Computational Social Network Analysis of Authority in the Blogosphere

Vom Fachbereich Informatik der TU Kaiserslautern

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften (Dr. rer. nat.)

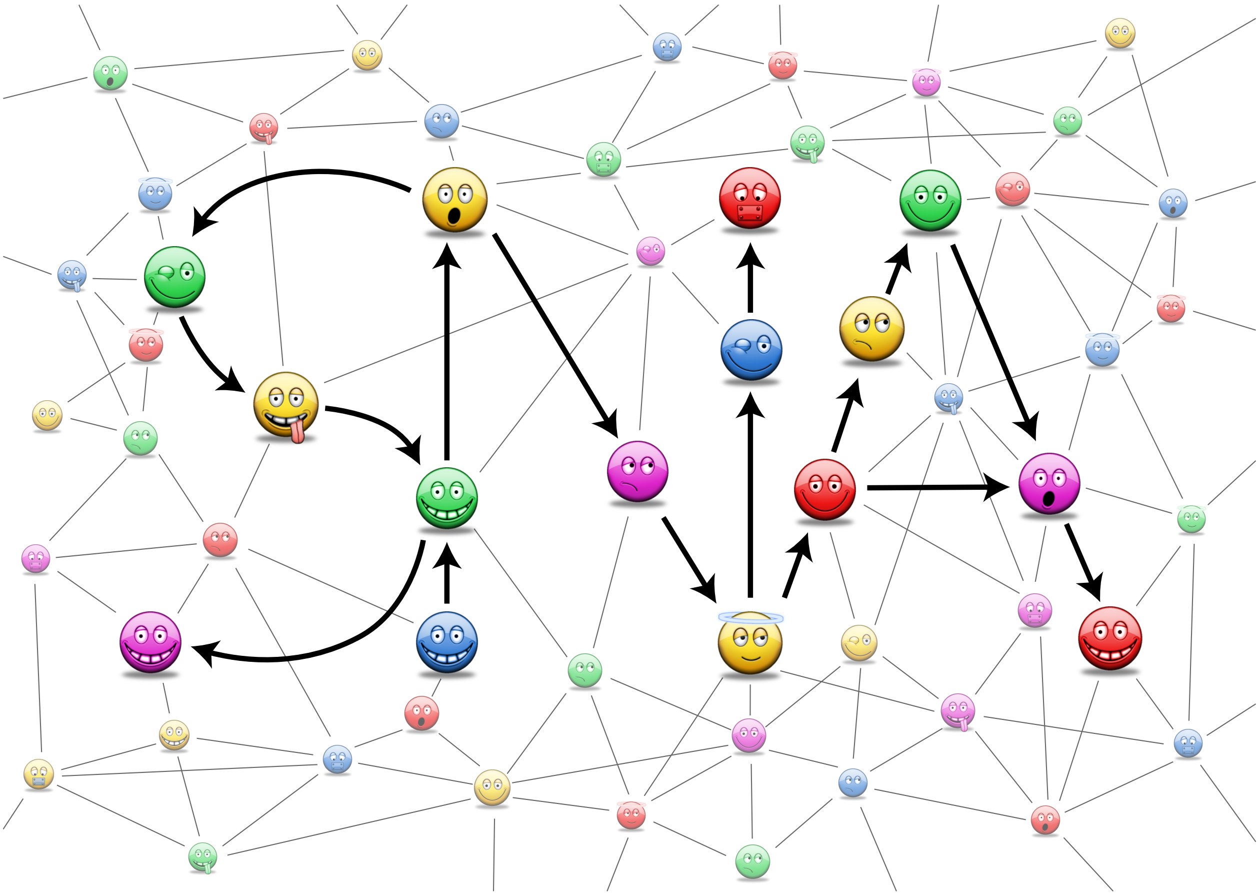
genehmigte Dissertation von

Dipl.-Inf. Darko Obradović
Deutsches Forschungszentrum
für Künstliche Intelligenz (DFKI) GmbH
Trippstadter Straße 122
67663 Kaiserslautern

Datum der wissenschaftlichen Aussprache: 28.09.2012

Erster Berichterstatter:	Prof. Dr. Prof. h.c. Andreas Dengel
Zweiter Berichterstatter:	Prof. Dr. Katharina Zweig
Vorsitzender:	Prof. Dr. Paul Müller
Dekan:	Prof. Dr. Arnd Poetzsch-Heffter

Zeichen der TU im Bibliotheksverkehr: D 386



Abstract

Social Media have gained more and more importance in many areas of our daily lives. One of the first media types in this field were weblogs, which allow everyone to easily publish content online. For weblogs, the reliable algorithmic detection of importance based on social reputation is still an open issue. In this thesis we attempt to measure this authority with algorithms from the field of Social Network Analysis, which have to be scalable, transparent and thoroughly evaluated.

Social scientists have identified very specific characteristics for the elite group of influential tob bloggers, which are well represented by the network core/periphery model from Borgatti & Everett. We approximate this model with a scalable algorithm based on the concept of *k*-cores from Seidman. For evaluation we collect datasets of thousands of top blogs in six different languages, in order to compare and cross-check the results. These are also compared to random networks, in order to show the significance of the findings. Remaining detection problems are engaged with anomaly detection and network filtering algorithms, which lead to an overall reliable detection process according to our evaluations.

In a second step, this thesis transfers these insights to a practical problem. A complete mining and analysis methodology for the monitoring of specific entities in the blogosphere is developed and evaluated. It consists of the search for relevant blog articles, which proves to be highly effective, and the authority measurement of these articles for potential end users in business scenarios, which are validated with respect to soundness. The resulting tool, the "Social Media Miner", integrates this methodology, combined with text processing methods, in an extensive analysis process and received very good feedback.

Contents

CHAPTER 1

Introduction

This chapter presents the motivation of this thesis, and introduces the two main concepts. The scientific field of Social Network Analysis (SNA), which is our toolbox of choice, and the blogosphere, which is the subject of our analyses. It also presents the rationale and the research goals for the following chapters.

1.1 Motivation

In recent years, Social Media have gained more and more importance in our daily lives, whether it is in journalism, politics, business or marketing. One of the first media types in this field were weblogs, which allow everyone to publish content online without the need for extensive technical knowledge about web page design and deployment.

An increase in importance naturally is followed by a demand in ranking, like the search engine competition has shown in the late nineties, when the Internet itself gained more and more importance.

For weblogs, this detection of importance, which is not based on keywords, but on social reputation, is still an open issue. Current solutions do not leverage the underlying structure to its full extent, as we will show.

Respecting the findings of the literature, and exploiting the underlying structure of the emerged social networks, it is our goal to find a computational and reliable way to detect the most important weblogs,

1.2 Social Network Analysis (SNA)

SNA is a relatively young interdisciplinary scientific field that deals with the thorough analysis of relational networks among specific groups of people. The discipline has its roots in the beginning of the 20$^{\text{th}}$ century within the field of sociology. The main idea is to analyse the structure of and the interactions within social groups.

The methods for this analysis are based on the mathematical field of graph theory, where the persons are represented by nodes and their relations by edges. Individual nodes or the network as a whole are measured with appropriate metrics. These include, for example, the centrality of a node in the network, the density of the network, and many more complex and more sophisticated metrics. A good comprehensive introduction into this field is given by Scott (2000).

The Historical Origins

A detailed overview of the historical development of SNA is given by Freeman (2004). The first methodological foundations of SNA were established by Jacob Levy Moreno's *Sociometry* in the 1930's. He originally came up with the idea to represent persons and their relations in a network structure and to analyse these systematically. Hence, this is called the "birth of Social Network Analysis" by Freeman. However, the ideas of Moreno did not spread widely, and the following decades were termed the "dark ages" in consequence, where the field hardly advanced for more than 30 years.

The next milestone was the "renaissance of social network analysis" starting in the year 1963 at Harvard University, when Harrison White joined the department of social relations. He enforced a structural perspective on social relations, and disseminated this idea in various courses and papers. His numerous students adopted this perspective, and since a lot of them became active researchers in the field, his ideas began to spread. This was the starting point for today's understanding of SNA.

Six Degrees of Separation

A very famous term from social networks are the *six degrees of separation* treated by Barabási (2003). It is based on a hypothesis of the Hungarian author Frigyes Karinthy from 1929, where he postulated the idea, that every person is connected to any other person in the world by at most five acquaintances, i. e., at most six steps away. This is then called the *small-world phenomenon*.

The psychologist Stanley Milgram from the Harvard University, tried to verify this with an experiment in 1967. He sent 60 packets to random persons in Omaha and Wichita, which should reach a specific target person in Boston via acquaintances only. Three of the packets actually reached the target persons, via 5.5 steps in average. This was considered to have validated the hypothesis.

There is a lot of criticism concering the scientific methodology, and thus the significance of the experiment. However, the result is mostly responsible for the popularity of this hypothesis, and it is known by a lot of people, who are not related to SNA otherwise.

The Internet Age

The upcoming success of the Internet and the World Wide Web (WWW) in the late 1990's, and especially the subsequent rise of the *Web 2.0* (O'Reilly, 2005), accompanied by numerous Online Social Network (OSN) sites like Facebook[1], gave a veritable boost to the discipline lately. It is now also of interest for computer scientists, especially in the field of Artificial Intelligence.

In retrospection, the invention of the PageRank algorithm for website rankings in 1998 (Page et al., 1998), along with the launch of the search engine Google[2] demonstrated the power of SNA in the Internet. Despite their late start in the market, the ranking quality convinced so many users that Google was able to overtake its well-established competitors, and has become the sole dominator in the search engine market by today.

There are many kinds of networks available for analysis on the Internet. These can be either closed and well-defined OSN sites like Facebook and LinkedIn[3], or open, less formal networks like the Usenet or the blogosphere. Research is partially driven by scientific curiosity, or by commercial interests in advertising, etc.

Modern SNA

Traditionally, SNA researchers conducted mainly qualitative studies on relatively small networks, like families, classrooms, etc. Since the Internet age, the focus has shifted to quantitative research on very large web-based networks. This led to an emphasis of highly sophisticated network metrics, efficient graph algorithms,

[1] http://www.facebook.com/

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

[3] http://www.linkedin.com/

and mathematical network models. The most popular book for this modern SNA methodology is written by Wasserman et al. (1994), more up-to-date overviews are given by Newman (2003) and Brandes & Erlebach (2005). Since the focus has shifted from sociological issues towards computational issues, this direction is occasionally called *Computational Social Network Analysis*.

Once the power of the SNA methodology increased, the range of applications also did. Nowadays the methods are not applied to social networks only, but also to biological networks, computer networks, semantic networks, etc.

There exist several tools like Pajek[4], that provide the researcher with all important metrics required for a standard analysis of a network. More special tools like Gephi[5] enable an explorative analysis of large networks via interactive network visualisations. For innovative research, which does not only apply the existing methods, but also includes the development of special algorithms and visualisations, standard tools are not applicable. There exists a number of extensible network analysis programming frameworks like JUNG[6] that are suited for this type of research.

1.3 The Blogosphere

Weblogs, usually abbreviated to *blogs*, are an interesting phenomenon that arised with the Web 2.0. They are commonly defined as "dynamic Internet pages containing articles in reverse chronological order" (Blood, 2002). The set of all blogs on the WWW forms the so-called *blogosphere*[7].

The revolutionary new thing about blogs was the ease-of-use for authors. Various blog hosting services like Wordpress[8] offer ready-to-use systems, where the author can concentrate on writing and publishing. No knowledge about web servers, software installation and web techniques is required. This dramatically extended the range of potential authors of content in the WWW.

Different Types of Blogs

Blogs can be utilised for various purposes by their authors, Herring et al. (2004) have conducted a genre analysis of weblogs, based on a two-dimensional categorisation

[4] http://vlado.fmf.uni-lj.si/pub/networks/pajek/

[5] http://gephi.org/

[6] http://jung.sourceforge.net/

[7] some authors prefer the term *blogspace* though

[8] http://www.wordpress.com

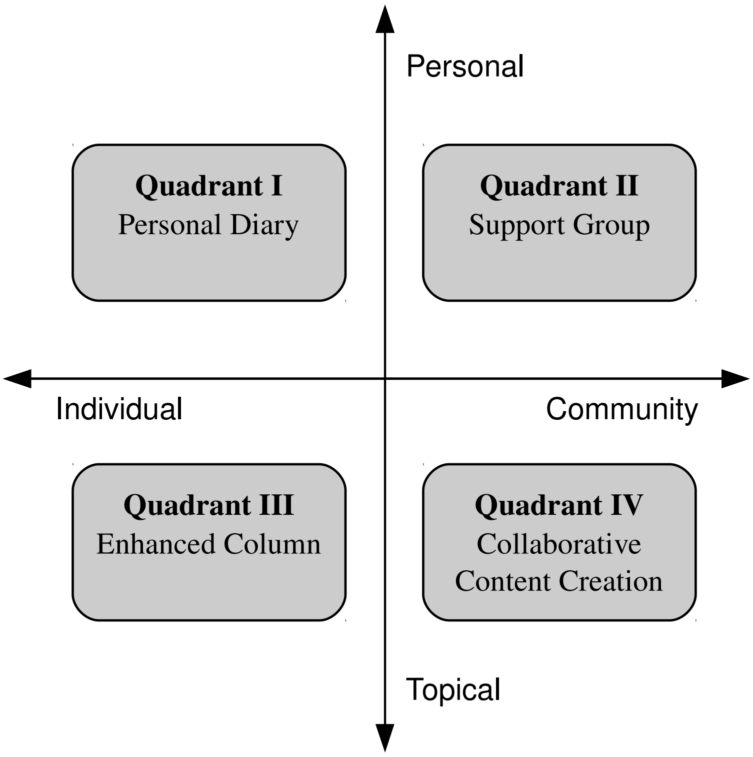


Figure 1.1: Two-dimensional genre classification for weblogs according to Krishnamurthy (2002)

for blogs from Krishnamurthy (2002) with four quadrants, as illustrated in Figure 1.1. The first dimension is the type of author, which is either an individual or a community of authors. The second dimension refers to the content of the blog articles, which can be either private or *topical*, i. e., focusing on a specifc topic of interest only.

The individual private quadrant contains the typical personal online diaries. The community private quadrant is termed *support groups* and plays only a minor role. The individual topical quadrant is referred to as *enhanced column*, where semi-professional authors comment daily politics, review mobile phones, etc. The community topical quadrant extends this with a variety of authors, and often a more professional editorial structure.

We try to adhere to these genres as close as possible in this thesis, but there always exist special cases and exceptions. Furthermore, the borderline between blogs and,

e. g., online news sites of journals like the New York Times[9] or corporate press release sites is not clearly defined, as those would also match the definition. We have to rely on a reasonable intuition here.

State of the Blogosphere

There exist two recent empirical overviews of the state of the blogosphere in the year 2010, published online by Technorati (Sobel, 2010) and The Blog Herald (Branckaute, 2010).

Reliable data is hard to obtain in an open, decentralised ecosystem like the blogosphere. Therefore, even the number of blogs worldwide is no more than a very uncertain estimate. Technorati's report is somewhat biased, since their data was gathered by respondents reached via their network, preferably from the United States. The Blog Herald's data is more universal here, since they based their findings on the Blogpulse index, with more than 150 million blogs.

Concerning blogger demographics, both studies agree in the main aspects. 70% of all bloggers are hobbyists with no income from their blog. The rest comprises part-timers, self-employeds and professionals. 66% of the authors are male, and about the same share is in the age group between 18 and 44 years.

The activity of bloggers in the frequency of postings varies a lot, ranging from less than once a month up to multiple times a day. Overall, 75% of all authors write at least one article per week and can be considered as active.

The various languages of blogs are measured by the Blog Herald's report. According to them, the majority of 37% of all blogs is written in Japanese, while English is used in 36% of all blogs. Chinese blogs make up 8% of the blogosphere, and all other languages have a share of less than 3%. The main still noticeable ones are Spanish, Italian (both 3%), Russian, Portuguese, French (all three 2%), Farsi and German (both 1%).

Linking in the Blogosphere

Following the principles of the Web 2.0 (O'Reilly, 2005), blogs offer very rich possibilities for interaction. Authors can include textual and multimedia content in their articles, but also link to related content of any form, refer to articles in other blogs, or let visitors post comments to the articles.

[9] http://www.nytimes.com/

Thus blogs can and do link to each other, either by mentioning other blog entries in their articles, in comments to these articles, or by explicitly recommending other blogs in a link set, the so-called *blogroll*. The blogroll typically comprises blogs the author recommends for reading, or the blogs of his friends and acquaintances.

The resulting network forms the complete blogosphere according to our understanding, i. e., not only the blogs themselves, but also all connections among them.

Research in the Blogosphere

The blogosphere attracted many researchers eagerly analysing its structure and dynamics. This is usually done quantitatively with methods and tools from the field of SNA. We briefly present a selection of the most prominent studies on the various aspects of the blogosphere.

A lot of studies focused on the structure of the blogosphere network and its dynamic evolution over time (Adar et al., 2004a; Kumar et al., 2004). Results implied the common hypothesis of the division into a minority of authoritative "opinion leaders" (Park, 2004; Delwiche, 2005), and the majority of less visible blogs in the "long tail" (Shirky, 2003).

A second structural aspect is the formation of communities in the blogosphere, usually based on shared interests, like politics, technology, etc. The first study on this aspect was conducted by Adamic & Glance (2005) on the political blogosphere around the 2004 U.S. presidentship elections. More case-studies, models and algorithms followed later (Chin & Chignell, 2006; Zhou & Davis, 2006; Chau & XU, 2007).

Another aspect of interest is the dynamics of article citations, e. g., news spread (Gruhl et al., 2004; Kumar et al., 2005) and discussions (Herring et al., 2005). These studies showed the wealth of information that could be harvested from link analyses on article level.

Other studies also investigated new, more sophisticated aspects like search (Bansal & Koudas, 2007) and credibility metrics (Ulicny & Baclawski, 2007).

A-List Blogs

One of the findings is the discovery of the *A-List* blogs (Blood, 2002; Marlow, 2004; Park, 2004; Delwiche, 2005), described by Herring et al. (2005) as "those that are most widely read, cited in the mass media, and receive the most inbound links from other blogs". These explorative and socially motivated studies have revealed that these blogs also heavily link among each other, but rarely to the rest of the blogosphere. This rest is often referred to as the *long tail* and consists of millions of blogs that are only partially indexed (Deep Web Phenomenon[10]).

