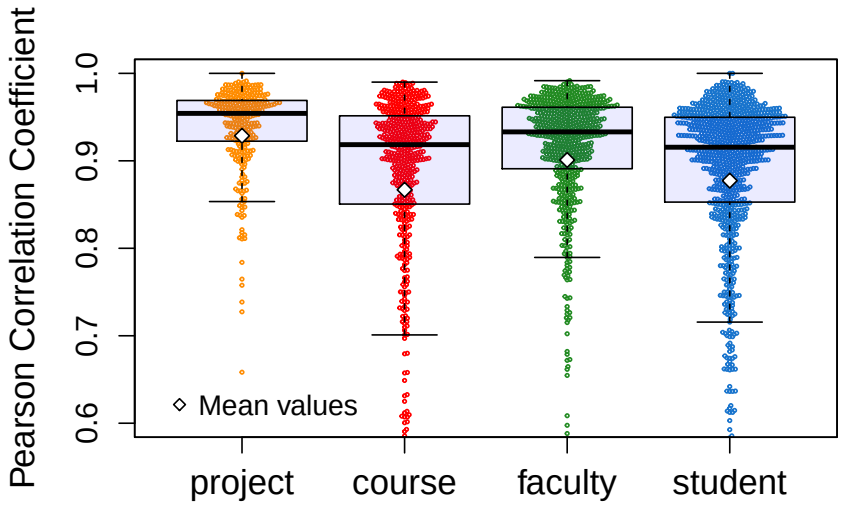


Figure 6.3: Correlation of the raw term centrality weights (degree) and frequencies per document, for each category of the WebKB dataset. Each point of the plot corresponds to the Pearson correlation coefficient of the above measures for a document.

correlation coefficient between the ranking achieved by the raw degree centrality and the raw term frequency. We perform this task per category of the dataset. Figure 6.3 presents the box plot of the results for each of the four categories of WebKB (each point corresponds to a document). Since the classification results for both methods are close to each other, we expect to have high correlation between the corresponding rankings. As we can observe, although the overall correlation is high, there exist many documents where the ranking produced by the methods deviates (low value of correlation coefficient). This is more evident in the "course" and "student" categories. We consider that such deviations in the rankings are able to cause alterations in the classification results.

Furthermore, we have examined deviations in the rankings produced by different weighting criteria. We consider the top-20 terms obtained per document, after ranking them with respect to TF and TW (degree). Then, we compute the Kendall rank correlation coefficient (or Kendall τ), in order to quantify the rank correlation between the top terms. Figure 6.4 presents

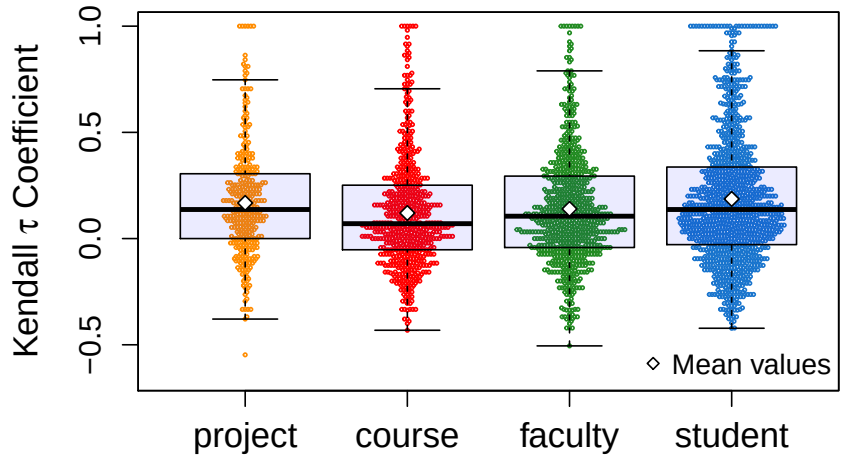
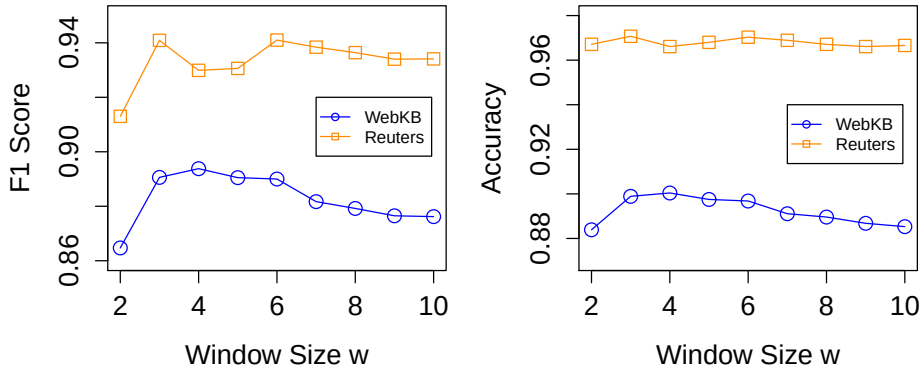


Figure 6.4: Kendall τ rank correlation coefficient of the top-20 terms ranked by TF and TW (degree) per document, for each category of the WEBKB dataset. Each point of the plot corresponds to the Kendall τ coefficient of the above measures for a document.

the box plot of the results for each category of the WEBKB dataset. As we can observe, the Kendall τ coefficient is slightly above zero, indicating disagreement in the top ranked terms per method. Even though we do not have ground-truth information about the discriminative nature of terms, these results indicate differences on the ranking of the features between graph-based and frequency-based schemes.

EFFECT OF WINDOW SIZE. Lastly, Fig. 6.5 depicts the F1 score and classification accuracy on the WEBKB and REUTERS datasets of the TW (degree) scheme for various windows sizes $w = \{2, 3, \ldots, 10\}$. As we can observe, the performance is rather stable, with the maximum F1 score and accuracy achieved close to $w = 3$. This is mainly the reason for setting $w = \{2, 3\}$ in all the experiments that we have performed. In any case, even if much larger values of w were able to get slightly better results, a smaller window size would be preferable, due to the overall overhead that could be introduced (mainly because of the increase on the density of the graph).



(a) F1 score (b) Accuracy

Figure 6.5: Classification performance vs. window size w. (a) F1 score and (b) classification accuracy of the TW (degree) scheme on the WebKB and Reuters datasets, for window size $w = \{2, \ldots, 10\}$.

6.5.4 *Graph-Based Feature Selection*

We have also examined the performance of the proposed unsupervised feature selection method that relies on the k-core decomposition of each graph. Table 6.5 shows the performance results for three of our schemes applied on all datasets (the rest weighting schemes show similar behavior). For completeness, we have examined two different window sizes ($w = \{2, 3\}$). In all cases, we report the F1 score and classification accuracy before and after the feature selection process, along with the percentage of feature reduction.

As we can observe, in most of the datasets we can achieve a quite significant amount of feature reduction that ranges from 3% to 33%. The smallest reduction is done on the WebKB dataset for both window sizes; as we described in Section 6.4.4, the amount of features that are kept by the k-core decomposition method depends both on the window size as well as on the properties of the graph. In the rest three datasets, the feature reduction is quite intense and as expected, increases with respect to w. We could even use higher values of w to achieve greater reduction; however, in this case the graphs become more dense increasing the overall complexity of the framework. In any case, the most important point here is that using a simple to compute but effective unsupervised mechanism, the classification

Weighting Schemes	Reuters				WebKB			
	$w = 2$ (23%)		$w = 3$ (28%)		$w = 2$ (3%)		$w = 3$ (6%)	
	F1	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**
TW (degree)	0.9130	0.9671	0.9409	0.9707	0.8647	0.8839	0.8906	0.8989
TW (degree) **core**	0.9144	0.9675	0.9409	0.9707	0.8600	0.8810	0.8913	0.9004
TW-IDF (degree)	0.9410	0.9748	0.9396	0.9716	0.8480	0.8753	0.8789	0.8875
TW-IDF (degree) **core**	0.9407	0.9739	0.9357	0.9716	0.8499	0.8746	0.8777	0.8875
TW-ICW (degree, degree)	0.9335	0.9721	0.9194	0.9661	0.8976	0.9040	0.8684	0.8839
TW-ICW (degree, degree) **core**	0.9308	0.9716	0.9193	0.9661	0.8972	0.9047	0.8695	0.8832

(a) Reuters and WebKB

Weighting Schemes	20ng				Imdb			
	$w = 2$ (26%)		$w = 3$ (33%)		$w = 2$ (6%)		$w = 3$ (17%)	
	F1	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**
TW (degree)	0.7660	0.7827	0.7954	0.8078	0.8620	0.8605	0.8728	0.8726
TW (degree) **core**	0.7640	0.7808	0.7907	0.8037	0.8620	0.8604	0.8730	0.8728
TW-IDF (degree)	0.8261	0.8377	0.8362	0.8454	0.8774	0.8780	0.8661	0.8680
TW-IDF (degree) **core**	0.8249	0.8366	0.8360	0.8453	0.8751	0.8755	0.8666	0.8648
TW-ICW (degree, degree)	0.8291	0.8374	0.8231	0.8301	0.8745	0.8755	0.8539	0.8566
TW-ICW (degree, degree) **core**	0.8273	0.8360	0.8201	0.8275	0.8744	0.8755	0.8544	0.8569

(b) 20ng and Imdb

Table 6.5: Classification performance (F1 and Accuracy) of the proposed TW (degree), TW-IDF (degree) and TW-ICW (degree, degree) schemes before and after (denoted by **core**) unsupervised feature selection for the (a) Reuters, WebKB datasets and (b) 20ng, Imdb. Close to each window size, we provide the feature reduction that was achieved.

performance on the reduced feature space is almost the same – and in some cases slightly better – compared to the original one.

6.6 CONCLUSIONS AND DISCUSSION

In this Chapter, we proposed a graph-based framework for text categorization. By treating the term weighting task as a node ranking problem of interconnected features defined by a graph, we were able to determine the term-to-document importance using node centrality criteria. Building on this

formulation, we proposed novel weighting schemes at the collection level, in order to penalize globally important terms (as analogous to "globally frequent terms"). We also proposed an unsupervised feature selection approach at the graph level, through a trimming process of the less important features based on the k-core decomposition.

We consider that the ICW scheme and the proposed feature selection method may also have applications in information retrieval. In fact, graph-based term weighting has already been applied there, so it would be interesting to also study the performance of the proposed penalization mechanism. Furthermore, another future direction could be to examine *supervised* graph-based feature selection techniques; that way, we can create graphs per category and then perform reduction of terms in those graphs.

7

CONCLUDING REMARKS

Graph data is ubiquitous, posing a wealth of fascinating and challenging problems. The goal of this dissertation was to develop methods for mining social and information networks. In particular, we built upon the concept of core decomposition, proposing efficient degeneracy-based graph mining methods in order to:

(i) Design models for studying and analyzing the dynamics of real-world social networks, towards extracting useful knowledge for real problems and applications.

(ii) Develop algorithmic tools with applications to large scale graph and text data analytics.

Both points constitute two interrelated aspects of data and network science. In the following sections, we provide an overview of the main contributions of the thesis and discuss future research directions.

7.1 SUMMARY OF CONTRIBUTIONS

The main contributions of the thesis can be summarized as follows.

SOCIAL ENGAGEMENT DYNAMICS. In Chapter 3 we studied the engagement dynamics of social networks, proposing node level and graph level models of engagement based on the properties of k-core decomposition. Through extensive experimental evaluation, we showed that the engagement properties present interesting connections to other graph characteristics, as well as a clear deviation from the corresponding behavior of random graphs. The proposed models can further help us to better understand the structural dynamics of social networks, with direct implications on how to build more stable and robust social networking systems.

VULNERABILITY ASSESSMENT IN SOCIAL NETWORKS. In Chapter 4 we introduced a novel concept of vulnerability assessment in social networks, motivated by the fact that current users may depart from the network or stop being active on it (similar to the aspect of churn in the business domain). In particular, we proposed a novel degeneracy-based model to capture the cascading effect on the network, where the departure of a user may affect the engagement level of his neighbors in the graph, leading to a disengagement epidemic. Based on the proposed model, we were able to study the vulnerability of real social networks under cascades triggered by the departure of nodes based on their engagement level. Our experimental results indicated that social networks are robust under cascades triggered by randomly selected nodes but highly vulnerable in cascades caused by targeted departures of nodes with high engagement level. Those observations complementing seminal results in network science about the robustness of real networks.

IDENTIFICATION OF INFLUENTIAL SPREADERS. In Chapter 5 we posed the question how to locate influential nodes that can efficiently spread information in complex networks – a problem of particular importance in many application domains, such as epidemiology and viral marketing. We showed that the K-truss decomposition, a triangle-based extension of the k-core decomposition, is an algorithmic tool that is able to identify highly influential nodes compared to previously used importance criteria. The detected nodes show better spreading behavior, leading to faster and wider diffusion on the network; they also dominate the small set of nodes that achieve the optimal spreading in the network.

GRAPH-BASED TEXT CATEGORIZATION. In Chapter 6 we studied how graph mining can be used to enhance large-scale text analytics and in particular the text categorization task which is central in a plethora of data science applications. We built upon the fact that graphs can be used to represent textual content; the nodes correspond to terms and the edges capture term co-occurrence relationships – addressing the term-independence assumption widely used in many text analytics tasks. That way, we proposed a graph-

based framework for text categorization, where the term weighting problem is treated as a node ranking task in the feature space defined by the graph, applying simple node centrality criteria. We also proposed an unsupervised graph-based feature (i.e., term) selection approach, based on the properties of the k-core decomposition. We showed that graph-based weighting mechanisms produce more discriminative feature weights for text categorization, outperforming existing frequency-based criteria. Additionally, using the degeneracy-based term selection technique, we are able to reduce the feature space of the classifier without sacrificing its accuracy.

7.2 FUTURE DIRECTIONS

In this section, we discuss future research directions for the topics covered in the thesis as well as more broad topics of interest in the areas of graph mining and network science (we have also discussed future directions for each topic at the end of each chapter).

ENGAGEMENT DYNAMICS ON GRAPHS WITH RICH SEMANTICS. In the model presented in Chapter 3, we considered the graphs as undirected. However, real-world networks convey rich semantics and interaction patterns; thus, they are modeled by more complex graph types beyond the simple undirected one. For example, a plethora of real-world social networks are by their nature directed (e.g., twitter's who-follows-whom network). Furthermore, many social networks represent trust relationships between individuals and are modeled by signed networks. Therefore, it is of practical importance to define models and metrics of engagement in the aforementioned cases. Additionally, the property of community structure [For10; MV13a] should also be taken into account while modeling the engagement dynamics; intuitively, network-effects are more strong between individuals (i.e., nodes) of the same community.

PREDICTION ALGORITHMS FOR NETWORK VULNERABILITY. A potential practical application that could make use of the disengagement model described in Chapter 4, is the prediction of the effect that the departure

of a user can have on the vulnerability of the network. As we discussed in Section 4.5 of Chapter 4, in order to examine the predictive capabilities of the proposed model, we need to further validate it on networks with ground-truth departure or inactivity information. Nevertheless, designing forecasting tools with respect to network vulnerability is an interesting future research direction.

IDENTIFICATION OF MULTIPLE INFLUENTIAL SPREADERS. Our K-truss decomposition-based method presented in Chapter 5, identifies nodes that act as single influential spreaders. As we discussed there, by simply choosing more than one nodes from the maximal k-truss subgraph is not a good approach to deal with the multiple influential spreaders problem due to the high overlap on the neighborhoods of those nodes (similar behavior occurs for the case of k-core decomposition; nodes with the maximum k-core number, typically belong to the same subgraph). How to utilize the good spreading properties of the nodes belonging to the maximal core subgraphs (K-truss or k-core) in order to detect a set of N nodes that can maximize the spread of influence [KKT03], is still an open topic and interesting research direction. One possible approach is to combine the community structure property [For10; MV13a] observed in real networks with the core decomposition, in the following way: (i) extract the N largest communities applying a community detection algorithm (e.g., [Blo+08]); (ii) find the most influential nodes per community by the truss or core decomposition methods. Since communities typical act as bottlenecks for information diffusion, it is interesting to examine what is the total spreading achieved by the "best spreaders" of each community.