In summary, there is a broad consensus about three attributes that characterise the group of A-List blogs, to which we will refer a number of times in this thesis:

1. A-List blogs are often linked to from the long tail
2. A-List blogs often link to each other
3. A-List blogs rarely link to the long tail

Ranking Blogs

There obviously is a demand for rankings in the blogosphere, serving as a motivation for blog authors on the one side, and as a filter for blog readers on the other side. Ranking lists are compiled and published by multiple commercial companies, e. g., Technorati[11], Alianzo [12], or Twingly[13].

When looking through these lists, one will usually find roughly the same set of blogs, but in a very different order, although all these rankings are based on algorithms counting inbound links. The discrepancy of the algorithms depends on the various parameters and weights of the unpublished ranking algorithms.

1.4 Rationale

In this thesis, we take a closer look at the aspect of *authority* in the blogosphere. This term summarises concepts like influence, reach, reputation, etc. It is a property of the small group of top blogs referred to as the *A-List* in the literature.

Problem Statement

As described in Section 1.3, blogs have become an important information channel for the distribution of mostly well-elaborated personal opinions and grassroot journalism.

[10] The Deep Web Phenomenon describes the difficulty to really know the size of the Internet, as it is open and decentralised. Estimates vary up to 80% of web pages that shall be unknown to search engines.

[11] http://technorati.com/blogs/top100

[12] http://www.alianzo.com/en/top-blogs

[13] http://www.twingly.com/top100

This applies to politics, economics, commercial products, personalities, etc. For a large audience, this channel is very valuable, opposed to corporate websites, webshops, online forums, etc. The key to this value however is a certain authority of these blogs, like described before.

There is a multitude of ranking services on the web, but almost all of them are intransparent with their algorithms. Furthermore, since they are usually based on counting inbound links of the blogs, results highly depend on their index of blogs. As the blogosphere is an open, decentralised, unorganised space, these indexes are usually far from complete, and lead to biased results. The same is true for the different parameters and weights.

This issue is seconded by Herring et al. (2005), who compiled a list of top blogs based on three different Top 100 lists. They included only blogs that were listed in two of these three Top 100 lists, ignoring their rank at first. They ended up with only 45 blogs, which illustrates well the enormous discrepancy in the ranking algorithms, since all of them tried to rank the very same thing.

Research Goals

While all ranking algorithms focus mostly on the first A-List characteristic, namely a large number of inbound links, we decide to look into the effect of the other two characteristics. These two, and especially the second one, the intensive linking among A-List blogs, demand a certain level of cohesion among A-List blogs, which has been mostly ignored up to date. This seems to be well-suited for further quantitative analyses concerning cohesion. By now, there has been no large-scale quantitative study yet, using these particular structural properties of the A-List subnetwork.

With a thorougly sound scientific network analysis methodology and a selection of parameters based on previous theoretical findings, we attempt to provide a transparent classification of authority for blogs.

In the course of this thesis we try to answer the following two research questions.

1. How can A-List blogs be identified reliably, and how can the borderline to the long tail be handled?
2. How can this knowledge be used in practical problems of specific information needs in the blogosphere?

While the first question is targeted at general, basically sociological insights about the blogosphere as a whole, the second question is more specific to concrete information needs. Whenever a user is interested in how a personality, a company, a product or a technology is perceived by the Internet audience, the authoritative blogs and their articles about this specific entity are of interest, regardless of the rest of the blogosphere.

Outline

The rest of this thesis is organised as follows. In Chapter 2 we introduce all the relevant SNA concepts and methods that we use in the remaining chapters to conduct and evaluate our analyses. In Chapter 3 we present our method for the data aggregation of the blog samples that are used for the A-List detection. This detection process is extensively described in the course of Chapter 4. This will answer the first research question, and constitutes the main aspect of this thesis. We then present an application of the findings in Chapter 5, where a highly automated blog monitoring tool for specific interests is described in detail. This will answer the second research question. Finally, the thesis in concluded in Chapter 6 with a critical discussion and an outlook to future work.

CHAPTER 2

SNA Methodology

This chapter first introduces the basic SNA concepts and notations, and then discusses the relevant aspects and the related literature when analysing large complex networks. It finally presents the specific methods that are used in the subsequent chapters for evaluatiing analysis results.

2.1 Basic Concepts and Notations

First of all, we summarise the SNA-specific terms and notations we adhere to in the following sections and chapters.

The Network

The term *network* from SNA and the term *graph* from Graph Theory are used synonymously in this thesis. It depends on the context, which one is preferred.

A graph G is defined as $G = (V, E)$, with V being the set of *vertices* or *nodes*, and $E = (V \times V)$ being the set of *edges* or *links* of the graph. $n = |V|$ is the number of nodes, and $m = |E|$ is the number of edges in the graph.

Graphs may be directed or undirected. In an undirected graph, the edge (a, b) is equal to the edge (b, a), and both endpoints have the same role. In the directed case, the order becomes important. An edge (s, t) implies a direction from the *source* node s to the *target* node t. There could be in parallel an edge (t, s) as well.

The function $succ(v)$ returns the set of all successor nodes of the node v, and the function $pre(v)$ returns the set of all predecessor nodes of v.

In a *simple graph*, parallel edges with the same endpoints cannot exist. Also, loops may not exist, that is, an edge with the same node on both ends. If parallel edges are allowed, the graph is called a *multi graph*.

Node Degrees

For each node v the function $deg(v)$ returns the nodal degree of a node in an undirected graph, which is the number of edges attached to that node.

In a directed graph, $indeg(v)$ returns the number of incoming edges, i. e., the number of edges in which v is the target, and $outdeg(v)$ returns the number of outgoing edges, i. e., the number of edges in which v is the source. We define the *summed degree* as $sumdeg(v) = indeg(v) + outdeg(v)$.

When listing the degrees of all nodes, this is called the *degree sequence* of the network. For an undirected network, this is a list of natural numbers including zero, for an undirected network this is a list of two-tuples, for the indegree and the outdegree of each node.

The statistical distribution of the degrees is called the *degree distribution*. It denotes for each degree value d the fraction of nodes in the network with exactly this degree. In a probabilistic view, the same function result is interpreted as the probability that a randomly selected node has the given degree d. The degree distribution is a very important characteristic of a network, at which we will have a closer look later in Section 2.2.4.

Paths

A *path* is a connection between two nodes, along one edge, if they are directly connected, i. e., *neighbours*, or along a number of subsequent edges if they are not neighbours. In case of a directed graphs, edges can only be considered in the right direction of course.

There can be multiple different paths from one node to another. The concept of a *shortest path* is of very high interest here. It is defined as the path with the least number of edges. The number of edges in between is defined as the *distance* between the nodes. In some cases there may be multiple shortest paths as well, but the distance remains the same.

Partitionings

A network can be partitioned into several disjoint sets of nodes.[1] A partitioning P of a network G is given as a set of n partitions P_1 to P_n, where each partition is a subset of V, and for all pairs $i \neq j$, $P_i \cap P_j = \emptyset$. The edges do not play any role here.

Connectivity

An undirected network is connected, if there exists a path from each node to every other node. If not, the network can be separated into a number of *connected components*, which are partitions of connected nodes, with no connections between nodes in the different partitions.

For directed networks, we have to distinguish the two concepts of *weak connectivity*, which is the same as the connectivity in an undirected network, when ignoring the directions of the edges. *Strong connectivity* is defined by respecting these directions. A group of nodes is strongly connected, if there exists a directed path from every node to every other one.

In these two cases the network is separated into a number of *weakly connected components*, or *strongly connected components* respectively.

2.2 Large Complex Networks

When analysing large networks with thousands or even millions of nodes, a couple of things have to be considered. Throughout the recent years, SNA researchers have provided according experiences and methodological suggestions in the literature (Newman, 2003), which we summarise in this section.

2.2.1 Metrics and Algorithms

As described in Section 1.2, the general focus in SNA shifted from explorative, visual methods to algorithms and metrics. The results of the metrics are then plotted to a suitable chart and interpreted accordingly. This can be based on the raw data or on statistical properties of the raw data.

[1] It is important to note that an arbitrary partitioning of a network is not necessarily related to the *Graph Partitioning Problem* in Mathematics, which specifically tries to find a partitioning with a minimal cut between all partitions.

For a more objective interpretation of metrics and their statistical properties, the relatively new method of comparison to random networks proved to be very useful (Alon, 2007). This method is described in detail in Section 2.3, and is used for evaluation purposes in this thesis.

2.2.2 Sparsity

One typical property of large networks is their sparsity with respect to the number of edges. Theoretically, the number of possible edges grows quadratically with the number of nodes contained in a network. A graph G with n nodes may contain up to $O(n^2)$ edges.

In real-world networks however, the number of edges is in the same order of magnitude as the nodes in nearly all cases. That means that a typical network G with n nodes contains $c \cdot n$ edges, with c being a constant number. This number depends a lot on the origin of the network. For example, we know from the Anthropologist Robin Dunbar (1993) that a human being has a hard time to maintain an intensive stable relationship to more than 150 other human beings at the same time.[2] So no matter how large the population may be, the number of relations will be constantly 150 times higher in such networks. The same is true for other types of networks, but with other constants of course.

As a consequence, the SNA literature assumes $n \approx m$ for large networks. This is an important fact for scalability issues, since a researcher may assume the network data to scale with the number of nodes.

2.2.3 Algorithmic Complexity

The algorithms used to analyse large networks should meet some requirements concerning their complexity.

The authors of Pajek suggested that the runtime of these algorithms has to be *sub-quadratic*, i. e., in $O(m \cdot \log m)$ or $O(m \cdot \sqrt{m})$. Optimally, an algorithm should run in linear time of course, i. e., in $O(m)$. Taking sparsity into account, it apparently does not matter if you base the runtime on n or m in terms of complexity classes.

Concerning storage complexity, algorithms should not need any more than linear space, i. e., $O(n+m)$. With thousands of nodes, a quadractic adjacency matrix, that needs to reside in main memory for satisfiable access times, would consume too

[2] this is often referred to as *Dunbar's number* in the literature

much storage space and dramatically slow down the algorithm. Furthermore, since large graphs tend to be very sparse, most of the space would be wasted anyway.

2.2.4 Degree Distribution

In the course of Computational SNA history, it has been recognised that the *degree distribution* of a network is a very important characteristic for it (see Newman, 2003, Section III.C). Once you know the degree distribution, and if it fits well to one of the well-known standard distributions, a lot of properties can be assumed to be similar to the reference model.

This becomes more complicated when dealing with directed networks, since there exists a degree distribution for indegrees and another one for outdegrees, which are coupled via the nodes. In most cases, these are regarded independently, although their correlation might contain additional insights. Furthermore, depending on the goals of the analysis, only one of the distributions might be of interest, which is the indegree distribution in most cases related to authority.

One of the most frequently observed class of degree distributions is the one of the *scale-free networks* described by Barabasi & Albert (1999), which follows a power law. This is found in many large online networks like the Internet, citation networks, phone call networks, biological networks, etc. (Faloutsos et al., 1999). A model that produces such a type of network is the "preferential attachment model", in which new nodes are most likely to connect with the most popular nodes in the network. and thus further strengthen this effect.

In our use-case, the link structure in the blogosphere, we also expect this class of networks. Numerous previous studies have already discovered power law degree distributions in the blogosphere (Shirky, 2003; Tricas et al., 2003).

For a better understanding, Figure 2.1 illustrates the degree distribution in an example. We have selected the indegrees of the Top 100 German blogs as listed by Technorati in October 2008[3]. The x-axis lists the Top 100 blogs in the order of the ranking, and the y-axis denotes the number of inbound links a blog receives from the rest of the blogosphere, as indexed by Technorati.

[3] The data is taken from the archives of `www.deutscheblogcharts.de/`, which is the last month before Technorati changed its service, and does not list link counts anymore.

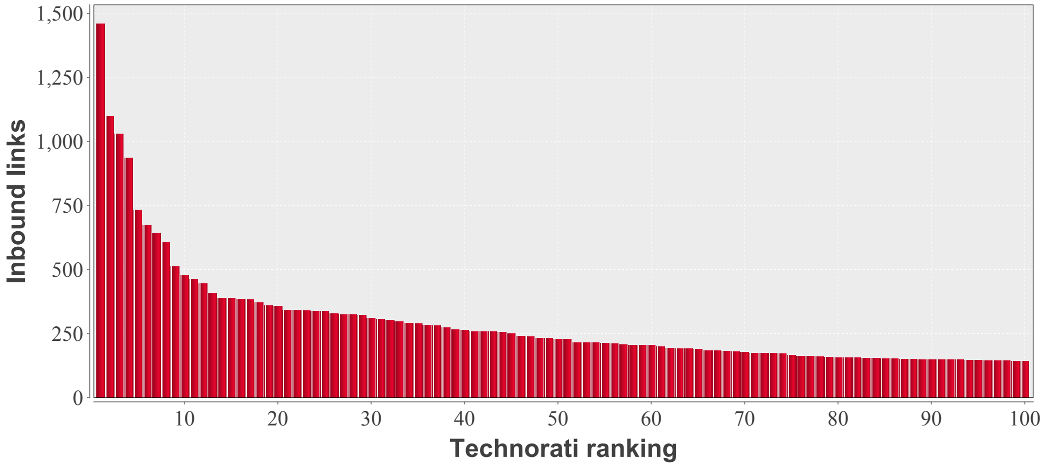


Figure 2.1: Degree distribution of the Top 100 German blogs as listed by Technorati

2.3 Evaluation with Random Networks

Whenever case-studies of social networks are performed, and methods and metrics from SNA are used, the evaluation of the findings is a decisive aspect of the scientific work. One option for such an evaluation is the comparison of the original graph's properties to those of randomly generated graphs with the same degree distribution. Conforming properties can be considered to be trivial, and non-conforming ones indicate a distinctive particular feature or an anomaly of the original graph.

This method has its roots in a paper of Watts & Strogatz (1998), in which they showed that the famous "small-world phenomenon" (see Section 1.2) is a common phenomenon in any graph with a small amount of randomness, and thus a trivial property of real-world networks, not a distinctive one.

This had a lasting effect on social network research, promoting network evaluation by comparing them with random networks. One important finding was the consideration of the degree distribution for these comparisons, as most large real-world networks show a highly heterogeneous power law distribution (compare with Section 2.2.4), opposed to the expected Poisson distribution in trivial random graphs (Erdos & Renyi, 1959). Thus, for a sound analysis of properties it is necessary to sample a random graph with the same degree distribution.

In this section we take a closer look at existing algorithms for random graph generation that enable a solid and reliable evaluation of interesting network properties.

2.3.1 Random Graph Models

Initiated by the random graph model of Erdos & Renyi (1959), the disciplines of mathematics and physics were the first ones to start the study of random graphs and probabilistic random graph models. These studies usually focus on solving the graph with stochastic methods, and investigate global or local graph properties when n is going towards infinity. Bollobas (1985) provides an extensive summary of work in this direction.

The biggest problem with Erdos' random graph in the modelling of social networks is its Poisson degree distribution. As previously mentioned, studies have shown that nearly all real-world networks have a highly heterogeneous degree distribution that follows at least asymptotically a power law in most cases.

These observations and their practical implications lead to new random graph models, which can be parameterised in order to make a given degree distribution fit this model well (Wasserman & Robins, 2005), and even models with prescribed arbitrary degree distributions and additional properties (Newman et al., 2002). These models are very appealing, because they are exactly solvable and hence can give researchers an idea of global and nodal properties of such random graphs in their generalised form.

Following the seminal paper of Watts & Strogatz (1998), practitioners in SNA are usually interested in the comparison of real-world network properties with random graph properties, in order to find uncommon differences. As there is hardly any software support, parameterising these models for a given real-world network, or even calculating metrics of interest, which are beyond those already solved, are highly non-trivial tasks.

This is most probably the reason why most practical network studies still use explorative and descriptive methods for their evaluation, which might be very helpful in the beginning, but is not strictly conclusive in the end. Using instances of randomly generated graphs and comparing the metrics of real-world and random graphs with methods of descriptive statistics is state-of-the-art in practice and can be considered sufficiently conclusive, given a large enough number of samples.

By no means we want to discourage the use of network models, an exact stochastic solution is always the optimum. But recognising that their application requires very good theoretical knowledge, and that there is still a multitude of properties unsolved for these models, we concentrate on the evaluation with randomly generated graphs.

2.3.2 Random Graph Generation

First of all, we have to distinguish two types of random graph generation. The first one, the generation of instances of models, serves more general purposes, for example the empirical evaluation of model properties or model parameter impacts. For most models, these networks can be generated very efficiently according to Batagelj & Brandes (2005), thanks to the exact mathematical properties of their degree distributions.

The second type of generation requires a given arbitrary degree distribution of a real-world network to be exactly realised. As mentioned before, this is useful for the evaluation of concrete real-world networks, which is our focus in this thesis.

Principal Evaluation Procedure

In principal, you sample a sufficiently large number of random networks, i. e., 30 or more are usually recommended for significance, and then determine the statistics of the property of interest. For a simple numeric network metric, this results in an *average value* ± *standard deviation*. You can then see the factor z, which denotes how many times the standard deviation your real-world network differs from the average. In consequence, a $z \leq 1$ indicates an average network structure, and a $z \geq 2$ indicates a significantly uncommon structure in this specific aspect.

This requires a different approach to network generation, which we will look at in the following sections. Milo et al. (2003) give a very good overview of this field, and we adhere to their terminology and method descriptions.

2.3.3 The Configuration Model

The simplest approach is the *configuration model*, which is well summarised by Newman (2003, Section IV.B). It is the set of all graphs with a given degree sequence. The generation algorithm is fairly easy. It starts by adding *stubs* for the required endpoints of a node, according to the degree distribution. It then chooses pairs of stubs uniformly at random and connects them, until all stubs were replaced by edge endpoints. This algorithm is the default generating algorithm in most network libraries that offer generation by degree sequence, e.g. NetworkX[4] for Python, while other packages do not even include this one, e.g. JUNG[5] for Java.