GRAPH-BASED TEXT ANALYTICS. As we discussed in Chapter 6, graphs have already been used in several problems in text analytics. We consider that more tasks from the text mining and information retrieval domains can also utilize the strong modeling capabilities of graphs as well as the mature learning and mining methods on graphs. In Section 6.6 of Chapter 6, we described such research directions with respect to the text categorization task. Another interesting problem is how to extract informative keywords

from documents that cover *multiple topics* in an unsupervised manner, since the problem has been addressed only for the single topic case.

TOPICS IN GRAPH MINING ALGORITHM DESIGN. Design of graph mining algorithms with respect to the following two points:

(i) *Scalability issues.* Scalability is always a concern while developing graph mining algorithms for large scale data. In the recent literature, a big research effort has been devoted to design or extend already existing algorithms to frameworks such as MAPREDUCE [DG04], SPARK [Zah+10] and GRAPHLAB [Low+10]. Other scalable graph mining approaches rely on sampling [LF06; MBW10] or sparsification techniques [SPR11]. Here, we emphasize on algorithms that carefully process the graph towards improving the total running time. Such an example is the LOUVAIN community detection algorithm [Blo+08], where modularity is optimized in an hierarchical manner in the graph, first locally for small size communities and then continuing by aggregating already formed communities.

At the same spirit moves our approach, called CORECLUSTER, for speeding up community detection algorithms [Gia+14a]. The main idea was to combine a computational intensive graph clustering algorithm (such as spectral clustering [Lux07]) with an easy to compute, clustering-preserving hierarchical representation of the graph – as produced by the k-core decomposition – leading to a scalable graph clustering tool. We consider that similar methodological approaches can also be applied to other graph mining tasks – with the k-core decomposition playing a central role due to its efficiency.

(ii) *Resilience to network uncertainty.* A particularly important point, which rarely taken into account while developing and evaluating graph mining algorithms, is their resilience (or stability) to network uncertainty or perturbation [AV13]. In many cases, the input graph data can be incomplete or noisy (e.g., due to noise introduced during the collection of the data or for privacy preserving reasons). Then, the following question arises: how stable are the results produced by an algorithms with

respect to the uncertainty (i.e., noise level) of the input data? It would be interesting to examine the resilience of graph mining algorithms (e.g., community detection), as an orthogonal performance evaluation criterion in addition to efficiency and effectiveness.

BIBLIOGRAPHY

[AV13] Abhijin Adiga and Anil Kumar S. Vullikanti. How Robust Is the Core of a Network? In *PKDD '13: Machine Learning and Knowledge Discovery in Databases*, 2013, pp. 541–556 (cit. on p. 137).

[AC05] Manu Aery and Sharma Chakravarthy. InfoSift: Adapting Graph Mining Techniques for Text Classification. In *FLAIRS '05: Proceedings of the Eighteenth International Florida Artificial Intelligence Research Society Conference, Clearwater Beach, Florida*, 2005, pp. 277–282 (cit. on p. 107).

[AJB00] R. Albert, H. Jeong, and A.L. Barabási. Error and attack tolerance of complex networks. *Nature* 406:6794 (2000) (cit. on pp. 51, 54, 56, 59, 62, 71).

[AB02] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.* 74 (1 2002), pp. 47–97 (cit. on pp. 44, 46, 51, 54, 56, 59, 62, 71).

[AJB99] Reka Albert, Hawoong Jeong, and Albert-Laszlo Barabasi. The diameter of the World Wide Web. *Nature* 401 (1999), pp. 130–131 (cit. on p. 1).

[AhBV06] J. Ignacio Alvarez-hamelin, Alain Barrat, and Alessandro Vespignani. Large scale networks fingerprinting and visualization using the k-core decomposition. In *NIPS '06: Advances in Neural Information Processing Systems*, 2006, pp. 41–50 (cit. on p. 17).

[AH+08] José Ignacio Alvarez-Hamelin, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. *k*-core Decomposition of Internet Graphs: Hierarchies, Self-similarity and Measurement Biases. *NHM* 3:2 (2008), p. 371 (cit. on pp. 17, 26, 32, 45, 62).

[AKM08] Aris Anagnostopoulos, Ravi Kumar, and Mohammad Mahdian. Influence and correlation in social networks. In *KDD '08: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 7–15 (cit. on p. 26).

[AC09] Reid Andersen and Kumar Chellapilla. Finding Dense Subgraphs with Size Bounds. In *WAW '09: Algorithms and Models for the Web-Graph*, 2009, pp. 25–37 (cit. on p. 17).

[And+00] I. Androutsopoulos, J. Koutsias, K. Chandrinos, G. Paliouras, and C. Spyropoulos. An Evaluation of Naive Bayesian Anti-Spam Filtering. In *MLNIA '00: Proceedings of the Workshop on Machine Learning in the New Information Age*, 2000 (cit. on p. 104).

[Aro+10] Shilpa Arora, Elijah Mayfield, Carolyn Penstein-Rosé, and Eric Nyberg. Sentiment Classification Using Automatically Extracted Subgraph Features. In *CAAGET '10: Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, 2010, pp. 131–139 (cit. on p. 107).

[Bac+06] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. Group Formation in Large Social Networks: Membership, Growth, and Evolution. In *KDD '06: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 44–54 (cit. on pp. 24, 26, 46, 53, 56).

[BK14] Joonhyun Bae and Sangwook Kim. Identifying and ranking influential spreaders in complex networks by neighborhood coreness. *Physica A: Statistical Mechanics and its Applications* 395 (2014), pp. 549–559 (cit. on p. 79).

[BYL12] R. Baeza-Yates and M. Lalmas. User Engagement:The Network Effect Matters! In *CIKM '12: Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, 2012, pp. 1–2 (cit. on p. 24).

[BYRN99] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1999 (cit. on pp. 104, 106, 109).

[BA99] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science* 286:5439 (1999), pp. 509–512 (cit. on p. 1).

[BBV08] Alain Barrat, Marc Barthlemy, and Alessandro Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, 2008 (cit. on pp. 53, 56, 68, 76, 78, 84, 85).

[BKT13] Pavlos Basaras, Dimitrios Katsaros, and Leandros Tassiulas. Detecting Influential Spreaders in Complex, Dynamic Networks. *Computer* 46:4 (2013), pp. 24–29 (cit. on p. 79).

[BZ03] Vladimir Batagelj and Matjaz Zaversnik. An $\mathcal{O}(m)$ Algorithm for Cores Decomposition of Networks. *CoRR* (2003) (cit. on pp. 13, 27, 71, 112).

[BZ02] Vladimir Batagelj and Matjaz Zaversnik. Generalized Cores. *CoRR* (2002) (cit. on pp. 15, 16).

[Bec+10] Luca Becchetti, Paolo Boldi, Carlos Castillo, and Aristides Gionis. Efficient Algorithms for Large-scale Local Triangle Counting. *ACM Trans. Knowl. Discov. Data* 4:3 (2010), pp. 1–28 (cit. on p. 74).

[Bha+11] Kshipra Bhawalkar, Jon Kleinberg, Kevin Lewi, Tim Roughgarden, and Aneesh Sharma. Preventing Unraveling in Social Networks: the Anchored k-core Problem. In *ICALP '11: Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming*, 2011, pp. 440–451 (cit. on pp. 25, 26, 30, 54–56).

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., 2006 (cit. on pp. 109, 121).

[BL12] Roi Blanco and Christina Lioma. Graph-based Term Weighting for Information Retrieval. *Inf. Retr.* 15:1 (2012), pp. 54–92 (cit. on pp. 108, 114, 116).

[BL07] Roi Blanco and Christina Lioma. Random Walk Term Weighting for Information Retrieval. In *SIGIR '07: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007, pp. 829–830 (cit. on p. 108).

[Blo+08] V.D. Blondel, J.L. Guillaume, R. Lambiotte, and E.L.J.S. Mech. Fast unfolding of communities in large networks. *J. Stat. Mech* (2008), P10008 (cit. on pp. 136, 137).

[Boc+06] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports* 424:4-5 (2006), pp. 175–308 (cit. on p. 7).

[BnPS02] Marián Boguñá and Romualdo Pastor-Satorras. Epidemic spreading in correlated complex networks. *Phys. Rev. E* 66 (4 2002), p. 047104 (cit. on p. 84).

[Bon+14] Francesco Bonchi, Francesco Gullo, Andreas Kaltenbrunner, and Yana Volkovich. Core Decomposition of Uncertain Graphs. In *KDD '14: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 1316–1325 (cit. on p. 15).

[BHRM12] Javier Borge-Holthoefer, Alejandro Rivero, and Yamir Moreno. Locating privileged spreaders on an online social network. *Phys. Rev. E* 85 (6 2012), p. 066123 (cit. on p. 79).

[BHTM11] F. Boudin, S. Huet, and J.M. Torres-Moreno. A Graph-based Approach to Cross-language Multi-document Summarization. *Polibits* 43 (2011), pp. 113–118 (cit. on p. 108).

[Bou13] Florian Boudin. A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction. In *IJCNLP '13: Proceedings of the Sixth International Joint Conference on Natural Language Processing*, 2013, pp. 834–838 (cit. on pp. 108, 119).

[BBD13] Adrien Bougouin, Florian Boudin, and Béatrice Daille. TopicRank: Graph-Based Topic Ranking for Keyphrase Extraction. In *IJCNLP '13: Proceedings of the Sixth International Joint Confer-*

ence on Natural Language Processing, 2013, pp. 543–551 (cit. on p. 108).

[BP98] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer networks and ISDN systems* 30:1 (1998), pp. 107–117 (cit. on p. 78).

[BH12] Andries E. Brouwer and Willem H. Haemers. *Spectra of Graphs*. New York, NY, 2012 (cit. on p. 85).

[Cal+00] Duncan S. Callaway, M. E. J. Newman, Steven H. Strogatz, and Duncan J. Watts. Network Robustness and Fragility: Percolation on Random Graphs. *Phys. Rev. Lett.* 85 (25 2000), pp. 5468–5471 (cit. on pp. 54, 56).

[Car+07] Shai Carmi, Shlomo Havlin, Scott Kirkpatrick, Yuval Shavitt, and Eran Shir. A Model of Internet Topology using k-shell Decomposition. *PNAS* 104:27 (2007), pp. 11150–11154 (cit. on pp. 17, 26, 62).

[CRMV15] Jordi Casas-Roma, Fragkiskos D. Malliaros, and Michalis Vazirgiannis. k-Degree Anonymity on Directed Networks. *Manuscript* (2015) (cit. on p. xii).

[CPS10] Claudio Castellano and Romualdo Pastor-Satorras. Thresholds for Epidemic Spreading in Networks. *Phys. Rev. Lett.* 105 (21 2010), p. 218701 (cit. on p. 84).

[CF12] Deepayan Chakrabarti and Christos Faloutsos. *Graph Mining: Laws, Tools, and Case Studies*. Morgan & Claypool Publ., 2012 (cit. on pp. 1, 7, 56).

[Cha+08] Deepayan Chakrabarti, Yang Wang, Chenxi Wang, Jurij Leskovec, and Christos Faloutsos. Epidemic Thresholds in Real Networks. *ACM Trans. Inf. Syst. Secur.* 10:4 (2008), 1:1–1:26 (cit. on pp. 84, 85).

[Che+13a] Duan-Bing Chen, Hui Gao, Linyuan Lü, and Tao Zhou. Identifying Influential Nodes in Large-Scale Directed Networks: The Role of Clustering. *PLoS ONE* 8:10 (2013), e77455 (cit. on p. 79).

[Che+13b] Duan-Bing Chen, Rui Xiao, An Zeng, and Yi-Cheng Zhang. Path diversity improves the identification of influential spreaders. *EPL (Europhysics Letters)* 104:6 (2013), p. 68006 (cit. on p. 79).

[Che+12] Duanbing Chen, Linyuan Lü, Ming-Sheng Shang, Yi-Cheng Zhang, and Tao Zhou. Identifying influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications* 391:4 (2012), pp. 1777–1787 (cit. on p. 79).

[CCC14] Pei-Ling Chen, Chung-Kuang Chou, and Ming-Syan Chen. Distributed algorithms for k-truss decomposition. In *Big Data '14: IEEE International Conference on Big Data*, 2014, pp. 471–480 (cit. on p. 76).

[Che+11] James Cheng, Yiping Ke, Shumo Chu, and M. Tamer Ozsu. Efficient Core Decomposition in Massive Networks. In *ICDE '11: Proceedings of the 2011 IEEE 27th International Conference on Data Engineering*, 2011, pp. 51–62 (cit. on p. 14).

[Chu97] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997 (cit. on p. 85).

[CSN09] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. Power-Law Distributions in Empirical Data. *SIAM Rev.* 51:4 (2009), pp. 661–703 (cit. on pp. 62, 88).

[Coh09] Jonathan Cohen. Graph Twiddling in a MapReduce World. *Computing in Science and Engg.* 11:4 (2009), pp. 29–41 (cit. on p. 76).

[Coh08] Jonathan Cohen. Trusses: Cohesive subgraphs for social network analysis. *National Security Agency Technical Report* (2008) (cit. on pp. 16, 72, 74, 75, 81, 84).

[CH10] Reuven Cohen and Shlomo Havlin. *Complex Networks: Structure, Robustness and Function*. Cambridge University Press, 2010 (cit. on pp. 54, 56).

[Coh+01] Reuven Cohen, Keren Erez, Daniel ben Avraham, and Shlomo Havlin. Breakdown of the Internet under Intentional Attack. *Phys. Rev. Lett.* 86 (16 2001), pp. 3682–3685 (cit. on pp. 54, 56, 71).

[Cui+14] Wanyun Cui, Yanghua Xiao, Haixun Wang, and Wei Wang. Local Search of Communities in Large Graphs. In *SIGMOD '14: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 2014, pp. 991–1002 (cit. on p. 17).

[Das+08] Koustuv Dasgupta, Rahul Singh, Balaji Viswanathan, Dipanjan Chakraborty, Sougata Mukherjea, Amit A. Nanavati, and Anupam Joshi. Social Ties and Their Relevance to Churn in Mobile Telecom Networks. In *EDBT '08: Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology*, 2008, pp. 668–677 (cit. on p. 54).

[DG04] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *OSDI '04: Proceedings of the 6th Conference on Symposium on Operating Systems Design & Implementation*, 2004, pp. 137–150 (cit. on pp. 14, 137).

[DGT02] F. Denis, R. Gilleron, and M. Tommasi. Text classification from positive and unlabeled examples. In *IPMU '02: Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 2002, pp. 1927–1934 (cit. on p. 106).

[Des+05] Mukund Deshpande, Michihiro Kuramochi, Nikil Wale, and George Karypis. Frequent Substructure-Based Approaches for Classifying Chemical Compounds. *IEEE Trans. on Knowl. and Data Eng.* 17:8 (2005), pp. 1036–1050 (cit. on p. 107).

[Dhi01] Inderjit S. Dhillon. Co-clustering Documents and Words Using Bipartite Spectral Graph Partitioning. In *KDD '01: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001, pp. 269–274 (cit. on p. 107).

[DR01] Pedro Domingos and Matt Richardson. Mining the Network Value of Customers. In *KDD '01: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001, pp. 57–66 (cit. on p. 70).

[DGM06] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes. *k*-core organization of complex networks. *Physical Review Letters* 96 (2006), p. 040601 (cit. on p. 17).

[EK10] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010 (cit. on pp. 26, 28, 36, 53, 56, 110, 111, 113, 114).

[EA13] Marius Eidsaa and Eivind Almaas. *s*-core network decomposition: A generalization of *k*-core analysis to weighted networks. *Phys. Rev. E* 88 (6 2013), p. 062819 (cit. on p. 15).

[EW04] David Eppstein and Joseph Wang. Fast Approximation of Centrality. *J. Graph Algorithms Appl.* 8 (2004), pp. 39–45 (cit. on p. 120).

[ER60] P. Erdös and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci* 5 (1960), pp. 17–61 (cit. on p. 1).