[4] `http://networkx.lanl.gov/`

[5] `http://jung.sourceforge.net/`

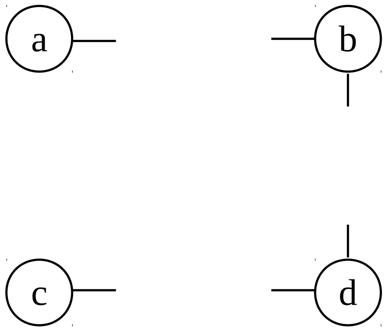


Figure 2.2: Example network with stubs for edge generation

However, it has one serious drawback for practical use cases. It is not restricted to simple graphs, it includes graphs with loops and parallel edges. In real-world networks, these are often forbidden properties, and hence an evaluation with this model is not fully accurate anymore. Figure 2.2 shows an undirected graph with a given degree sequence, for which we want to create edges by random. With the configuration model, any two stubs are chosen by random, which allows eight different connections in the first step. If you are however restricted to simple graphs, there are only five legal connections left, because (b,b) and (d,d) would directly violate the simplicity criteria, while (a,c) would inevitably lead to a violation in the next steps by producing loops or two parallel edges (b,d).

This vulnerability decreases with higher n, but when using the configuration model for evaluations, you nevertheless will have to discard loops and parallel edges afterwards, at the price of a more or less different degree sequence than initially prescribed. Viger & Latapy (2005) have empirically demonstrated that this can introduce a noticeable bias in network properties.

Another solution suggests to repeat the algorithm until it succeeds without loops and parallel edges, which is however extremely unprobable in real-world networks. A usable algorithm based on such a modification is evaluated by Milo et al. (2003) under the name *matching algorithm*. Creations of parallel edges do not stop the generation, but are just rejected. This increases the chances to succeed in generating a simple graph. However, this algorithm has a noticeable bias in the uniformness of its samples. On the other hand, it is empirically shown that the consequences appear to be negligible. Still they suggest to use a Markov Chain Monte Carlo (MCMC) algorithm instead.

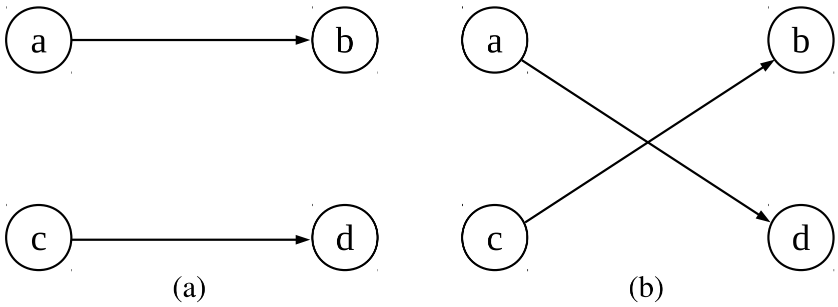


Figure 2.3: Example of a legal edge swap from (a) the initial situation to (b) the new situation, that hence changes the network structure

2.3.4 Markov Chain Monte Carlo (MCMC) Algorithms

As claimed by Viger & Latapy (2005):

> Although is has been widely investigated, it is still an open problem to directly generate such a random graph, or even to enumerate them in polynomial time [...]

This enumeration has been accomplished by Snijders (1991), but because of the resulting exponential runtime complexity, most researchers turned towards *Monte Carlo* methods for random graph generation.

According to Milo et al. (2003), the fastest of these algorithms are MCMC algorithms. They have the additional benefit to be extendable to guarantee the creation of *connected* simple graphs.

These algorithms do not directly create random graphs, but works with *edge swaps*. In an edge swap, we randomly pick two edges and swap them, if the new situation adheres to the simple graph requirements, and also to connectivity requirements, if desired. Figure 2.3 provides a minimal example, in which the edges (a,b) and (c,d) are selected and swapped.

The algorithm proceeds in the following three steps.

1. Generate a simple graph realising the prescribed degree sequence, or use an existing real-world graph if available.

2. Connect the graph with edge swaps, if this is desired, and if it is not yet connected.

3. Perform a series of edge swaps, until the graph appears to be a random one. This is called *shuffling* the graph.

Viger & Latapy (2005) validate empirically that $O(m)$ edge swaps are sufficient for nearly perfect uniform sampling, but a formal proof is still missing. Milo et al. (2003) estimate the constant factor of this bound to be around 100. Furthermore, they describe that a naive implementation has a runtime complexity within $O(m^2)$. This naive algorithm is called *switching algorithm*. Viger & Latapy (2005) propose a speed-up to a runtime complexity of $O(m \cdot \log m)$ for undirected graphs, based on a corrolar that also has the issue of a missing proof, but is backed up with a thorough empirical validation.

In the summary of all discussed aspects, we decide to go with the enhanced MCMC algorithm that guarantees connected random networks for our evaluation. Its time complexity is acceptable for large networks (see Section 2.2.3).

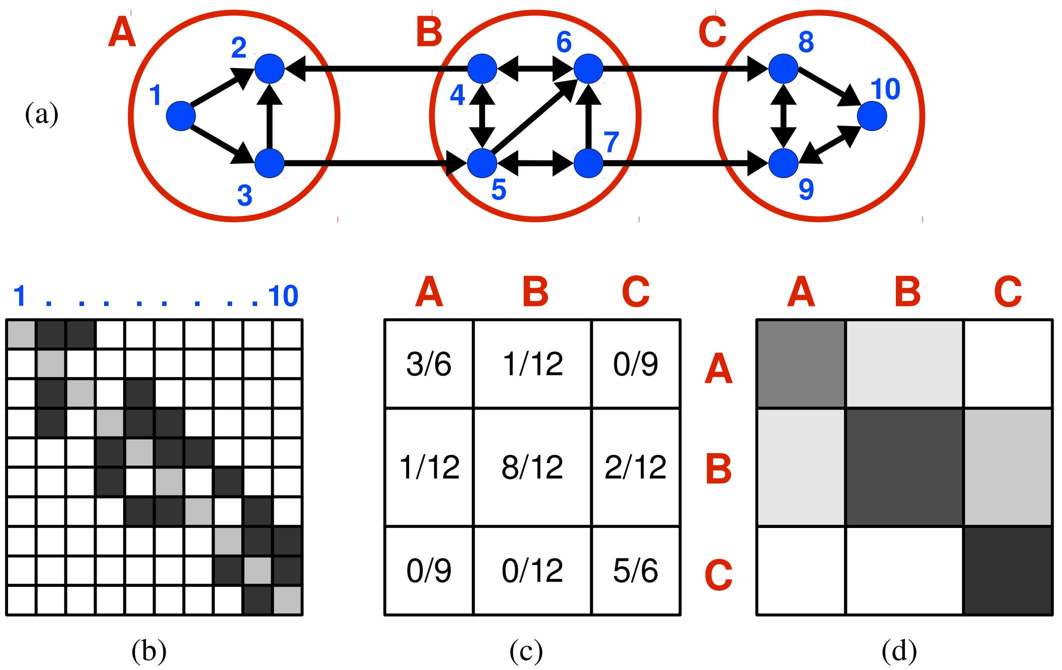
2.4 Visual Evaluation of Partitionings

In the course of this thesis, we will often deal with partitionings of networks, either into disjoint or into nested groups of nodes. For a good impression of the resulting structure of such a partitioning, which is very hard to communicate with numbers only, we propose a new visualisation based on abstracted adjacency matrices.

2.4.1 Group Adjacency Matrices (GRAMs)

Using the directed example network from Figure 2.4a, we first construct the adjacency matrix as shown in Figure 2.4b, where each black entry represents a “1”. The diagonal entries cannot have any value, since we assume simple graphs only. Next, let us assume a partitioning into three groups A, B and C as depicted in Figure 2.4a. Having the nodes grouped by partition in the adjacency matrix, this allows us to zoom out of individual entries, and to focus on the areas of the partitions instead, which form rectangles. For each of these areas, the local standard density value can be computed as shown in Figure 2.4c. Finally, these local density values can be mapped to greyscale values, which are used to paint the rectangle with, as visible in Figure 2.4d.

In such a plot, which we call a GRAM, one can easily spot the structural relations among all partitions for a given partitioning of the network. Looking at Figure 2.4d only, it is visible from the diagonal that all the partitions are very cohesive, and



	A	B	C
A	3/6	1/12	0/9
B	1/12	8/12	2/12
C	0/9	0/12	5/6

Figure 2.4: Example of (a) a network with (b) its adjacency matrix, (c) the densities per section and (d) the resulting grey values in the GRAM

that there are only a few links crossing partition borders. One can also see in the middle-right field that there are links from B to C, but no links from C to B, as the bottom-center field is all white. In consequence, the chosen partitioning looks like a suitable clustering of the graph.

It is important to note that this is a powerful method to judge a given partitioning, but it will not help much in finding a partitioning with the desired characteristics.

2.4.2 Scaling of Density Saturations

As most large real-world networks are typically sparse networks with very low densities, a linear mapping of density to greyscale saturation would not produce anything visible. In order to make the existing relative differences visible, we introduce two modifications.

Function 2.1 presents how to determine a greyscale saturation between 0 (white) and 1 (black) for a given density value d. d_{max} is used to normalise the grey values to the highest occuring local density of all partitions, and α controls the shift of

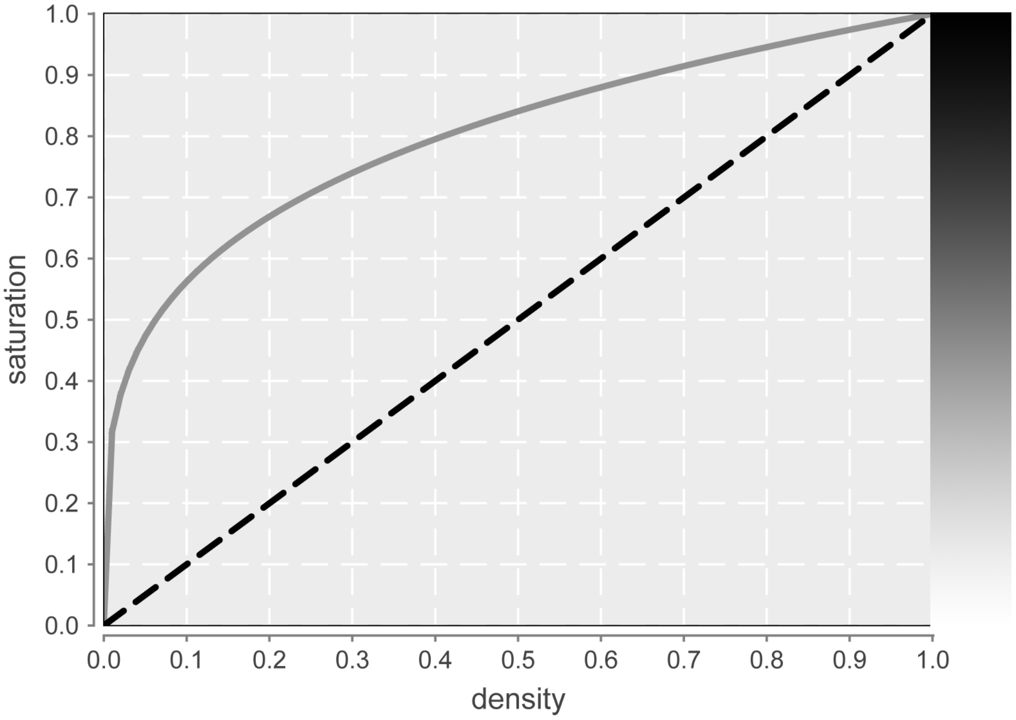


Figure 2.5: Greyscale saturation function for partition densities with $\alpha = 0.25$ (dashed line shows $\alpha = 1$)

accuracy/resolution to lower density values, as illustrated in Figure 2.5.

$$greyscale(d) = \frac{d^{\alpha}}{d_{max}}, 0 < \alpha \leq 1 \tag{2.1}$$

The best usage for these parameters has to be determined dynamically in each case. The normalisation sometimes provides only little effect, since small partitions in large networks may have very high local density values relative to the larger partitions. However, this can usually be compensated quite well with a lower exponent α.

In the following chapters, whenever GRAMs are used, we will select suitable parameters to optimally visualise the partitions' relations, but we will not enumerate the used parameters for every single matrix, since direct absolute comparions between different GRAMs from differently sized networks are not of interest for us.

CHAPTER 3

Sampling Blogroll Networks

For the intended analysis of the general authority of blogs, as defined in our first research question, we need suitable datasets. The process of collecting these datasets is described in this chapter, preceded by the justifications for the main decisions in that process.

3.1 Blogs and Blogrolls

The blogroll of a blog is an explicit list of recommendations of other blogs by the author. We choose to use these links instead of references from articles or comments for a number of reasons.

From a social network point of view, an explicit recommendation link by the blog author(s) is much more expressive and better to interpret than an arbitrary reference, whose semantics is unknown without a reliable link analysis. Additionally, there are no weights and no timeframes to be considered. All entries are equal, and if an author decides not to recommend a blog anymore, he should remove the corresponding link from his blogroll. Of course, in certain cases the blogroll might be outdated, but we expect this to be rather an exception than the rule in a popular blog.

Nevertheless there are some doubts about the expressiveness of blogroll links in the blogging community to be aware of. Some people argue that bloggers might use their blogroll more for identity management than for real recommendations, i. e., they choose the links in order to communicate a desired impression they want others to have about them. Psychologically this is neither new nor implausible, but we decide to stick to the objective facts here, keeping this possibility in mind.

3.2 Snowball Sampling

For the collection of a representative share of the most authoritative blogs on the Internet, we use a variant of *snowball sampling* (Doreian & Woodard, 1992). We start with a seed of the most authoritative blogs and iteratively include new authoritative blogs by examining the outbound links of the actual set. According to the A-List characteristics described in Section 1.3, frequently referenced blogs should be part of the A-List as well.

3.2.1 Blog Seeds

In order to find a large set of popular blogs, we need a starting point, i. e., a seed list of some highly popular blogs. First of all, we decide to sample six different datasets according to their language. As mentioned in Section 1.3, blogs of different languages are small blogospheres on their own, and thus we will be able to cross-check our results between these datasets. We have chosen six European languages, English (en), German (de), French (fr), Spanish (es), Italian (it) and Portuguese (pt), which we can all understand, so that the interpretation of the results is assured.

We start with Top 100 lists from existing ranking services, ignoring their positions in these lists. For English blogs, we use the market leader Technorati. For German blogs, we use the German Blogcharts[1], a Technorati-based list. For Spanish, French, Italian and Portuguese blogs, Alianzo[2] provides good lists by language, which we use for these cases.

3.2.2 Crawling Blogroll Links

We implemented a set of scripts to find the entries from the individual blogrolls, if present. We encountered three pitfalls in this task. First, we had to develop a sufficiently good heuristic for locating the blogroll entries, as their inclusion on the blog page is not standardised in a way we could rely on.

The second pitfall is the existence of multiple Uniform Resource Locators (URLs) for one blog. We check every single blogroll entry with a Hypertext Transfer Protocol (HTTP) request in order to not insert blog links to synonymous or redirected URLs another time into our database. This would cause a split of one blog into two separate nodes and thus distort our network and our results. This is a common

[1] http://www.deutscheblogcharts.de/

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

problem, e. g., Technorati often ranks a blog multiple times, which leads to biased results in consequence.

The last pitfall is the reachability of a blog. Blogs that are not reachable during our crawl, either because of network timeouts or because they prohibit spiders, are ignored with respect to their own blogroll links, but remain in the dataset and can be recommended by other blogs of course.

3.2.3 Extending the Datasets

Starting from the seeds and their blogroll links, we iteratively include new blogs. The most often referenced URLs are checked and included, if they are indeed blogs written in the matching language.

To decide whether an URL hosts a blog, we check it via the Technorati Application Programming Interface (API).[3] This works very well for popular blogs, as they are usually indexed. Small blogs from the long tail might remain undetected though. This is less of a problem for our goals, as only popular blogs with a certain number of inbound links are candidates for inclusion anyway.

The language is detected by counting stop words in the blog articles. The full texts of the recent articles of a blog are easily accessible via its feed. Having complete stop word lists in different langauges, a simple counting and majority voting reveals the most probable language of the text. We used an according implementation from Perl's CPAN module `Text::Language::Guess` [4]. Thanks to the usually rich textual content of blog articles, this works very reliably.

The extension process is iteratively repeated, and the dataset thus grows in size. Since we have to stop at some point, we decided for a very pragmatic criterion here. Once the number of the most frequently referenced candidates with exactly the same number of recommendations from the dataset exceeds 500, we stop the extension process. The reason for this is, that the Technorati API, which we use for the blog detection, limits us to 500 queries per day. So this criterion simply guarantees us to be able to finish at least one extension step per day, and practical results will show that it works out very well for the different datasets.

[3] Unfortunately, the Technorati API has been closed in March 2010, which currently prevents a repetition of this method.

[4] `http://search.cpan.org/~mschilli/Text-Language-Guess-0.02/`

	en	**es**	**de**	**fr**	**it**	**pt**
blogs	100	100	100	100	49	100
links	183	376	289	181	52	58
density (in %)	1.85	3.80	2.92	1.83	2.21	0.59
average degree	3.7	7.5	5.7	3.6	2.1	1.2
isolated blogs	31	10	11	18	25	50

Table 3.1: Overview and comparison of the seed networks

3.3 Resulting Datasets

The data for the English, German, French and Spanish blogs has been collected throughout September to December 2008, and the data for the Italian and Portuguese blogs throughout August to October 2009. All resulting networks are available as Pajek files on the author's homepage[5].

Table 3.1 lists the relevant interconnectivity measures of the seed lists, i. e., the number of links, the density and the number of isolated blogs with respect to weak connectivity. Notably, all metrics indicate a good interconnection in the language-specific seeds, with the exception of the Italian and Portuguese ones. The seed lists with 49 and 100 blogs were too small in these cases, but we will see later that nevertheless these seeds were sufficient to deliver good datasets, after having applied our iterative extension.

Table 3.2 lists our final datasets after the iterative extensions. As density is hard to compare in networks of different sizes, we additionally list the average total degrees of the sets. Noticeably, we end up with very well interconnected sets of blogs. As expected in blog networks, the degree distributions for both, incoming and outgoing edges, resemble power laws in all six networks (Shirky, 2003).

Due to the special nature of our extension process, we also list the minimum indegree a candidate URL must have had in order to be checked and eventually included into the dataset, according to the snowball sampling procedure described above. This value will be of importance later on, as it is a decisive value for the analyses in the next chapter.

[5] `http://www.dfki.uni-kl.de/~obradovic/data/`

	en	es	de	fr	it	pt
blogs	8,401	5,373	1,837	3,402	2,773	3,776
links	452,234	104,241	24,065	90,546	75,421	93,770
density (in %)	0.64	0.36	0.71	0.78	0.98	0.66
avgerage degree	107.7	38.8	26.2	53.2	54.4	49.7
isolated blogs	3	0	2	7	11	25
min. indegree	12	8	5	8	7	9

Table 3.2: Overview and comparison of the extended networks

3.4 The Multi-Language Network

We initially formulated the hypothesis that blogs in different languages form local blogospheres on their own. In consequence, we have sampled six different datasets. In this section, we try to validate this hypothesis against the actual data.

3.4.1 Merging the Language Networks

Since we collected all blogroll entries from all the blogs we included in the six local datasets, we also have access to those links traversing language borders, e. g., the recommendation of an Italian blog in the blogroll of a German blog.

Going through these lists, we explicitly connect all blogs with such links across the six different datasets, and end up with a new network that contains all 25,562 blogs, the 840,277 links of the six local datasets, and 10,813 newly established links between the local datasets, adding up to 851,090 links in total in this new network, which we call the *multi-language network* from now on.

Table 3.3 lists the links within the local datasets on the diagonal, and the links from each local dataset to all other ones, with the rows indicating the source of the links, and the columns indicating the target.

It is apparent that the partitioning into languages provides an extremely good clustering for this network, as assumed in the beginning of this chapter. This means that it indeed makes more sense to analyse the isolated language datasets against the A-List structure in the following analyses.

Concerning the relations among the local datasets, such plain numbers are difficult to interpret without reference to the individual dataset sizes and densities. The visualisation with GRAMs, presented in Section 2.4, seems more suitable here.

	en	**es**	**pt**	**fr**	**it**	**de**
en	452,234	184	100	73	657	190
es	2,195	104,241	582	771	65	40
pt	2,449	787	93,770	285	49	43
fr	550	158	56	90,546	24	74
it	1,142	71	221	50	75,421	14
de	1,228	41	1	75	20	24,065

Table 3.3: Links between the local datasets, from row to column

3.4.2 Visualisation with a GRAM

In order to illustrate the interpretation of a partitioning, we have plotted the language partitions of the multi-language network in Figure 3.1. We have chosen $\alpha = 0.2$ for the plot and normalised the greyscales to the highest occuring partition density $d_{max} = 0.022$.

First of all, the cohesion inside the language datasets is also visually very apparent. Comparing the intensities of linking between the different languages, we observe a few interesting facts.

The English blogs receive the most links from the rest of the network, but do not link back as much. The French blogs are pointing only little to other languages, affirming some prejudices about the notorious “francophony”. However, it is the German blogs linking to the Portuguese-speaking community that is the least intensive one in the network, represented by the most lightly coloured field.

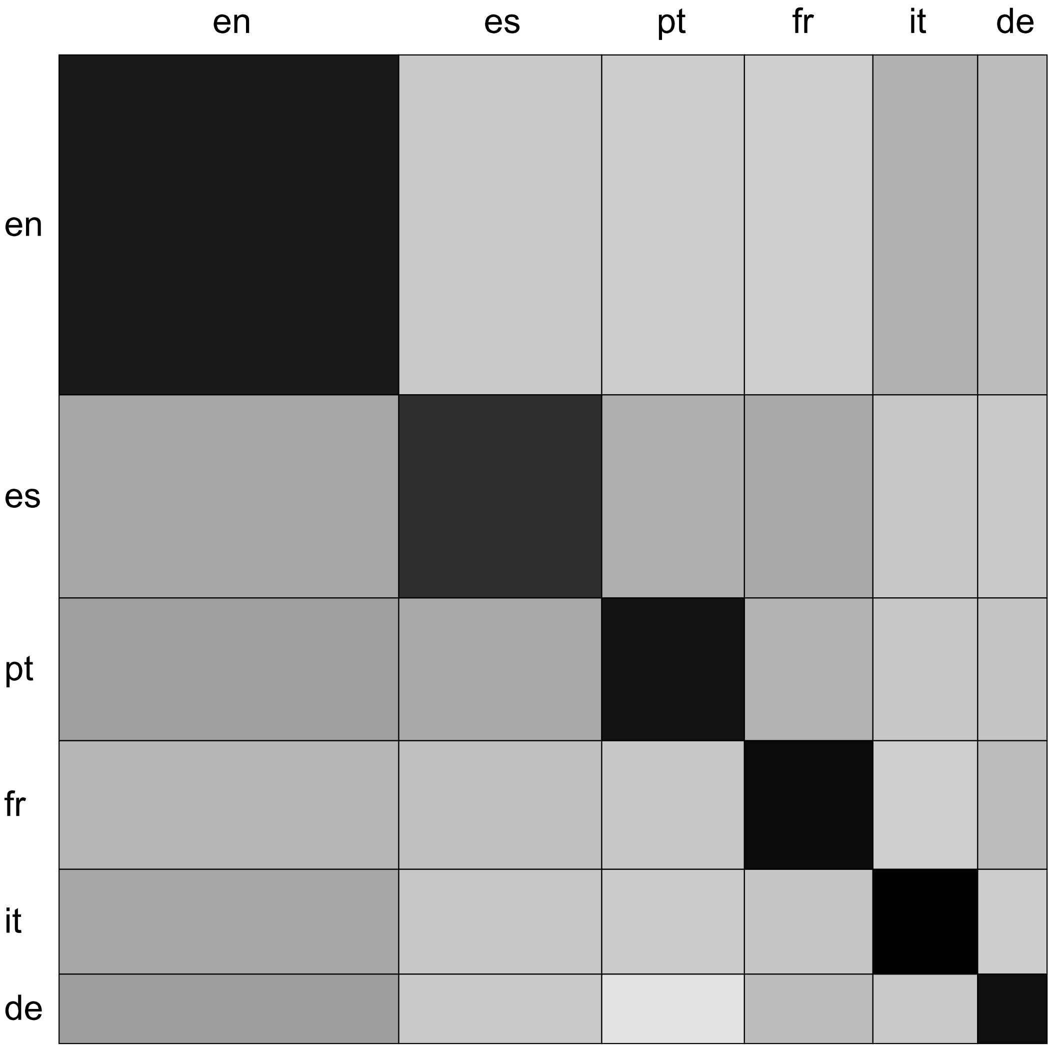


Figure 3.1: GRAM of the multi-language network grouped by language

CHAPTER 4

Identifying A-List Blogs

This chapter analyses the datasets from the previous chapter with the goal to reliably identify the group of A-List blogs as defined in Section 1.4.

4.1 The Core/Periphery Model

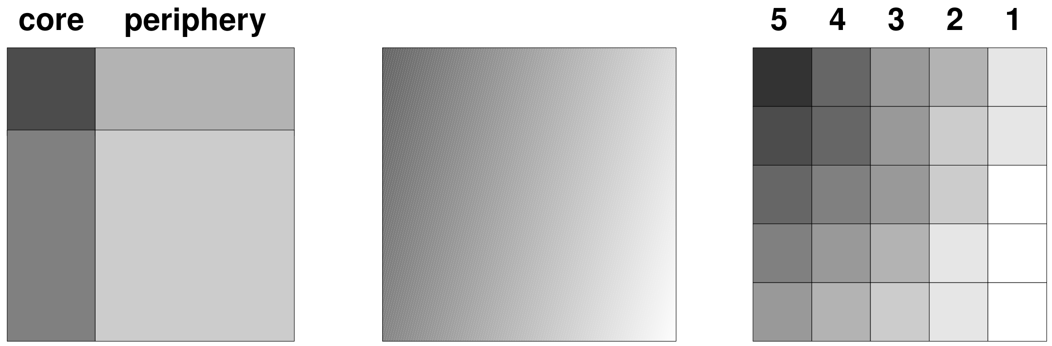
Borgatti & Everett (1999) present a model for networks in which a heterogeneous distribution of authority is assumed. Their approach comes very close to the theory of the A-List characteristics.

The initial idea is to partition a directed network into two groups. An authoritative one called “the core”, and a peripheral one. The core should receive many links from the periphery, and link more to other core members than to the periphery. On the other side, the periphery should link mostly to nodes in the core and only little to other peripheral nodes.

They present a goodness-of-fit measure for a given partitioning, and propose a genetic algorithm to find the most suitable partitioning by re-ordering the nodes in the adjacency matrix. However, as there are $n!$ possibilities to order a network with n nodes, they only give examples for networks with a few dozens of nodes.

Seeing that often there is no sharp border between the core and the periphery, but a smooth transition, they also suggest an extension with a continuous model that only considers the ordering of the nodes, without a partitioning.

A GRAM plot of a typical core/periphery structure for a network is shown in Figure 4.1a. An abstracted view of an adjacency matrix for a good fit of the continuous model is given in Figure 4.1b.



(a) (b) (c)

Figure 4.1: Examples of (a) a GRAM for a typical core/periphery partitioning, (b) an abstracted adjacency matrix for the continuous model and (c) an idealised result of an in-core collapse sequence's partitioning

However, a large drawback is the use of the adjacency matrix and a genetic algorithm for finding a re-arrangement with a good fit out of the $m!$ possibilities. This makes it very expensive to apply this model to large graphs, or even impossible if the adjacency matrix does not fit into memory.

While the model is what we are looking for, as explained in Section 1.4, the computational solution is not applicable in our cases with relatively large networks, as outlined in Section 2.2. Due to this conflict, we are looking for an alternative approach that computes a similar model with a scalable algorithm.

4.2 The Concept of a k-Core

The intuitive notion of a *k-core* has been initially formalised by Seidman (1983). He defines k-cores in an undirected network as subgraphs that contain only nodes with a minimum degree of k.

Thus each node has a maximum k, so that it is part of a k-core, but not part of a $(k+1)$-core. All nodes with the same maximum k together form the *k-frontier*. This results in a Core Collapse Sequence (CCS) of the network, which is the sequence of the nested k-cores. A corresponding algorithm can be implemented in very good polynomial runtime complexity, as we will show in Section 4.3. However, this model has not yet been properly transferred to directed graphs, as we would need for the analysis of our datasets.

Doreian & Woodard (1994) provide a good comparison of the core model with other measures of cohesion like cliques, *n*-cliques, *n*-clans, *k*-plexes or density (see Doreian & Woodard, 1994, p. 269f). In summary, the main advantage of *k*-cores for the identification of cohesive subgroups is the fact that it partitions the graph in a discrete and iterative manner, where results are relatively easy to interpret, opposed to long overlapping lists of cliques and the like. Additionally, blogroll links have no real meaning for transitivity, which favours the *k*-core model for our approach, opposed to *k*-cliques and the like, which are based on distances.

Core Models for Directed Graphs

Seidman's definition of *k*-cores can be intuitively extended to directed graphs, which is what we need to do in order to apply it in our blogroll networks. Adhering to the terminology for directed graphs and following some initial ideas from Doreian & Woodard (1994), we see five options to define a *k*-core in a directed network:

weak k-core: when each node has at least *k* links of any kind to the rest of the core

strong k-core: when each node has at least *k* strong connections to the rest of the core, i. e., reciprocal links

k-in-core: when each node has at least *k* incoming links from the rest of the core

k-out-core: when each node has at least *k* outgoing links to the rest of the core

balanced k-core: when each node has at least *k* incoming and *k* outgoing links to the rest of the core

Options number one and four are uninteresting for us, because they allow blogs that have no inbound links at all to be part of the core. Thus, anyone could make himself part of such a core easily, without any external legitimation. As blogs do not have to maintain a blogroll in order to be important, they could have been temporarily unreachable during our data acquisition, or have not been covered by our blogroll detection heuristics, requiring outgoing links does not make sense here. Consequently, options number two and five are also not applicable in our case.

When remembering the characteristics of an A-List set, it is obvious that incoming links are the decisive element, and that we consequently will focus on option number three, namely *k-in-cores*. For each core member, it assures a certain authority by the rest of the core. This is consistent to the requirements of Borgatti and Everett's core/periphery model for directed graphs presented in the previous section.

4.3 The In-Core Algorithm

We present a possible procedure for determining the in-core values of all nodes in a graph, and discuss the runtime complexity afterwards.

Starting with $k = 1$, all nodes marked as non-collapsed, and their initial indegree stored in the number of non-collapsed predecessors, we iteratively repeat the following steps.

1. for each non-collapsed node, check if it has at least k non-collapsed predecessors; if not, let it collapse with an in-core value of $k-1$

2. for each node v collapsed in this iteration, for all nodes in $succ(v)$ decrement the number of non-collapsed predecessors by 1 and recursively repeat the check of the previous step

3. if there were no more collapses in the last step, either terminate the algorithm in case that all nodes have collapsed, or proceed to the next iteration with $k = k+1$

First of all we take a look at the maximum possible value for k in a graph with m edges. To form a k-in-core with n_k nodes, we need at least $n_k \cdot k$ directed edges, with $n_k > k$ when operating on a simple graph. Due to this last condition, in order to maximise k to k_{max}, we will use a maximally connected component with $k_{max}+1$ nodes. In consequence, with a given number of m edges, we can reach at most $k_{max} = \lfloor \sqrt{m} \rfloor - 1$, which is thus the maximum number of iterations for the algorithm described above.

Counting the indegrees in the initialisation costs m. In each iteration, step 1 requires to check at most n nodes, with constant cost for each node. This results in costs of at most n per iteration. Independently from the loop, step 2 is executed exactly n times throughout the algorithm, as each node collapses once. With m successors in total to be checked again, and each check being done with constant cost, step 2 costs at most m.

Step 3 can be performed during step 1 of the next iteration, so the total maximum cost for executing the algorithm is within $O(m + \sqrt{m} \cdot n + m)$. When assuming an equal order of magnitude for nodes and edges in large graphs (see Section 2.2.2), i. e., $n \approx m$, this results in a runtime complexity of $O(m^{1.5})$ for large real-world graphs.

According to the general requirements for scalable algorithms, as mentioned in Section 2.2.3, this algorithm is scalable and applicable to large networks thanks to a subquadratic runtime behaviour. As the upper bound is mainly determined by k_{max}, we can expect the algorithm to run even closer to linear time in real networks, as k_{max} is typically not that close to the theoretical maximum.

As the only addition to the network data structure is the in-core value for each node, the storage complexity remains linear within $O(n+m)$.

Batagelj & Zaversnik (2002) have proved the working of such an algorithm for core decomposition, and also came up with a more complicated algorithm that achieves a linear runtime complexity in $O(m)$ (Batagelj & Zaversnik, 2003).

4.4 Evaluation

In this section we evaluate the application of the in-core algorithm to our six datasets from Chapter 3. We use different perspectives in order to achieve a maximally reliable conclusion. This includes empirical results by a comparison with random networks, cross-validation among the similar datasets of different languages, and a comparison to the core-periphery model.

4.4.1 Comparison to Random Networks

In a first step, we compare the CCS of each dataset with the one from an average randomly generated network[1] that has exactly the same degree distribution. This means, that for every node from the original network, there exists a node in the random network with the same indegree and out-degree. The random networks are generated with an MCMC algorithm as described in Section 2.3.

The plots in the Figures 4.2 to 4.7 illustrate the in-core structure of each dataset. For each k on the x-axis, the y-axis indicates the number of blogs that are part of this k-in-core. Each plot contains the sizes of the k-in-cores of the original blog dataset, marked by filled blue square points that are joined by straight lines, as well as the sizes of the k-in-cores of the random network, marked by red circles that are joined by dotted lines.

In all six cases, we can clearly see that the original datasets tend to contain in-cores with a higher k than expected from the network degree distribution. This means that, beyond the preferential attachment model (Barabasi & Albert, 1999) these blog datasets have an unexpected tendency towards core-centralisation. This highly conforms to the second A-List characteristic as defined in Section 1.4.