[EH66] P. Erdős and A. Hajnal. On chromatic number of graphs and set-systems. *Acta Mathematica Academiae Scientiarum Hungarica* 17:1-2 (1966), pp. 61–99 (cit. on p. 13).

[ER04] Günes Erkan and Dragomir R. Radev. LexRank: Graph-based Lexical Centrality As Salience in Text Summarization. *J. Artif. Int. Res.* 22:1 (2004), pp. 457–479 (cit. on p. 108).

[FFF99] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On Power-law Relationships of the Internet Topology. In *SIGCOMM '99: Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 1999, pp. 251–262 (cit. on pp. 1, 56, 71).

[For10] Santo Fortunato. Community detection in graphs. *Physics Reports* 48:3–5 (2010), pp. 75–174 (cit. on pp. 135, 136).

[Fre82] Eugene C. Freuder. A Sufficient Condition for Backtrack-Free Search. *J. ACM* 29:1 (1982), pp. 24–32 (cit. on p. 13).

[Für98] Johannes Fürnkranz. *A Study Using n-gram Features for Text Categorization*. Tech. rep. OEFAI-TR-98-30. Austrian Research Institute for Artificial Intelligence, 1998 (cit. on p. 106).

[GSH12] Antonios Garas, Frank Schweitzer, and Shlomo Havlin. A k-shell decomposition method for weighted networks. *New Journal of Physics* 14:8 (2012) (cit. on p. 15).

[GMS13] David Garcia, Pavlin Mavrodiev, and Frank Schweitzer. Social Resilience in Online Communities: The Autopsy of Friendster. In *COSN '13: Proceedings of the First ACM Conference on Online Social Networks*, 2013, pp. 39–50 (cit. on pp. 55, 56).

[GTV11a] Christos Giatsidis, Dimitrios M. Thilikos, and Michalis Vazirgiannis. D-cores: Measuring Collaboration of Directed Graphs Based on Degeneracy. In *ICDM '11: Proceedings of the 11th IEEE International Conference on Data Mining*, 2011, pp. 201–210 (cit. on p. 15).

[GTV11b] Christos Giatsidis, Dimitrios M. Thilikos, and Michalis Vazirgiannis. Evaluating Cooperation in Communities with the k-Core Structure. In *ASONAM '11: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, 2011, pp. 87–93 (cit. on pp. 15, 17).

[Gia+14a] Christos Giatsidis, Fragkiskos D. Malliaros, Dimitrios M. Thilikos, and Michalis Vazirgiannis. CoreCluster: A Degeneracy Based Graph Clustering Framework. In *AAAI '14: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014, pp. 44–50 (cit. on pp. xii, 17, 137).

[Gia+14b] Christos Giatsidis, Bogdan Cautis, Silviu Maniu, Dimitrios M. Thilikos, and Michalis Vazirgiannis. Quantifying trust dynamics in signed graphs, the S-Cores approach. In *SDM '14: Proceedings of the 2014 SIAM International Conference on Data Mining*, 2014, pp. 668–676 (cit. on p. 15).

[GN02] M. Girvan and M. E. Newman. Community structure in social and biological networks. *Proc Natl Acad Sci* 99:12 (2002), pp. 7821–7826 (cit. on p. 1).

[GS12] David F. Gleich and C. Seshadhri. Vertex neighborhoods, low conductance cuts, and good seeds for local community methods.

In *KDD '12: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012, pp. 597–605 (cit. on p. 47).

[GL96] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996 (cit. on pp. 11, 112).

[Har13] Andrew Harkins. *Network Games with Perfect Complements*. Tech. rep. University of Warwick, 2013 (cit. on pp. 25, 26, 30, 56).

[HMB07] Samer Hassan, Rada Mihalcea, and Carmen Banea. Random-Walk Term Weighting for Improved Text Classification. In *ICSC '07: Proceedings of the Internation Conference on Semantic Computing*, IEEE Computer Society, 2007, pp. 242–249 (cit. on p. 107).

[Hea+08] John Healy, Jeannette Janssen, Evangelos Milios, and William Aiello. Characterization of Graphs Using Degree Cores. In *WAW '08: Algorithms and Models for the Web-Graph*, 2008, pp. 137–148 (cit. on p. 17).

[HD+13] Laurent Hébert-Dufresne, Antoine Allard, Jean-Gabriel Young, and Louis J. Dubé. Percolation on random networks with arbitrary k-core structure. *Phys. Rev. E* 88 (6 2013), p. 062820 (cit. on p. 17).

[Het00] Herbert W. Hethcote. The Mathematics of Infectious Diseases. *SIAM Rev.* 42:4 (2000), pp. 599–653 (cit. on p. 76).

[HYL12] Bonan Hou, Yiping Yao, and Dongsheng Liao. Identifying all-around nodes for spreading dynamics in complex networks. *Physica A: Statistical Mechanics and its Applications* 391:15 (2012), pp. 4012–4017 (cit. on p. 79).

[Hua+14] Xin Huang, Hong Cheng, Lu Qin, Wentao Tian, and Jeffrey Xu Yu. Querying K-truss Community in Large and Dynamic Graphs. In *SIGMOD '14: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 2014, pp. 1311–1322 (cit. on p. 76).

[Jia+10] Chuntao Jiang, Frans Coenen, Robert Sanderson, and Michele Zito. Text classification using graph mining-based feature extraction. *Knowl.-Based Syst.* 23:4 (2010), pp. 302–308 (cit. on p. 107).

[Joa98] Thorsten Joachims. Text Categorization with Suport Vector Machines: Learning with Many Relevant Features. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, 1998, pp. 137–142 (cit. on pp. 106, 121).

[Joa99] Thorsten Joachims. Transductive Inference for Text Classification Using Support Vector Machines. In *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, 1999, pp. 200–209 (cit. on p. 106).

[KKT05] David Kempe, Jon Kleinberg, and Éva Tardos. Influential Nodes in a Diffusion Model for Social Networks. In *ICALP '05: Proceedings of the 32nd International Conference on Automata, Languages and Programming*, 2005, pp. 1127–1138 (cit. on p. 70).

[KKT03] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the Spread of Influence Through a Social Network. In *KDD '03: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, pp. 137–146 (cit. on pp. 70, 81, 136).

[KKT15] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the Spread of Influence through a Social Network. *Theory of Computing* 11:4 (2015), pp. 105–147 (cit. on pp. 70, 81).

[KM27] W. O. Kermack and Ag McKendrick. A Contribution to the Mathematical Theory of Epidemics. *Proceedings of the Royal Society of London* 115:772 (1927), pp. 700–721 (cit. on p. 76).

[Kim+06] Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung-Hyon Myaeng. Some Effective Techniques for Naive Bayes Text Classification. *IEEE Trans. Knowl. Data Eng.* 18:11 (2006), pp. 1457–1466 (cit. on p. 106).

[KT96] Lefteris M. Kirousis and Dimitris M. Thilikos. The Linkage of a Graph. *SIAM J. Comput.* 25:3 (1996), pp. 626–647 (cit. on p. 13).

[Kit+10] M. Kitsak, L. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. Stanley, and H. Makse. Identification of Influential Spreaders in Complex Networks. *Nature Physics* 6:11 (2010), pp. 888–893 (cit. on pp. 26, 65, 71, 72, 79, 82, 86).

[KT05] Jon Kleinberg and Eva Tardos. *Algorithm Design*. Addison-Wesley Longman Publishing Co., Inc., 2005 (cit. on p. 81).

[KLB09] Jérôme Kunegis, Andreas Lommatzsch, and Christian Bauckhage. The Slashdot Zoo: Mining a Social Network with Negative Edges. In *WWW '09: Proceedings of the 18th International Conference on World Wide Web*, 2009, pp. 741–750 (cit. on p. 15).

[Kun+10] Jérôme Kunegis, Stephan Schmidt, Andreas Lommatzsch, Jürgen Lerner, Ernesto William De Luca, and Sahin Albayrak. Spectral Analysis of Signed Graphs for Clustering, Prediction and Visualization. In *SDM '10: Proceedings of the SIAM International Conference on Data Mining*, 2010, pp. 559–570 (cit. on p. 15).

[LCC14] Shibamouli Lahiri, Sagnik Ray Choudhury, and Cornelia Caragea. Keyword and Keyphrase Extraction Using Centrality Measures on Collocation Networks. *CoRR* (2014) (cit. on pp. 18, 116).

[Lan+] Man Lan, Chew-Lim Tan, Hwee-Boon Low, and Sam-Yuan Sung. A Comprehensive Comparative Study on Term Weighting Schemes for Text Categorization with Support Vector Machines. In *WWW '05: Proceedings of the 14th International Conference on World Wide Web*, pp. 1032–1033 (cit. on pp. 106, 124).

[Lan+09] Man Lan, Chew Lim Tan, Jian Su, and Yue Lu. Supervised and Traditional Term Weighting Methods for Automatic Text Categorization. *IEEE Trans. Pattern Anal. Mach. Intell.* 31:4 (2009), pp. 721–735 (cit. on p. 106).

[Lat08] Matthieu Latapy. Main-memory Triangle Computations for Very Large (Sparse (Power-law)) Graphs. *Theor. Comput. Sci.* 407:1-3 (2008), pp. 458–473 (cit. on p. 75).

[LS09] Silvio Lattanzi and D. Sivakumar. Affiliation networks. In *STOC '09: Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, 2009, pp. 427–434 (cit. on p. 26).

[LK02] Edda Leopold and Jörg Kindermann. Text Categorization with Support Vector Machines. How to Represent Texts in Input Space? *Mach. Learn.* 46:1-3 (2002) (cit. on p. 121).

[LF06] Jure Leskovec and Christos Faloutsos. Sampling from Large Graphs. In *KDD '06: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 631–636 (cit. on p. 137).

[LH08] Jure Leskovec and Eric Horvitz. Planetary-scale Views on a Large Instant-messaging Network. In *WWW '08: Proceedings of the 17th International Conference on World Wide Web*, 2008, pp. 915–924 (cit. on pp. 1, 17).

[LHK10a] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Predicting Positive and Negative Links in Online Social Networks. In *WWW '10: Proceedings of the 19th International Conference on World Wide Web*, 2010, pp. 641–650 (cit. on p. 19).

[LHK10b] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Signed Networks in Social Media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10. 2010, pp. 1361–1370 (cit. on p. 19).

[LKF07] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data* 1:1 (2007) (cit. on pp. 19–21).

[LKF05] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations. In *KDD '05: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2005, pp. 177–187 (cit. on p. 20).

[LK14] Jure Leskovec and Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. http://snap.stanford.edu/data. June 2014 (cit. on p. 18).

[Les+09] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large welldefined clusters. *Internet Mathematics* 6:1 (2009), pp. 29–123 (cit. on pp. 19, 42).

[Lew92] David Dolan Lewis. Representation and Learning in Information Retrieval. *Ph.D. Thesis, University of Massachusetts, Amherst, MA, USA* (1992) (cit. on p. 106).

[Li+14] Qian Li, Tao Zhou, Linyuan Lü, and Duanbing Chen. Identifying influential spreaders by weighted LeaderRank. *Physica A: Statistical Mechanics and its Applications* 404 (2014), pp. 47–55 (cit. on p. 78).

[Li+15] Rong-Hua Li, Lu Qin, Jeffrey Xu Yu, and Rui Mao. Influential Community Search in Large Networks. *Proc. VLDB Endow.* 8:5 (2015), pp. 509–520 (cit. on p. 17).

[LL08] Marina Litvak and Mark Last. Graph-based Keyword Extraction for Single-document Summarization. In *MMIES '08: Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, 2008, pp. 17–24 (cit. on p. 108).

[Liu+10] Lu Liu, Jie Tang, Jiawei Han, Meng Jiang, and Shiqiang Yang. Mining Topic-level Influence in Heterogeneous Networks. In *CIKM '10: Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010, pp. 199–208 (cit. on p. 56).

[Lod+02] Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. Text Classification Using String Kernels. *J. Mach. Learn. Res.* 2 (2002), pp. 419–444 (cit. on p. 106).

[Low+10] Yucheng Low, Joseph Gonzalez, Aapo Kyrola, Danny Bickson, Carlos Guestrin, and Joseph M. Hellerstein. GraphLab: A New Framework For Parallel Machine Learning. In *UAI '10: Proceed-*

ings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, 2010, pp. 340–349 (cit. on p. 137).

[Lü+11] Linyuan Lü, Yi-Cheng Zhang, Chi Ho Yeung, and Tao Zhou. Leaders in social networks, the delicious case. *PloS one* 6:6 (2011), e21202 (cit. on p. 78).

[Lux07] Ulrike Luxburg. A Tutorial on Spectral Clustering. *Statistics and Computing* 17:4 (2007), pp. 395–416 (cit. on p. 137).

[Maa+11] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. In *HLT '11: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 2011, pp. 142–150 (cit. on pp. 106, 122).

[MBW10] Arun S. Maiya and Tanya Y. Berger-Wolf. Sampling Community Structure. In *WWW '10: Proceedings of the 19th International Conference on World Wide Web*, 2010, pp. 701–710 (cit. on p. 137).

[MMF14] Fragkiskos D. Malliaros, Vasileios Megalooikonomou, and Christos Faloutsos. Estimating Robustness in Large Social Graphs. *Knowledge and Information Systems* (2014), pp. 1–34 (cit. on p. xii).

[MRV15] Fragkiskos D. Malliaros, Maria-Evgenia G. Rossi, and Michalis Vazirgiannis. Locating Influential Nodes in Complex Networks. *Manuscript* (2015) (cit. on pp. xii, 3).

[MS15] Fragkiskos D. Malliaros and Konstantinos Skianis. Graph-Based Term Weighting for Text Categorization. In *ASONAM '15: 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Social Media and Risk Workshop*, 2015 (cit. on pp. xi, 3).

[MSV15] Fragkiskos D. Malliaros, Konstantinos Skianis, and Michalis Vazirgiannis. A Graph-Based Framework for Text Categorization: Term Weighting and Selection as Ranking of Interconnected Features. *Manuscript* (2015) (cit. on pp. xi, 3).

[MV13a] Fragkiskos D. Malliaros and Michalis Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics Reports* 533:4 (2013), pp. 95 –142 (cit. on pp. xi, 135, 136).

[MV13b] Fragkiskos D. Malliaros and Michalis Vazirgiannis. To Stay or Not to Stay: Modeling Engagement Dynamics in Social Graphs. In *CIKM '13: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, 2013, pp. 469–478 (cit. on pp. xi, 3, 54–56).

[MV15] Fragkiskos D. Malliaros and Michalis Vazirgiannis. Vulnerability Assessment in Social Networks under Cascade-based Node Departures. *EPL (Europhysics Letters)* 110:6 (2015), p. 68006 (cit. on pp. xi, 3).

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008 (cit. on p. 106).

[MJ09] Vahideh H. Manshadi and Ramesh Johari. Supermodular Network Games. In *Allerton*, 2009, pp. 1369–1376 (cit. on pp. 25, 26, 30, 36, 56).

[MLK07] Alex Markov, Mark Last, and Abraham Kandel. Fast Categorization of Web Documents Represented by Graphs. In *Advances in Web Mining and Web Usage Analysis*, vol. 4811. 2007, pp. 56–71 (cit. on p. 107).

[MF09] Justin Martineau and Tim Finin. Delta TFIDF: An Improved Feature Space for Sentiment Analysis. In *ICWSM '09: Proceedings of the Third International Conference on Weblogs and Social Media*, 2009 (cit. on p. 106).

[MB83] David W. Matula and Leland L. Beck. Smallest-last Ordering and Clustering and Graph Coloring Algorithms. *J. ACM* 30:3 (1983), pp. 417–427 (cit. on p. 13).

[MN98] Andrew McCallum and Kamal Nigam. A Comparison of Event Models for Naive Bayes Text Classification. In *AAAI '98: Pro-*

ceedings of the Workshop on Learning for Text Categorization, 1998, pp. 41–48 (cit. on p. 106).

[Mie11] Piet Van Mieghem. *Graph Spectra for Complex Networks*. Cambridge University Press, 2011 (cit. on pp. 11, 85).