[1] selected from 30 samples based on their CCSs, while differences were only marginal in all cases

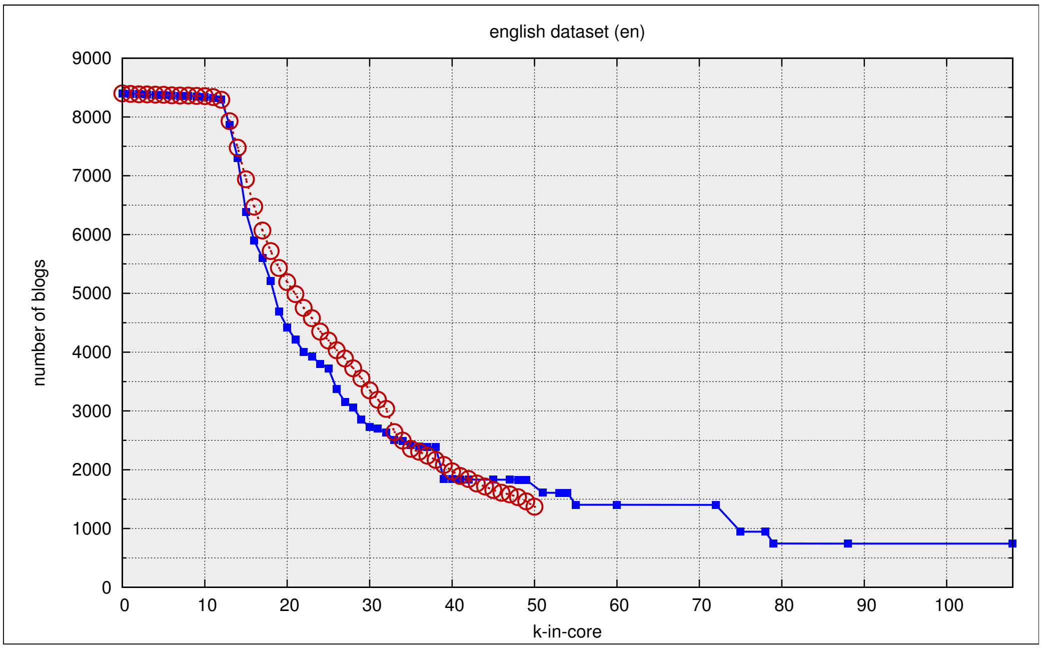


Figure 4.2: In-CCS for the real and the random English network

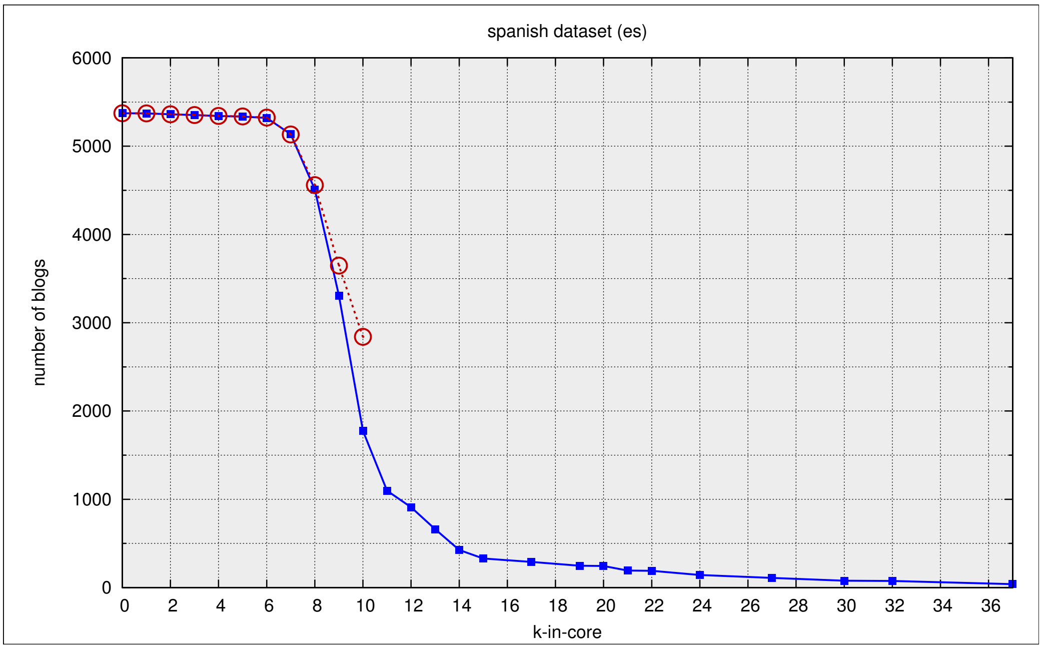


Figure 4.3: In-CCS for the real and the random Spanish network

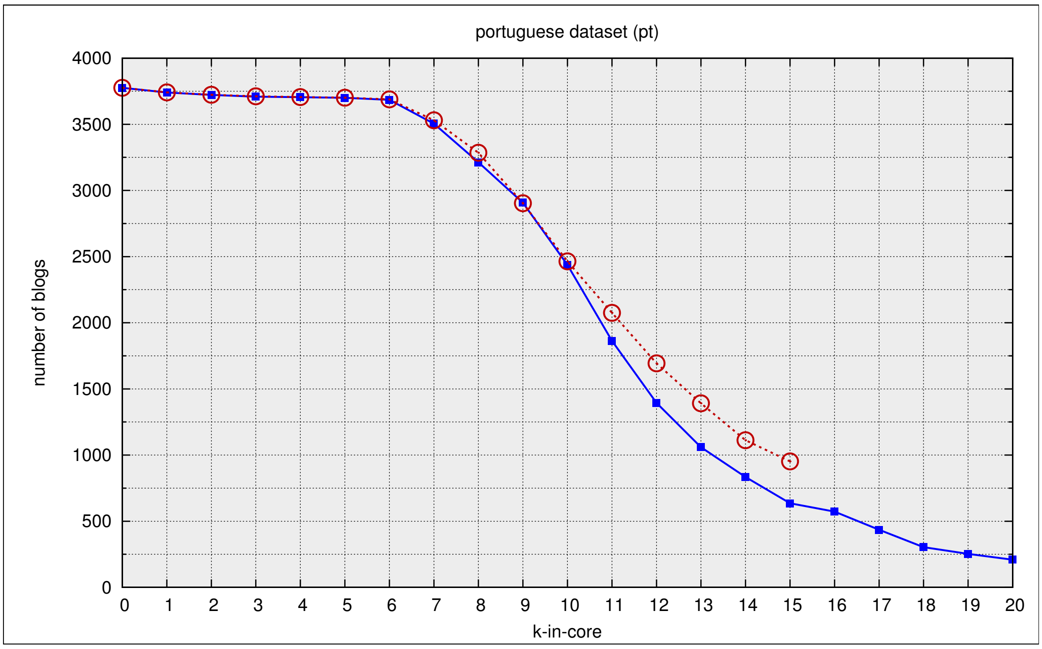


Figure 4.4: In-CCS for the real and the random Portuguese network

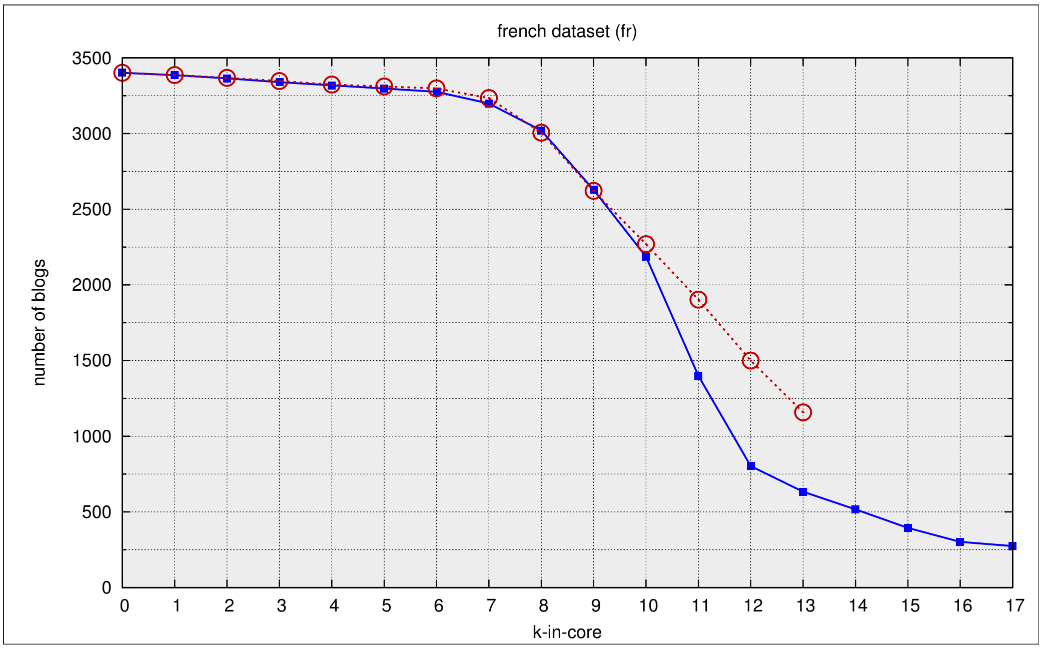


Figure 4.5: In-CCS for the Real and rhe Random French network

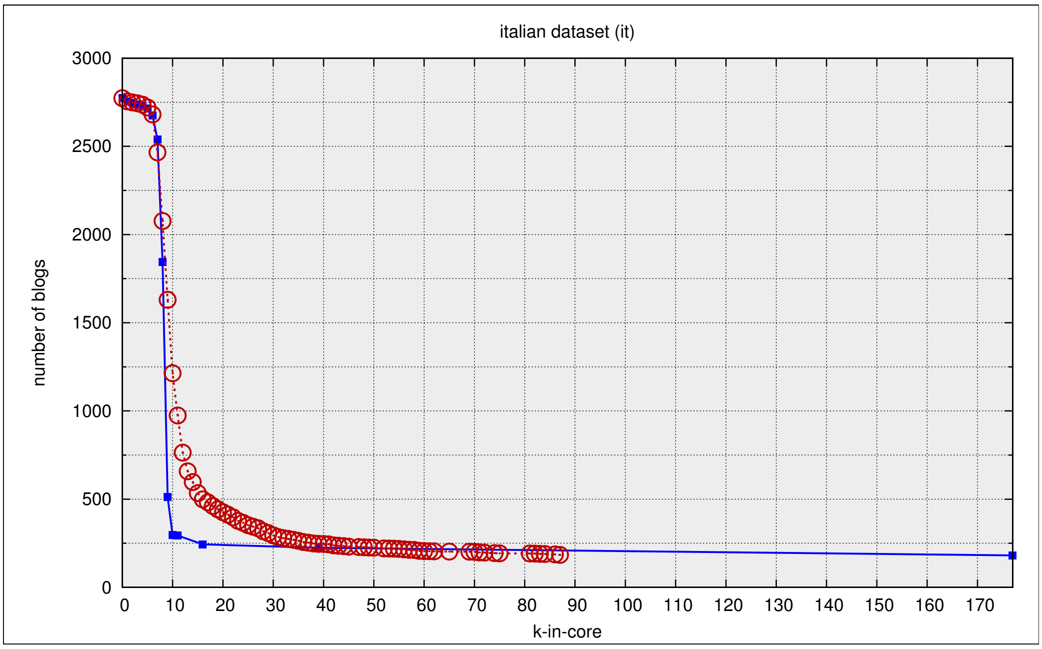


Figure 4.6: In-CCS for the real and the random Italian network

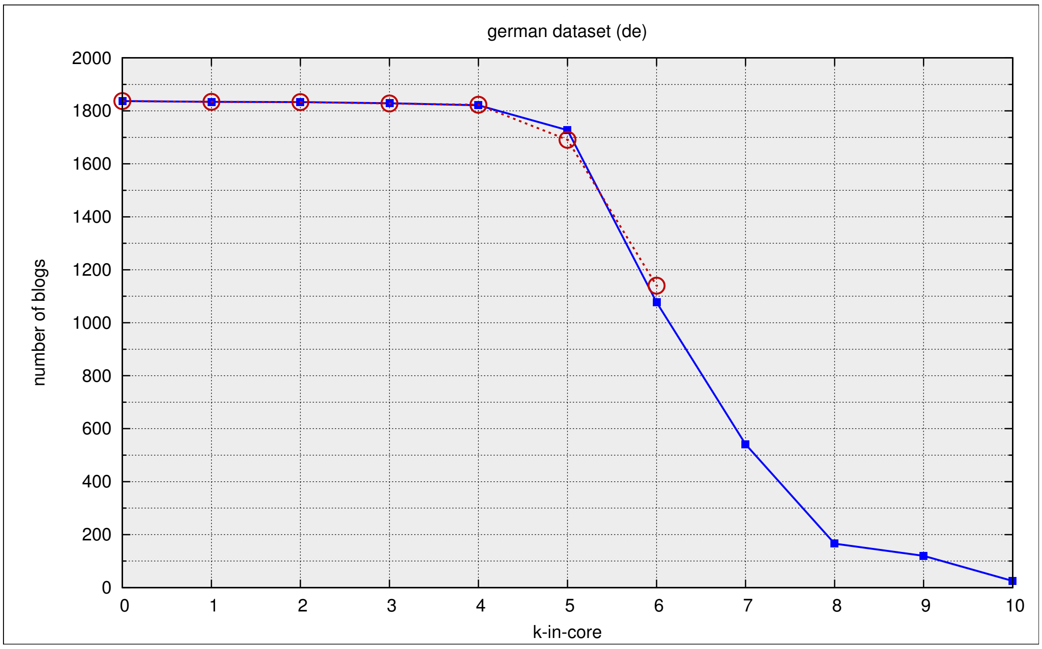


Figure 4.7: In-CCS for the real and the random German network

4.4.2 Comparing the Datasets

In a second step, we compare the results of the different datasets with each other, and thus exploit the fact that we have six similarly structured networks, which can disguise anomalies in a network that do not occur in the other ones.

When looking through the plots, one will immediately notice that the tendency towards this core-centralisation is different among the datasets. For the random networks, there is a correlation between the average degree and the curve of the expected k-in-cores. The lower the average degree, the steeper the curve falls, i. e., the less core-centralisation is normally expected, and thus, the resulting cores of the German blogs have to be judged differently than those of the English ones.

Furthermore, we notice that the German and the Spanish blogs contain a very small core at their highest k, i. e., a 10-in-core of 25 German blogs and a 37-in-core of 39 Spanish blogs, a phenomenon that does not appear in the other four datasets. A survey of the blogs in these small cores reveals two interesting explanations. The 25 German blogs all deal with cooking and recipes, and are well-interconnected. The 39 Spanish blogs are all run by the commercial blog network BlogsFarm[2], which runs about 50 blogs that are nearly completely connected. The same explanation applies to the rest of the 78 blogs that form the Spanish 28-in-core, these are run by the commercial blog network WeblogsSL[3], which maintains about 25 blogs. The arising question in both cases is, whether these blogs are only popular among themselves, due to commercial interests, or if they are also fulfilling the most important A-List characteristic, namely to be massively linked by other blogs from outside the core. This question cannot be answered by core-analysis, but needs to be examined further, what we will engage in the next section.

Another thing to notice is the much higher than expected maximum k in the Spanish, the Italian and the English datasets, which is very different from what is observed in the German, the Portuguese and the French ones. This is an indicator for a large, well-interconnected group beyond the core-centralisation as emerged by the A-List phenomenon, according to our understanding. This issue has already been partially clarified for the Spanish blogs, but in the English dataset, we find 744 blogs that form a 108-in-core, and in the smaller Italian dataset we even find a 177-in-core of 181 blogs.

Despite the size of the English dataset, this number appears too high for a sane community, and indeed we have found an interesting explanation. Our first suspicion,

[2] `http://blogsfarm.com/about/`

[3] `http://www.weblogssl.com/quienes-somos`

to have encountered a circle of spam blogs (splogs) did not hold. Instead, this core is composed of about 150 blogs that all include the “Blogging Chicks Blogroll”[4], a so-called *collaborative blogroll* with these 744 blogs, which aims to “take over the Internet, one blog at a time”. This is a unique phenomenon in the English dataset, which prohibits a reliable A-List detection with in-core-analysis only.

For the Italian dataset, the explanation is the same as for the Spanish one, albeit on a significantly larger scale. The highest in-core is formed by blogs from the commercial blog network Blogosfere[5], which runs roughly 200 blogs on different topics. Here again, the same question of general popularity has to be examined.

We also notice a high dominance of one single blog-engine provider in the French dataset, which is a unique phenomenon as well. From the 274 blogs in the French 17-in-core, 89% are hosted by *canalblog.com*, opposed to 68% in the whole dataset of 3,402 blogs. A survey reveals no signs for a systematic favorisation between these blogs, so we regard it as a purely cultural phenomenon and consider the French blog dataset to be free of anomalies. The same holds true for the Portuguese dataset, which also seems to be free of anomalies beyond the expected core centralisation phenomenon. Consequently, these two datasets will serve as references for a sane manifestation of the core centralisation phenomenon for A-List detection.

4.4.3 Comparison with the Core/Periphery Model

In a third step, we validate the approach by comparing it with Borgatti and Everett’s core/periphery model presented in Section 4.1. In the case of a directed network, the variation of their “asymmetric model” is the one relevant to us.

Their example, citations among 20 scientific journals, is comparable to our problem in its goal to identify a core/periphery structure. There indeed emerges something similar to an A-List, namely a subset of journals that fulfills the three A-List characteristics reasonably well.

With the in-core analysis, we detect a 4-in-core that contains 6 journals. This is one more than identified by them as “the core” (see Borgatti & Everett, 1999, p. 385). The journal in question, “ASW”, is included in our 4-in-core, because it is referenced by four other journals from that core. This is a strong argument for a certain authority, according to the second A-List characteristic, since it is referenced by multiple really authoritative journals.

[4] http://bloggingchicks.blogspot.com/

[5] http://blogosfere.it/about.html

On the other side, it is not included by the core/periphery model, because there are no links at all from the periphery to that journal. Hence it cannot be considered as an authoritative one with confidence, since the first A-List characteristic is not met at all.

This limitation of the core-analysis towards anomalies against the first A-List characteristic has already been observed in the blog datasets, and is independently confirmed here. As mentioned before, this problem will be addressed in the following Section 4.5.

4.4.4 Graphical Evaluation

The in-CCS of a network partitions the network into the *k*-frontiers. Hence this leads to a disjoint partitioning of all nodes of the network.

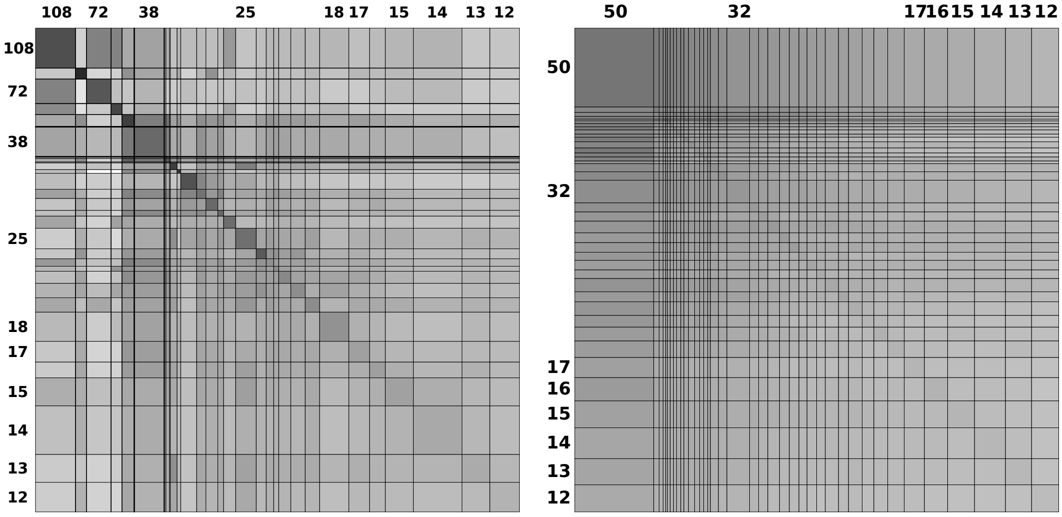
This partioning can be plotted using the GRAMs presented in Section 2.4. Figures 4.8 to 4.13 show the GRAMs for all six languages, including both, the CCSs of the real and the corresponding random networks.

It is well visible that all the random networks contain a very clean and smooth nesting of the in-cores, as discussed in Section 4.4.1. They conform very well to the illustrated GRAMs of the core/periphery model from Figure 4.1, which also depicts an idealised GRAM for a discrete partitioning in Figure 4.1c.

For the real networks, the phenomenons already discussed in the Sections 4.4.1 and 4.4.2 become graphically visible and give some additional insights over the plots of the Figures 4.2 to 4.7. Apparently, only the Portuguese and the French GRAMs conform to the graphical core/periphery model.

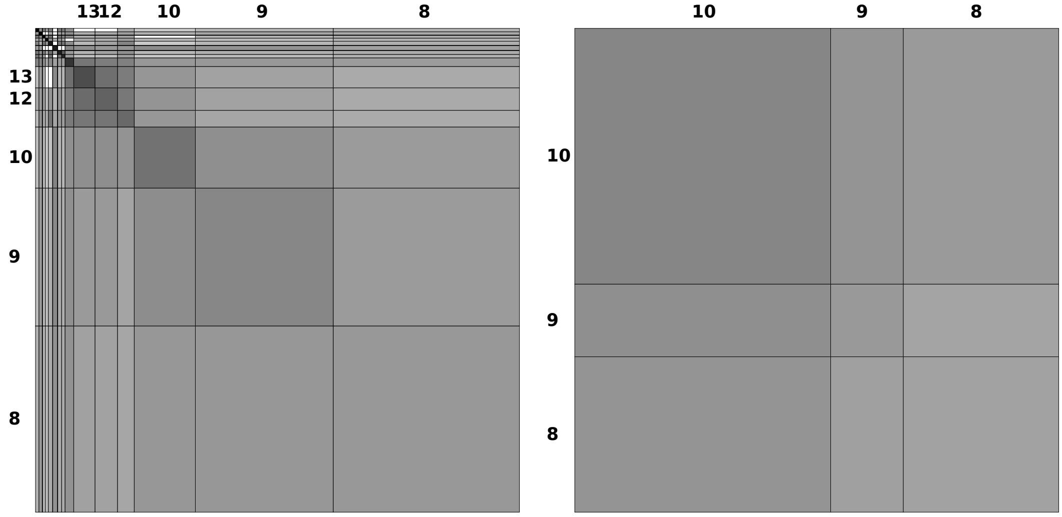
The GRAMs of the other four languages are more or less skewed in the top left corner. There are either small cohesive groups with little authority from the lower *k*-frontiers, which is represented by only lightly coloured columns, as well as disjoint groups, which are less connected to the neighbouring cores when compared with their internal density.

We have already addressed one reason for this problem in Section 4.4.2, namely the highly cohesive large subgroups with relatively little authority from the long tail. However it is relatively hard to measure this effect in the graphical representation, and thus impossible to find a solution for our overall detection problem. That is why we have to look for a computational solution with a suitable measure and a corresponding algorithm.



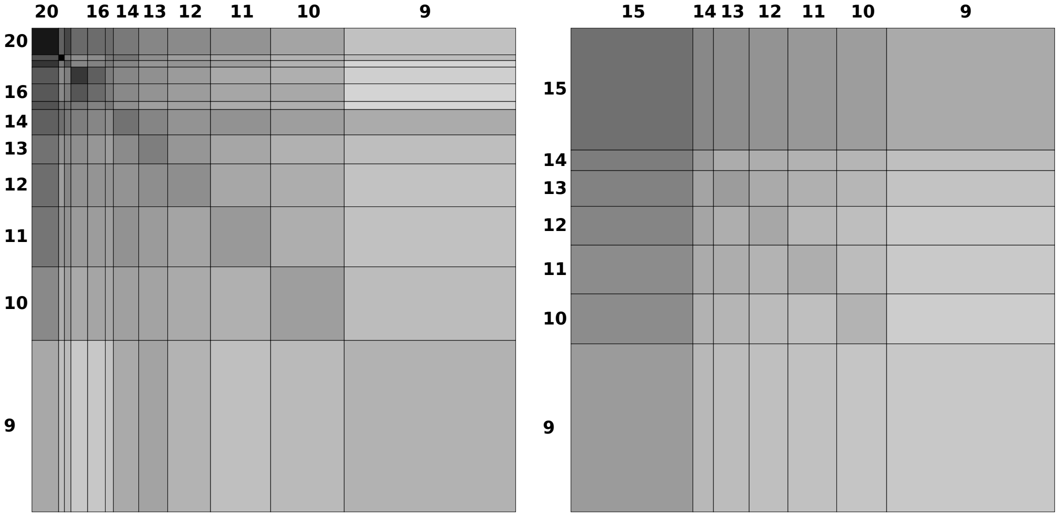
(a) real (b) random

Figure 4.8: GRAMs of the in-CCS of the real and the random English network



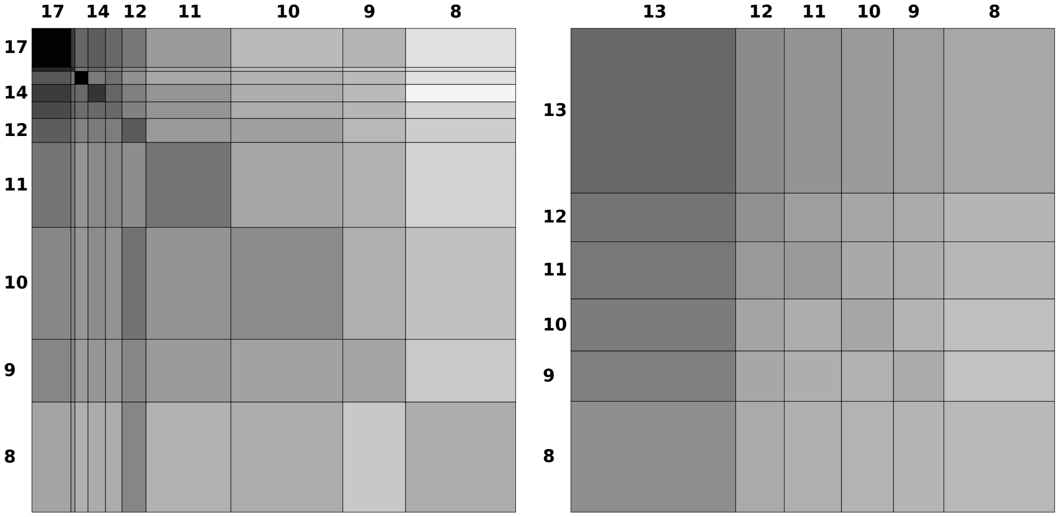
(a) real (b) random

Figure 4.9: GRAMs of the in-CCS of the real and the random Spanish network



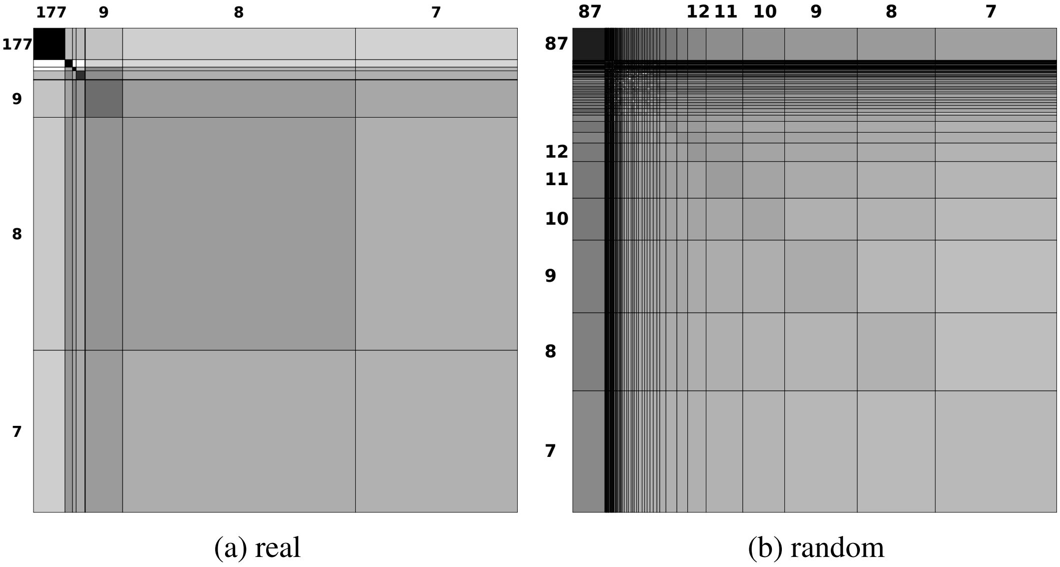
(a) real (b) random

Figure 4.10: GRAMs of the in-CCS of the real and the random Portuguese network



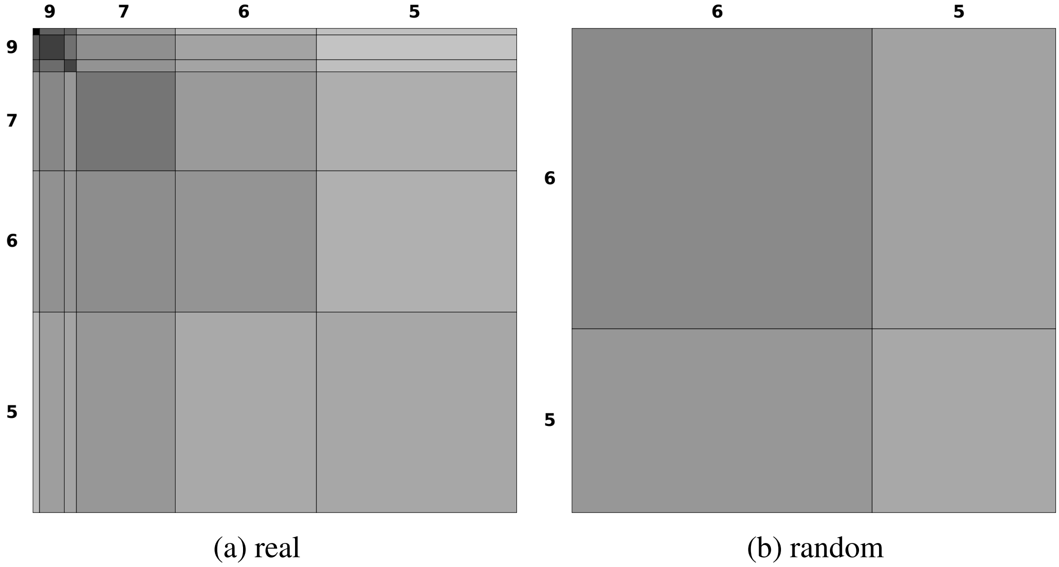
(a) real (b) random

Figure 4.11: GRAMs of the in-CCS of the real and the random French network



(a) real (b) random

Figure 4.12: GRAMs of the in-CCS of the real and the random Italian network



(a) real (b) random

Figure 4.13: GRAMs of the in-CCS of the real and the random German network

4.5 Anomaly Detection

This section addresses the problems observed in the previous section, when non-authoritative cohesive subgroups form a high *k*-in-core, and thus harden the detection of the real A-List cores. After a thorough look at the given constraints and the problematic structural properties, we develop a method to measure this anomaly quantitatively.

4.5.1 Constraints

In order to reliably detect A-List blogs, all three characteristics must be fulfilled. The in-core-analysis is mostly based on the second characteristic. However, the first and the third characteristic require an analysis of the core's relation to the periphery, which is not directly addressed by our method. In fact, the emerging cores do comply to all three characteristics in the random networks, but not necessarily in the real-world networks with their special anomalies, as we could see multiple times in the previous section.

The highest *k*-in-core of the French dataset, a 17-in-core with 274 nodes, and the highest *k*-in-core from the Portuguese dataset, a 20-in-core with 209 nodes, are the only ones that are free of such anomalies and can be immediately used as an A-List represenation. For all other original datasets, a combination with further analyses is required, where different methods have to be considered and compared.

In a first step towards this goal, we try to explicitly quantify the anomalies observed in the four problematic datasets by measuring how well core members comply to the expected characteristics of core centralisation as observed in the French, the Portuguese and the random networks.

We have to be aware of the fact that the long tail of the blogosphere is missing in our datasets, due to the nature of the data acquisition method (see Chapter 3). For example, the number of incoming links from the collaborative blogroll in the English dataset is higher than any number of incoming links a blog receives from the periphery. This would not remain true in a larger dataset with many more blogs in the lower cores. In order to detect the anomalies properly, we thus have to find a metric that is immune to the absence of periphery blogs under the assumption that these blogs are connected to the core as expected.

4.5.2 Structural Analysis

Members of higher k-in-cores in average receive more incoming links from the rest of the network than members of lower k-in cores do, which conforms to the first A-List characteristic. This is true for all random networks, but in the original blog datasets, this is true only for the French and the Portuguese ones. When not true. it is an indicator for the fact that the higher cohesion is only added by a local effect, as observed in the recipes and cooking community in the German 10-in-core for example (see Section 4.4.2).

This would work for the German and the Spanish datasets, but the average number of incoming links is not immune to the missing long tail links, as the nodes in the highest in-cores of the English and the Italian datasets have the highest average indegrees, despite being referenced less often from the long tail than many nodes in lower in-cores. This is a result of their extremely high linking amongst each other. To eliminate this effect of intra-core links, we could count only incoming links from outside the node's k-in-core.

This in turn does not account for the iterative nature of nested cores. With this metric, we would still see nodes with little incoming links from the periphery, but with high indegrees from outside their k-in-core, because a large portion of their cohesive subgroup forms an in-core with a slightly lower k, e. g., a $(k-1)$-in-core.

4.5.3 Core Independency

Our final solution is to weight each incoming link of a target node based on the core-distances between the target node and the source node, i. e., the lower the in-core of the source node relative to the in-core of the target node, the more valuable that link is for determining the effect of the first A-List characteristic.

We call this metric *core independency*, as it measures how little a node's authority depends on its fellow core members and the members of the directly surrounding cores.

Given a function $k(v)$ returning the maximum k for which a node v is a member of a k-in-core, we can define the core independency $indep(v)$ of a node v with $k(v) \geq 1$ as follows.

$$indep(v) = \sum_{i=0}^{k(v)-1} \frac{k(v)-i}{k(v)} \cdot \frac{|\{(s,t) \in E \mid t = v \wedge k(s) = i\}|}{indeg(v)} \tag{4.1}$$

For nodes that are not members of any k-in-core, the independency is 0 by definition. The values of this metric will be in the interval $[0,1[$, and the complementary metric *core dependency* can be defined as $dep(v) = 1 - indep(v)$.

4.5.4 Evaluation on the Datasets

The Figures 4.14 to 4.19 plot the core independency metric for all of our datasets, whereby the x-axis denotes the k-in-core and the y-axis denotes the corresponding average core independency of the core members. Again, the red circles represent the results from the random network and the blue squares represent the values of the original datasets.

We clearly see the constantly increasing independency values in all random networks. For the real-world networks, this is only true for the Portuguese and the French datasets. This quantitatively validates the visual impressions from the previous section. Looking at the four problematic datasets, one might want to start guessing the "real" A-List core from the peak of the independency curves, but this is a slightly misleading impression produced by the plots. Single nodes may lower the independency score of the whole in-core, while there still might be enough members inside to maintain it with an increasing core independency, and thus with the expected high authority.

Apparently, this metric is capable to visualise all the different anomalies we observed in Section 4.4. If more periphery blogs were present, the independency values in higher k-in-cores would increase in average, but the curve shapes would remain the same ones.

4.5.5 Discussion

As a metric for individual nodes, the core independency could be used to remove nodes under a certain threshold from the final A-List candidate list, or "the core" according to the interpretation of the core/periphery model. In fact, the problematic journal "ASW" mentioned in Section 4.4.3 has a core independency of 0, which makes it a candidate for removal, no matter what threshold above 0 would be chosen.

However, instead to define an arbitrary core independency threshold for A-List blogs, which would have to be experimentally guessed for each new dataset, we are looking for a more systematic and reliable solution in the next section.