[MT04] Rada Mihalcea and Paul Tarau. TextRank: Bringing Order into Text. In *EMNLP '04: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004, pp. 404–411 (cit. on pp. 108, 116, 119).

[Mil67] Stanley Milgram. The small-world problem. *Psychology Today* 1:1 (1967), pp. 61–67 (cit. on p. 1).

[Mis+07] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *IMC '07: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, 2007, pp. 29–42 (cit. on p. 18).

[MPM13] Alberto Montresor, Francesco De Pellegrini, and Daniele Miorandi. Distributed k-Core Decomposition. *IEEE Transactions on Parallel and Distributed Systems* 24:2 (2013), pp. 288–300 (cit. on p. 14).

[MPSV02] Y. Moreno, R. Pastor-Satorras, and A. Vespignani. Epidemic outbreaks in complex heterogeneous networks. *The European Physical Journal B - Condensed Matter and Complex Systems* 26:4 (2002), pp. 521–529 (cit. on p. 78).

[ML02] Adilson E. Motter and Ying-Cheng Lai. Cascade-based attacks on complex networks. *Phys Rev E* 66 (14 2002), p. 065102 (cit. on pp. 54, 56, 59, 62).

[New02] M. E. J. Newman. Spread of epidemic disease on networks. *Physical Review E* 66:1 (2002), p. 016128 (cit. on p. 78).

[New10] Mark Newman. *Networks: An Introduction*. Oxford University Press, Inc., 2010 (cit. on pp. 1, 110, 111, 113, 114).

[New03] M.E.J. Newman. The Structure and Function of Complex Networks. *SIAM Review* 45:2 (2003), pp. 167–256 (cit. on pp. 1, 7, 53, 56).

[Nig+00] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text Classification from Labeled and Unlabeled Documents Using EM. *Mach. Learn.* 39:2-3 (May 2000), pp. 103–134 (cit. on p. 106).

[OS14] Michael P. O'Brien and Blair D. Sullivan. Locally Estimating Core Numbers. In *ICDM '14: 2014 IEEE International Conference on Data Mining*, 2014, pp. 460–469 (cit. on p. 14).

[OCL08] Kazuya Okamoto, Wei Chen, and Xiang-Yang Li. Ranking of Closeness Centrality for Large-Scale Social Networks. In *FAW '08: Proceedings of the 2nd Annual International Workshop on Frontiers in Algorithmics*, 2008, pp. 186–195 (cit. on p. 111).

[PT10] Georgios Paltoglou and Mike Thelwall. A Study of Information Retrieval Weighting Schemes for Sentiment Analysis. In *ACL '10: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 1386–1395 (cit. on p. 107).

[Par+14] Ha-Myung Park, Francesco Silvestri, U. Kang, and Rasmus Pagh. MapReduce Triangle Enumeration With Guarantees. In *CIKM '14: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, 2014, pp. 1739–1748 (cit. on p. 74).

[PSV02] Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic dynamics in finite size scale-free networks. *Phys. Rev. E* 65 (3 2002), p. 035108 (cit. on p. 84).

[PSV01] Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic Spreading in Scale-Free Networks. *Phys. Rev. Lett.* 86 (14 2001), pp. 3200–3203 (cit. on pp. 71, 78).

[PLG13] Akshay Patil, Juan Liu, and Jie Gao. Predicting Group Stability in Online Social Networks. In *WWW '13: Proceedings of the 22nd*

International Conference on World Wide Web, 2013, pp. 1021–1030 (cit. on p. 56).

[Pat+12] Akshay Patil, Juan Liu, Bob Price, Hossam Sharara, and Oliver Brdiczka. Modeling Destructive Group Dynamics in On-Line Gaming Communities. In *ICWSM '12: Proceedings of the Sixth International Conference on Weblogs and Social Media*, 2012, pp. 290–297 (cit. on p. 56).

[PKT14] Katerina Pechlivanidou, Dimitrios Katsaros, and Leandros Tassiulas. MapReduce-Based Distributed K-Shell Decomposition for Online Social Networks. In *SERVICES '14: Proceedings of the 2014 IEEE World Congress on Services*, 2014, pp. 30–37 (cit. on p. 14).

[PM13] Sen Pei and Hernán A. Makse. Spreading dynamics in complex networks. *Journal of Statistical Mechanics: Theory and Experiment* 2013:12 (2013), P12002 (cit. on p. 80).

[PSW96] B. Pittel, J. Spencer, and N. Wormald. Sudden emergence of a giant k-core in a random graph. *J. Combin. Theory Ser. B* 67:1 (1996), pp. 111–151 (cit. on p. 39).

[Pra+11] B. Aditya Prakash, Deepayan Chakrabarti, Michalis Faloutsos, Nicholas Valler, and Christos Faloutsos. Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks. In *ICDM '11: Proceedings of the 2011 IEEE 11th International Conference on Data Mining*, IEEE Computer Society, 2011, pp. 537–546 (cit. on p. 85).

[Pra+12] B. Aditya Prakash, Deepayan Chakrabarti, Nicholas C. Valler, Michalis Faloutsos, and Christos Faloutsos. Threshold conditions for arbitrary cascade models on arbitrary networks. *Knowledge and Information Systems* 33:3 (2012), pp. 549–575 (cit. on p. 85).

[QWH12] Louise Quick, Paul Wilkinson, and David Hardcastle. Using Pregel-like Large Scale Graph Processing Frameworks for Social Network Analysis. In *ASONAM '12: Proceedings of the 2012*

International Conference on Advances in Social Networks Analysis and Mining, 2012, pp. 457–463 (cit. on p. 76).

[RHC12] Ursula Redmond, Martin Harrigan, and Pádraig Cunningham. Mining Dense Structures to Uncover Anomalous Behaviour in Financial Network Data. In *MSM '11: Proceedings of the 2011 International Conference on Modeling and Mining Ubiquitous Social Media*, Boston, MA, 2012, pp. 60–76 (cit. on p. 76).

[Ree+06] Joel W. Reed, Yu Jiao, Thomas E. Potok, Brian A. Klump, Mark T. Elmore, and Ali R. Hurson. TF-ICF: A New Term Weighting Scheme for Clustering Dynamic Data Streams. In *ICMLA '06: Proceedings of the 5th International Conference on Machine Learning and Applications*, 2006, pp. 258–263 (cit. on p. 106).

[RAD03] Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. Trust Management for the Semantic Web. In *ISWC '03: Proceedings of the Second International Semantic Web Conference*, 2003, pp. 351–368 (cit. on p. 19).

[RD02] Matthew Richardson and Pedro Domingos. Mining Knowledge-sharing Sites for Viral Marketing. In *KDD '02: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002, pp. 61–70 (cit. on p. 70).

[Rob+96] S.E. Robertson, S. Walker, S. Jones, M.M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *TREC '96: Proceedings of Text Retrieval Conference*, 1996, pp. 109–126 (cit. on p. 106).

[Rob04] Stephen Robertson. Understanding inverse document frequency: On theoretical arguments for IDF. *Journal of Documentation* 60 (2004), p. 2004 (cit. on pp. 106, 117, 118).

[RMK11] Daniel M. Romero, Brendan Meeder, and Jon Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *WWW*, 2011, pp. 695–704 (cit. on pp. 26, 46).

[RMV15] Maria-Evgenia G. Rossi, Fragkiskos D. Malliaros, and Michalis Vazirgiannis. Spread It Good, Spread It Fast: Identification of Influential Nodes in Social Networks. In *WWW '15: Proceedings of the 24th International Conference on World Wide Web Companion*, 2015, pp. 101–102 (cit. on pp. xi, 3).

[Ros14] Ryan A. Rossi. Fast Triangle Core Decomposition for Mining Large Graphs. In *PAKDD '14: Advances in Knowledge Discovery and Data Mining*, 2014, pp. 310–322 (cit. on p. 76).

[RA14] Ryan A. Rossi and Nesreen K. Ahmed. Coloring Large Complex Networks. *Social Network Analysis and Mining* 4:1, 228 (2014), pp. 1–37 (cit. on p. 76).

[RV13] François Rousseau and Michalis Vazirgiannis. Graph-of-word and TW-IDF: new approach to ad hoc IR. In *CIKM '13: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, 2013, pp. 59–68 (cit. on pp. 108, 114, 116).

[RV15] François Rousseau and Michalis Vazirgiannis. Main Core Retention on Graph-of-Words for Single-Document Keyword Extraction. In *ECIR '15: Proceedings of the 37th European Conference on Information Retrieval*, 2015, pp. 382–393 (cit. on pp. 18, 108, 116, 119).

[RKV15] François Rousseau, Emmanouil Kiagias, and Michalis Vazirgiannis. Text Categorization as a Graph Classification Problem. In *ACL '15: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2015 (cit. on p. 107).

[SB88] Gerard Salton and Christopher Buckley. Term-weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manage.* 24:5 (1988), pp. 513–523 (cit. on p. 106).

[Sar+15] Ahmet Erdem Sariyuce, C. Seshadhri, Ali Pinar, and Umit V. Catalyurek. Finding the Hierarchy of Dense Subgraphs Using Nucleus Decompositions. In *WWW '15: Proceedings of the 24th*

International Conference on World Wide Web, 2015, pp. 927–937 (cit. on p. 16).

[Sar+13] Ahmet Erdem Saríyüce, Buğra Gedik, Gabriela Jacques-Silva, Kun-Lung Wu, and Ümit V. Çatalyürek. Streaming Algorithms for K-core Decomposition. *Proc. VLDB Endow.* 6:6 (2013), pp. 433–444 (cit. on p. 14).

[SPR11] Venu Satuluri, Srinivasan Parthasarathy, and Yiye Ruan. Local Graph Sparsification for Scalable Clustering. In *SIGMOD '11: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, 2011, pp. 721–732 (cit. on p. 137).

[Sch07] Thomas Schank. Algorithmic Aspects of Triangle-Based Network Analysis. *Ph.D. Thesis, University of Karlsruhe, Germany* (2007) (cit. on p. 75).

[Seb02] Fabrizio Sebastiani. Machine Learning in Automated Text Categorization. *ACM Comput. Surv.* 34:1 (2002), pp. 1–47 (cit. on pp. 104, 106–108, 110, 123).

[Sei83] Stephen B. Seidman. Network Structure and Minimum Degree. *Social Networks* 5 (1983), pp. 269–287 (cit. on pp. 11, 13, 17, 25–27, 33, 112).

[SHN15] Masumi Shirakawa, Takahiro Hara, and Shojiro Nishio. N-gram IDF: A Global Term Weighting Scheme Based on Information Distance. In *WWW '15: Proceedings of International World Wide Web Conference*, 2015 (cit. on p. 106).

[SBM96] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted Document Length Normalization. In *SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996, pp. 21–29 (cit. on p. 110).

[Sin+99] Amit Singhal, John Choi, Donald Hindle, David D. Lewis, and Fernando Pereira. AT&T at TREC-7. In *TREC '99: Proceedings of the 7th Text Retrieval Conference*, 1999, pp. 239–252 (cit. on p. 110).

[Str88] Gilbert Strang. *Linear Algebra and Its Applications*. Brooks Cole, 1988 (cit. on pp. 11, 112).

[SL01] Aixin Sun and Ee-Peng Lim. Hierarchical Text Classification and Evaluation. In *ICDM '01: Proceedings of the 2001 IEEE International Conference on Data Mining*, 2001, pp. 521–528 (cit. on p. 106).

[Tan+08] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. ArnetMiner: Extraction and Mining of Academic Social Networks. In *KDD '08: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 990–998 (cit. on p. 1).

[Tan+09] Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social Influence Analysis in Large-scale Networks. In *KDD '09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 807–816 (cit. on p. 56).

[TG15] Nikolaj Tatti and Aristides Gionis. Density-friendly Graph Decomposition. In *WWW '15: Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 1089–1099 (cit. on p. 16).

[Dbl] *The DBLP Computer Science Bibliography*. URL: http://www.informatik.uni-trier.de/~ley/db/ (cit. on p. 20).

[TBP09] Michael Trusov, Randolph E. Bucklin, and Koen Pauwels. Effects of Word-of-Mouth Versus Traditional Marketing: Findings from an Internet Social Networking Site. *Journal of Marketing* 73:5 (2009), pp. 90–102 (cit. on p. 70).

[Tso08] Charalampos E. Tsourakakis. Fast Counting of Triangles in Large Real Networks without Counting: Algorithms and Laws. In *ICDM '08: Proceedings of the Eighth IEEE International Conference on Data Mining*, 2008, pp. 608–617 (cit. on p. 74).

[Tso+09] Charalampos E. Tsourakakis, U. Kang, Gary L. Miller, and Christos Faloutsos. DOULION: Counting Triangles in Massive Graphs with a Coin. In *KDD '09: Proceedings of the 15th ACM*

SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009, pp. 837–846 (cit. on p. 74).

[Uga+11] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow. The Anatomy of the Facebook Social Graph. *Arxiv Preprint* (2011) (cit. on p. 17).

[Uga+12] Johan Ugander, Lars Backstrom, Cameron Marlow, and Jon Kleinberg. Structural Diversity in Social Contagion. *PNAS* 109:16 (2012), pp. 5962–5966 (cit. on pp. 26, 28, 53, 56).

[Vis+09] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the evolution of user interaction in Facebook. In *WOSN '09: Proceedings of the 2nd ACM Workshop on Online Social Networks*, 2009, pp. 37–42 (cit. on p. 18).

[WC12] Jia Wang and James Cheng. Truss decomposition in massive networks. *Proc. VLDB Endow.* 5:9 (2012), pp. 812–823 (cit. on pp. 16, 72, 74, 75, 81).

[WDL05] Wei Wang, DiepBich Do, and Xuemin Lin. Term Graph Model for Text Classification. In *ADMA '05: First International Conference on Advanced Data Mining and Applications*, 2005, pp. 19–30 (cit. on p. 107).

[Wan+03] Yang Wang, Deepayan Chakrabarti, Chenxi Wang, and Christos Faloutsos. Epidemic Spreading in Real Networks: An Eigenvalue Viewpoint. In *SRDS '03: Proceedings of the 22nd International Symposium on Reliable Distributed Systems*, IEEE Computer Society, 2003, pp. 25–34 (cit. on pp. 84, 85).

[Wat02] D.J. Watts. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences* 99:9 (2002), p. 5766 (cit. on p. 66).

[Wu+13] Shaomei Wu, Atish Das Sarma, Alex Fabrikant, Silvio Lattanzi, and Andrew Tomkins. Arrival and Departure Dynamics in Social Networks. In *WSDM '13: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, 2013, pp. 233–242 (cit. on pp. 26, 31, 36, 46, 56).

[YP97] Yiming Yang and Jan O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, 1997, pp. 412–420 (cit. on p. 119).

[Zah+10] Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, and Ion Stoica. Spark: Cluster Computing with Working Sets. In *HotCloud '10: Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, 2010 (cit. on p. 137).

[ZZ13] An Zeng and Cheng-Jun Zhang. Ranking spreaders by decomposing complex networks. *Physics Letters A* 377:14 (2013), pp. 1031–1035 (cit. on p. 79).

[Zha+08] Guo-Qing Zhang, Guo-Qiang Zhang, Qing-Feng Yang, Su-Qi Cheng, and Tao Zhou. Evolution of the Internet and its cores. *New Journal of Physics* 10:12 (2008), pp. 123027+ (cit. on p. 17).

[Zha+13] Xiaohang Zhang, Ji Zhu, Qi Wang, and Han Zhao. Identifying influential nodes in complex networks with community structure. *Knowledge-Based Systems* 42 (2013), pp. 74–84 (cit. on p. 79).

[ZP12] Yang Zhang and Srinivasan Parthasarathy. Extracting Analyzing and Visualizing Triangle K-Core Motifs within Networks. In *ICDE '12: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, 2012, pp. 1049–1060 (cit. on pp. 16, 17, 26, 72, 74, 76, 81).

COLOPHON

This document was typeset in LaTeX using the typographical look-and-feel `classicthesis`. The graphics in this dissertation are generated using the `Matlab` numerical computing environment, the `R` language, the `Ipe` extensible drawing editor and `pgf/tikz`. The bibliography is typeset using `biblatex`.