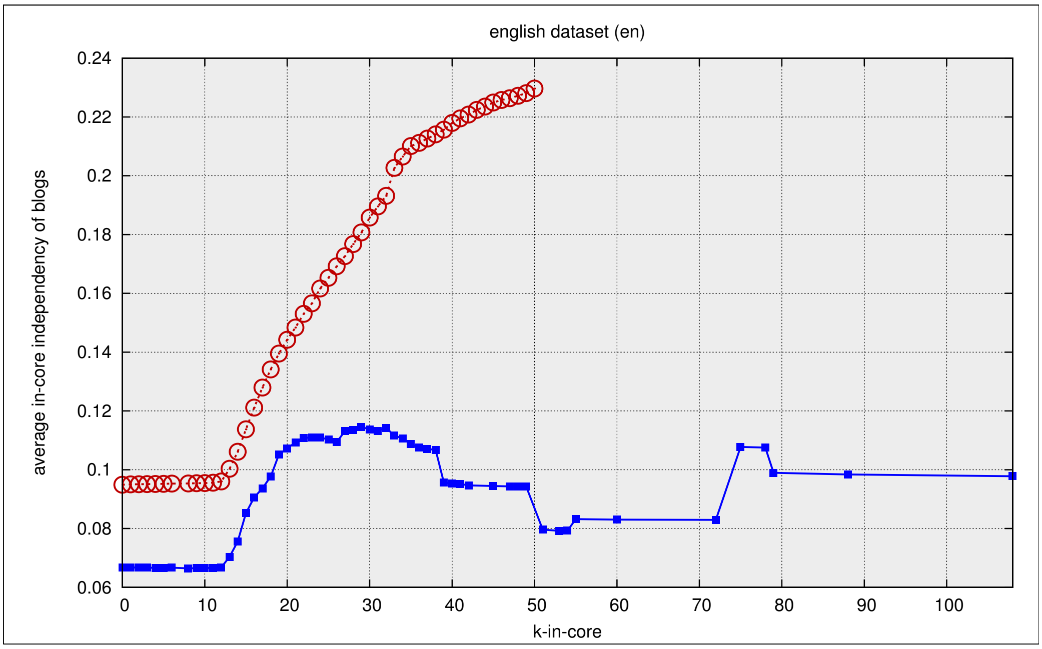


Figure 4.14: Average independencies in the English in-cores

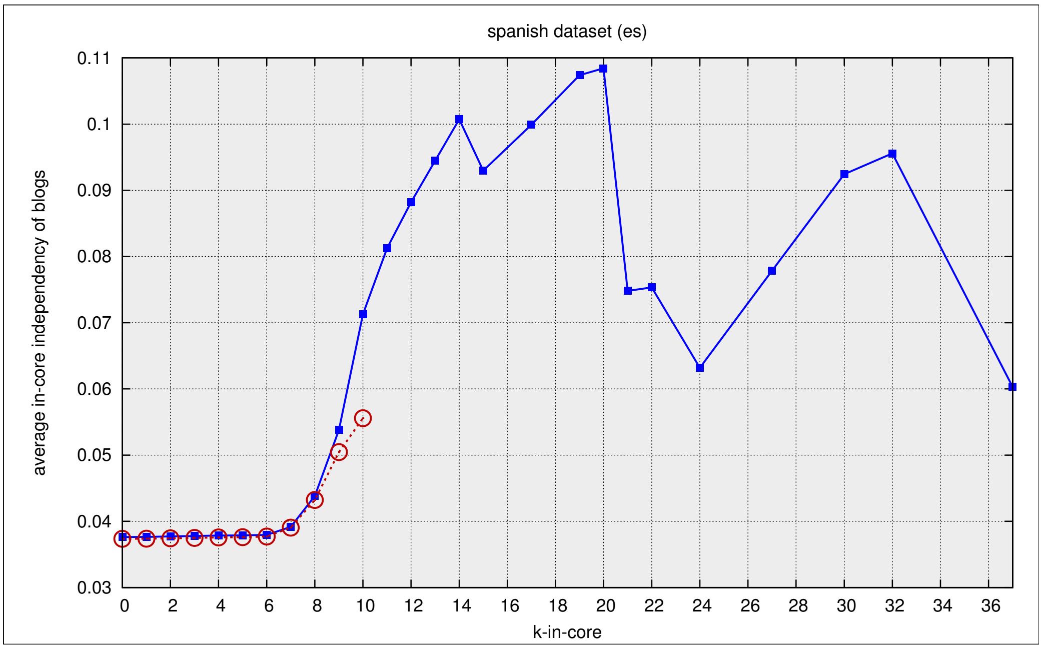


Figure 4.15: Average independencies in the Spanish in-cores

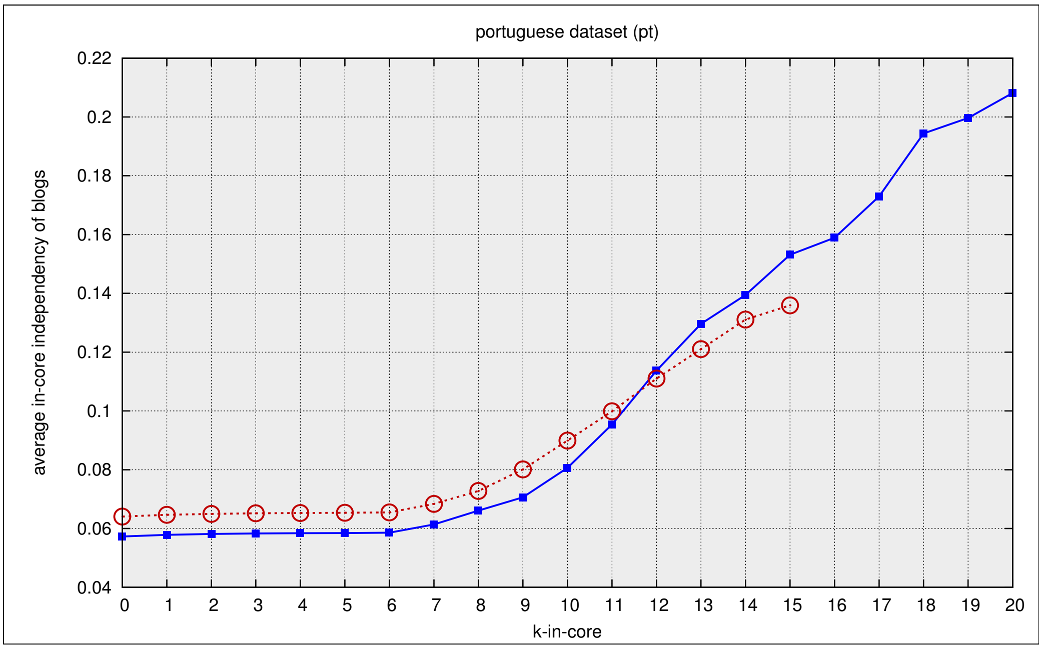


Figure 4.16: Average independencies in the Portuguese in-cores

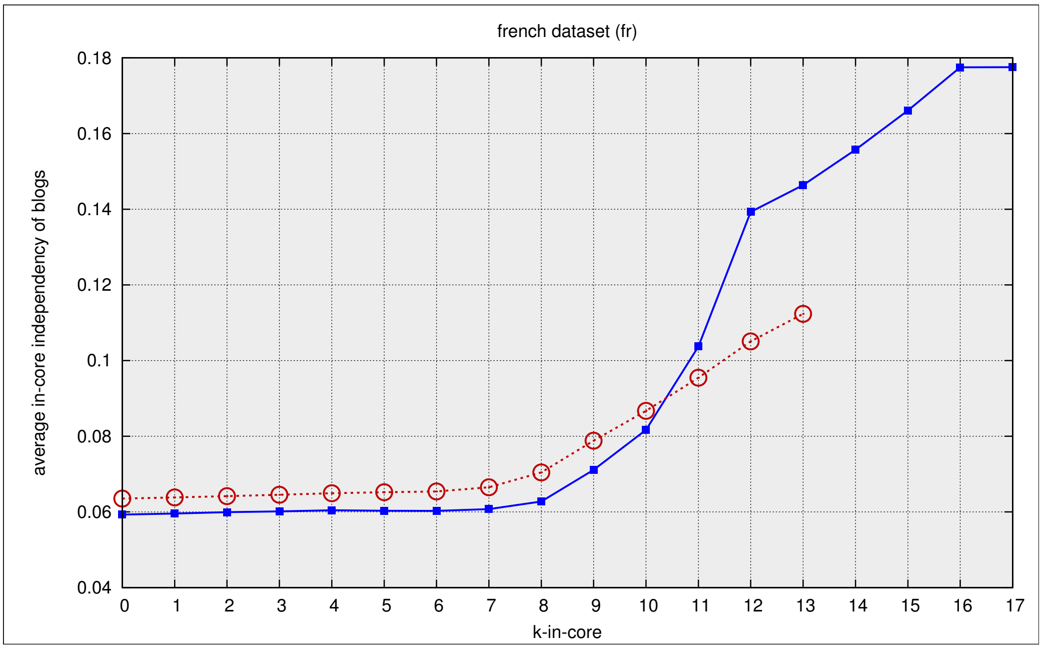


Figure 4.17: Average independencies in the French in-cores

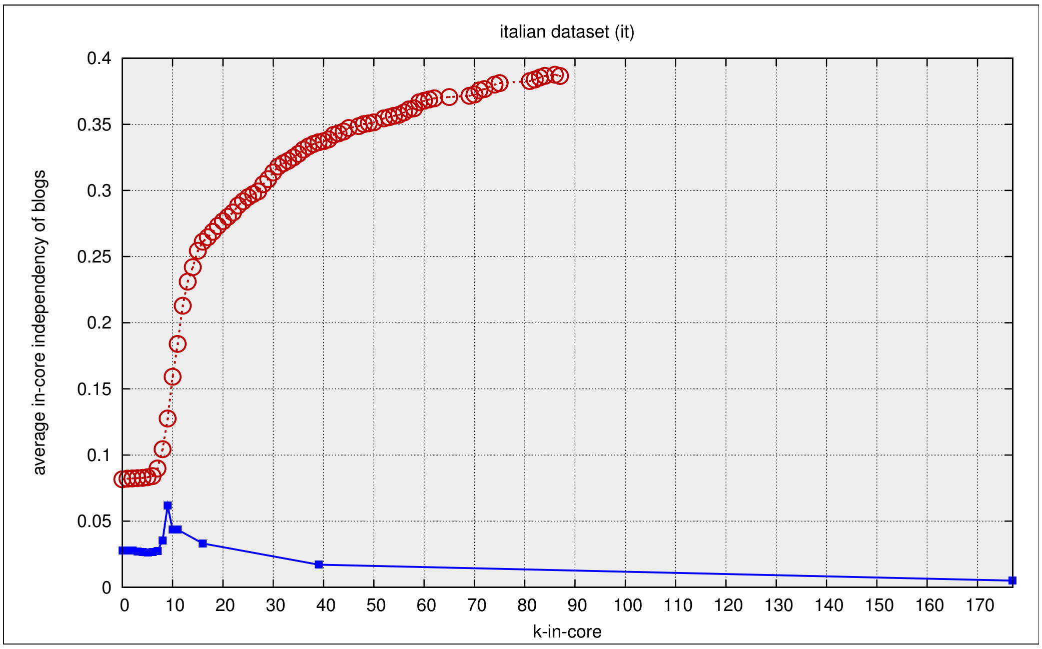


Figure 4.18: Average independencies in the Italian in-cores

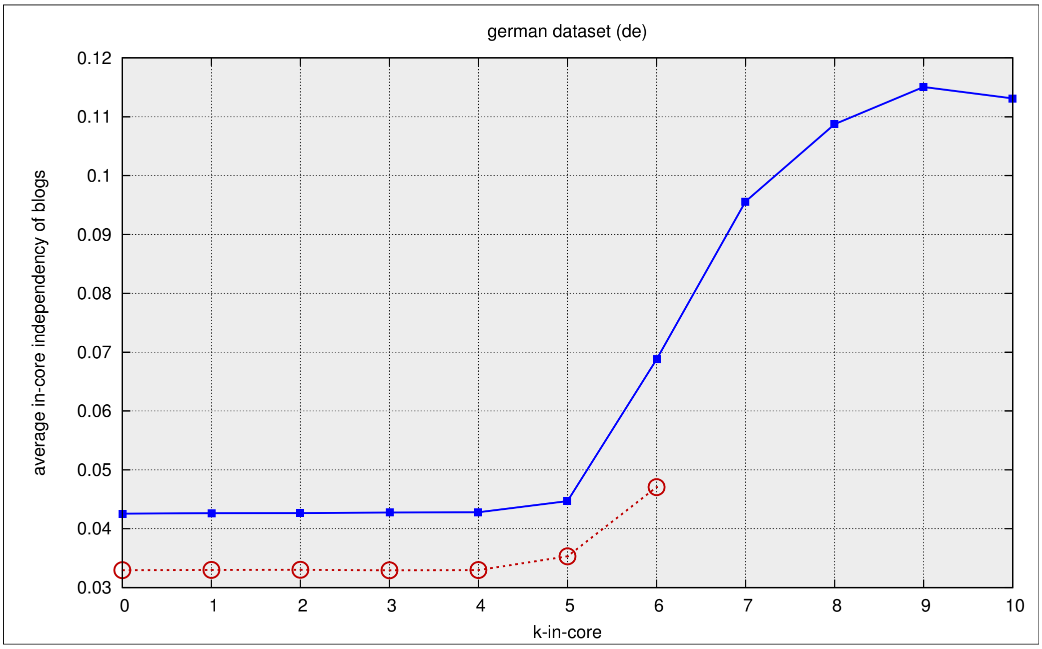


Figure 4.19: Average independencies in the German in-cores

4.6 Community Detection

The previous sections clearly showed that dense cohesive subgroups are the reason behind anomalies that emerged in the attempt to detect the A-List group with the in-core algorithm. In order to work around this issue, we take a closer look at this concept. In this section we look at structural clustering methods for community identification, and analyse the blogroll networks with respect to this structural property. These insights should be helpful for finding a solution to the A-List detection problem afterwards.

4.6.1 The Community Concept

The identification of structural communities in graphs is an active research topic for a long time, but also a very difficult one, due to the usually complex structures in large real-world graphs. A good recent review on related methods and algorithms is given by Fortunato (2010). In this thesis we adhere to the concept from Newman (2003), who defines communities as "groups of vertices that have a high density of edges within them, with a lower density of edges between groups" (Newman, 2003, p. 17).

In the context of this thesis, we limit ourselves to this concept of disjoint communities. There also exists some research on overlapping community concepts and detection.

Visualisation

This definition is apparently well suited for visualisations with GRAMs (see Section 2.4), where one can directly compare the density inside a community, corresponding to the greyscale saturation in the diagonal field of the partition, with the densities to other groups, corresponding to the saturations in all the other row and column fields of the partition. A good example has already been given with the multi-language network in Figure 3.1.

Notation

When partitioning a network into different disjoint communities, this is also called a *clustering* of the network. A clustering C is a set of clusters $\{c_1, c_2, ..., c_t\}$, with $c_i \subset V$, $c_i \cap c_j = \emptyset, i \neq j$, and $\cup_{1..t} c_i = V$.

In the context of such a clustering $E(c_i, c_j)$ denotes the set of all edges, which are incident to both, a node of c_i and a node of c_j. Similarly, $E(c_i)$ is synonymous to $E(c_i, c_i)$ and returns the set of edges inside a cluster.

4.6.2 Quality Metrics

For a quantitative measurement of the quality of a community, we consider the measures of *modularity* from Newman (2006) and *conductance* from Leskovec et al. (2009). Both naturally use the relation of internal links to external links for their computation.

Conductance

The conductance value of a single cluster c_i is simply the number of external links of a group divided by its number of internal links (see Leskovec et al., 2009, p. 3).

$$conductance(c_i) = \frac{|E(c_i, C \setminus c_i)|}{|E(c_i)|} \tag{4.2}$$

This means that a lower value indicates a better community character. However, the interpretation of this value is extremely difficult, since it is independent of the size of the cluster, the rest of the network and the overall number of edges.

Modularity

The modularity of a clustering is a value in the interval $[-1, 1]$, defined to measure the overall quality of the community structure of the clustering. It is the sum of the *module values* of all clusters, and should be maximised in order to obtain an optimal clustering. The module value measures the density within a group relative to the average density in its row and column and the rest of the network. The modularity formula for directed networks can be found in (Fortunato, 2010, p. 34, eq. 37). Expressed in our notation, the module value for a cluster is calculated as follows.

$$module(c_i) = \frac{|E(c_i)|}{m} - \left(\frac{|E(c_i)| + |E(c_i, V)|}{2m}\right)^2 \tag{4.3}$$

The modularity is a very recognised measure for clustering optimisation, but the utility of the module values as a quality metric is limited. It is not normalised to the cluster size, since it is designed to provide its effect in an overall sum over all clusters.

Density Ratio

During our analyses we often found a correlation between the two metrics, but also often a discrepancy. Both metrics have different potential biases, especially related to the cluster size. While conductance is easier to understand, modularity matches the community definition better. We will consider both metrics in the rest of this thesis, but we also add a third metric measuring cluster quality with respect to the size of the cluster and the size of the rest of the network.

Following the community definition and the graphical representation with GRAMs, the natural consequence is to measure the relation of the density inside a cluster to the density of its connections to the rest of the network. We call this metric *density ratio* and define it as follows.

$$ratio(c_i) = \frac{|E(c_i)|}{|c_i|^2 - |c_i|} / \frac{|E(c_i, V \setminus c_i)|}{2 \cdot |c_i| \cdot |V \setminus c_i|} \tag{4.4}$$

We generally prefer this metric for the measurement of a community's strength, since it is suitable for communities of any size and independent of the clustering of the rest of the network. Additionally, the ratio is equivalent to the factor over which a community node is statistically more probable to be connected to a community peer, as opposed to an external node.

4.6.3 The Louvain Method

Rueger (2010) evaluated a number of popular existing clustering algorithms on our blog datasets. Among them are divisive, agglomerative, hierarchical and non-hierarchical ones. One basic task was to separate the extremely cohesive language communities in the multi-language network with its nearly one million vertices (see Figure 3.1). Most algorithms failed here, producing an endless number of small communities, with worse quality metrics than those of the predefined language groups. Also, some algorithms with quadratic or even cubic runtime could not complete the task in an acceptable time (see Section 2.2.3).

In this thesis we need one algorithm that can efficiently identify a good share of the communities that are present in our specific datasets. In summary, the Louvain method by Blondel et al. (2008) seems the most suitable algorithm for us. Based on modularity maximisation, it returns hierarchical results in apparently linear runtime, without the need to play around with parameters.

Example: The Multi-Language Network

We first evaluate the algorithm's performance on our multi-language network. On the most coarsely granular level of the multi-language network, it clusters the network into 18 clusters. Figure 4.20 shows the corresponding GRAM, in which we already rearranged the clusters, ordering them by language just like in Figure 3.1.

In each cluster one language is highly dominating, so the separation is considered to work as desired. This is seconded when looking at the modularities of the clusterings. The clustering by blog language, as given in Chapter 3, has a modularity of 0.637, while the Louvain method's clustering has a modularity of 0.826. So by the means of this metric, it yields an even better clustering.

This is also a good example to illustrate the interpretation problem with module values. Originally, the English blogs have a module value of 0.24. The best cluster identified by the algorithm is an English subcommunity with a module value of 0.16.

4.6.4 Clustering in the Blogroll Networks

We use the Louvain method for identifying communities in the six language-specific blogroll networks. We expect communities to be formed because of similar interests, or due to some kind of organisational ties among the member blogs.

Based on the feed entries of the blogs, we extracted the ten most characteristic keywords for each cluster based on the TF-IDF values (Baeza-Yates & Ribeiro-Neto, 1999). Additionally, we manually annotated around 50% of the Portuguese and the German blogs with general tags about the blogs' topic, in order to get a representative insight into the community's topics. Frequent tags were *politics, culture, internet, personal*, etc.

In all datasets we are able to identify specific communities with an explorative analysis. Some of them are organisational communities, like the `blogosfere.it` group or the "blogging chicks" (see Section 4.4.2), but most of them are communities of shared interest. There often are technical and political communities, as seen in previous studies (Herring et al., 2005; Zhou & Davis, 2006).

Example: The Portuguese Network

For a representative insight, we take a closer look at the Portuguese dataset. Figure 4.21 displays the GRAM for the clustering of the Portuguese blogs, whose communities are described in Table 4.1, with their quality metrics and the most frequently associated tags.

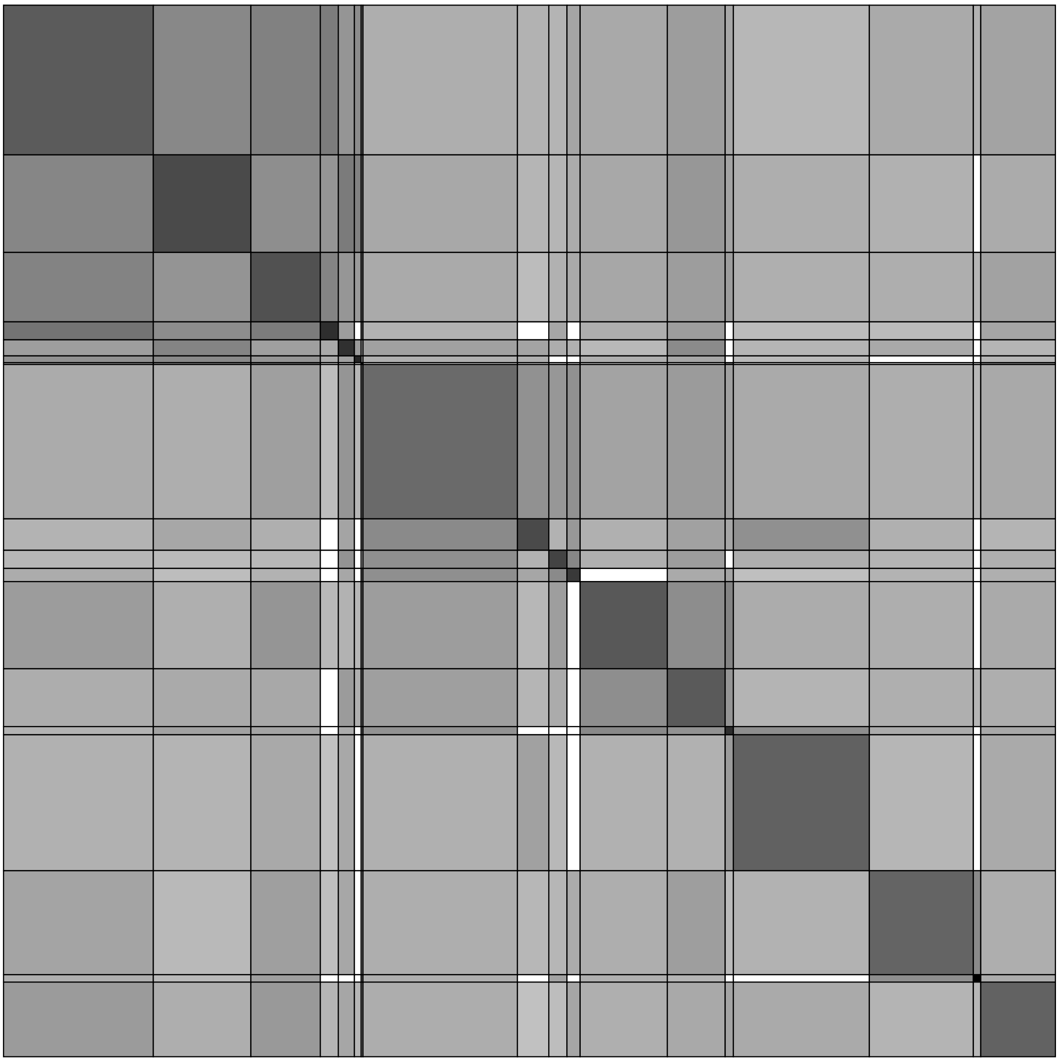


Figure 4.20: GRAM of the Louvain clustering of the multi-language network, with groups ordered by language (compare with Figure 3.1)

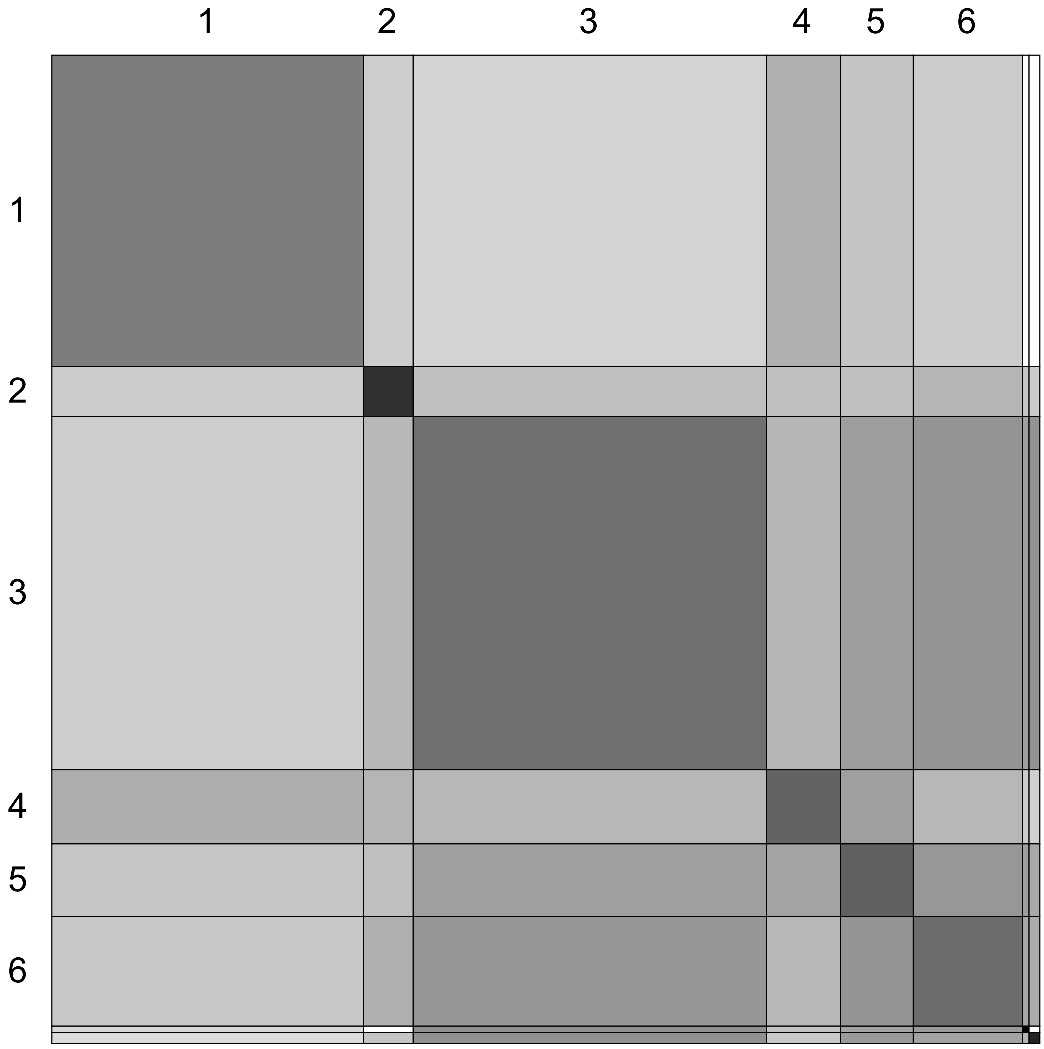


Figure 4.21: GRAM of the Louvain clustering of the Portuguese dataset

id	size	conductance	module	ratio	tags
1	1,181	0.08	0.166	32.9	blogging, technology, internet
2	189	0.12	0.054	191.9	culinary
3	1,339	0.25	0.193	12.2	politics, culture, personal
4	281	0.75	0.029	30.3	personal
5	276	1.46	0.030	16.4	*unspecific*
6	415	1.63	0.042	9.6	*unspecific*
7	24	1.75	0.003	184.0	politics
8	41	2.40	0.004	76.4	politics, left

Table 4.1: Characteristics of the identified Portuguese clusters

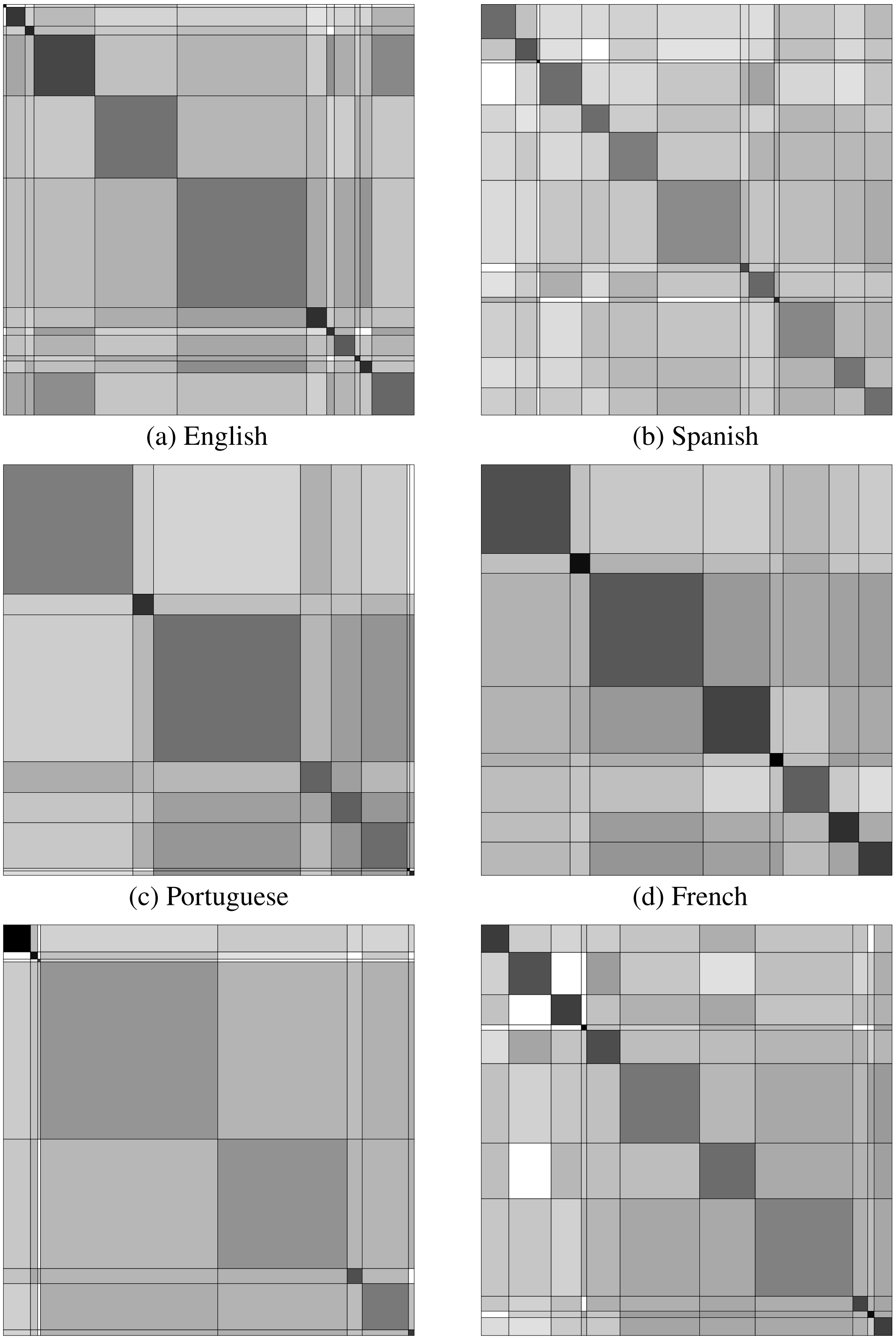
Cluster 1 is a well-defined “technology & web” community. The visually best cluster (and thus also by density ratio) is number 2, whose members share recipes and food information. These *culinary* communities are also the best defined communities in the French and the German network, a phenomenon not seen before in other studies. One reason for that might be their contentual distance to typical A-List blogs about politics, culture and technology.

Community number 3 is a mix of political and cultural blogs. It is neither cohesive by topic nor well detached from the remaining communities. A comparison to the core/periphery model of the Portuguese dataset, shown in Figure 4.10a, reveals that 205 of the 209 members of the Portuguese 20-in-core are members of community number 3 in this clustering. Since politics and culture are the topics of the most popular Portuguese blogs, the core/periphery structure resulting from the A-List effect prevents the two communities from being separable by a clustering algorithm in this case, as this core group is cohesive, and also has good connections to the rest of the network, This is an effect often seen in clusterings of real-world networks, called “the absence of large well-defined clusters” by Leskovec et al. (2009).

The Other Networks

The community structure in the other five datasets is very similar. Figure 4.22 shows the GRAMs for the most coarsely granular clusterings of all six datasets in direct comparison, where all of them are plotted with the same parameters for density saturation scaling. The emerging structural communities are well visible.

The modularity values second this impression, with 0.651 for the English, 0.750



(a) English

(b) Spanish

(c) Portuguese

(d) French

(e) Italian

(f) German

Figure 4.22: GRAMs of the clusterings of all the six datasets in direct comparison

for the Spanish, 0.524 for the Portuguese, 0.619 for the French, 0.592 for the Italian, and 0.602 for the German clustering.

These observations definitely confirm our assumption that there is a strong community structure in the datasets, which is present simultaneously with the A-List structure. This fact causes the problems described in Section 4.5.

4.7 Network Filtering

In Sections 4.4 and 4.5 we have seen that certain patterns of community structure in a network harden the detection of core/periphery structure. And vice versa, in Section 4.6 we have seen that core/periphery structure may harden the detection of community structure. In this section, we show how the detection of core/periphery structure with the in-core algorithm can be made more reliable when using clustering knowledge.

4.7.1 Sparsification

We suggest a sparsification of community-internal links for the problematic large and very cohesive communities, which do not play any role in global core/periphery structure. Following the definition of communities from Section 4.6.1 and the corresponding density ratio metric from Section 4.6.2, a community is defined by having a density ratio clearly greater than 1.0.

Once such a problematic community is identified, we can eliminate the community structure without impacting the real core/periphery structure. Eliminating the community structure can easily be achieved by bringing the density ratio to exactly 1.0. Alternatively, you also may just reduce the community's strength by bringing its density ratio to a clearly lower value.

Selecting the communities that need to be sparsified, and deciding how exactly to sparsify them, always results in a heuristic approach, and thus always depends on experience and the datasets in question. Remember that our datasets are just a small authoritative excerpt from the blogosphere, as described in Chapter 3. In order to fully consider the first A-List characteristic, the massive linking from the long tail (see Section 1.4), we would need the set of all blogs, or at least a large representative part of the long tail. In this thesis, we show that the sparsification approach can provide very good results on an example, where the missing long tail does not have too much impact.

id	size	conductance	module	ratio	links	sparsification
1	182	0.01	0.244	2,621	32,230	32,217
2	49	0.12	0.022	793	1,720	1,717
3	18	0.21	0.001	1,001	71	70
4	1,192	0.41	0.149	6	16,501	-
5	871	0.56	0.107	6	10,349	-
6	101	0.57	0.019	89	1,511	1,494
7	311	0.92	0.044	17	3,731	-
8	38	1.16	0.005	122	396	392

Table 4.2: Characteristics of the identified Italian clusters

We choose the problematic communities by selecting a threshold for the density ratio. For these problematic communities we then sparsify the internal links to turn their density ratios to 1.0. This is achieved by randomly removing the required fraction of cluster-internal links. The required fraction is computed as follows.

$$p(c_i) = 1 - \frac{1}{ratio(c_i)} \tag{4.5}$$

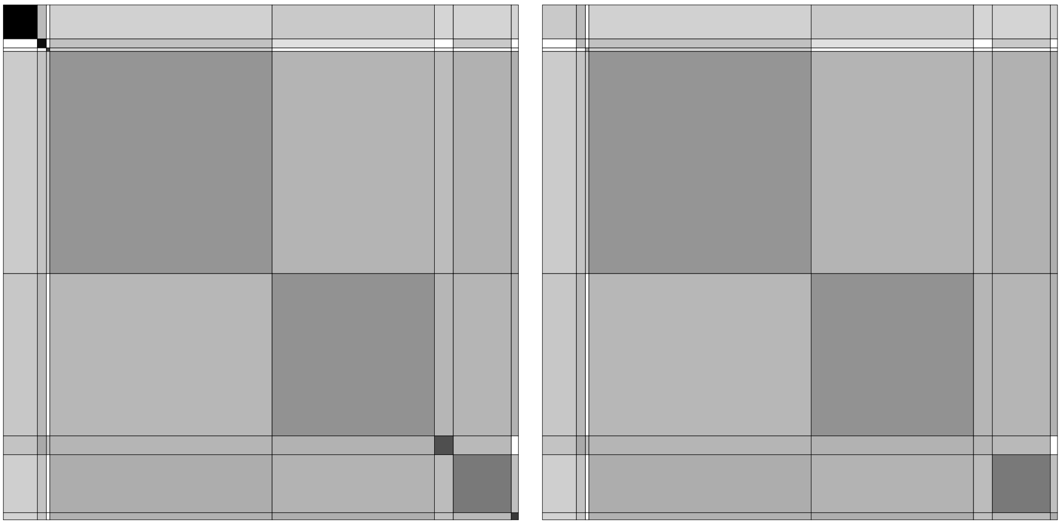
This can be implemented either by removing each edge in the cluster with the probability $p(c_i)$, or by randomly selecting $p(c_i) \cdot 100\%$ of the edges $E(c_i)$, and by deleting this selection. That way, the community structure is completely eliminated, and the underlying anomaly that prevented a direct core/periphery detection is removed, such that a new run of the in-core algorithm on this sparsified network should yield a more accurate approximation.

However, while the revised result should yield the right group of A-List blogs, one has to be aware that the sparsified network is a slightly different one.

4.7.2 Filtering the Italian Network

In search for an easy example we choose the Italian dataset, whose original CCS is shown in Figure 4.12a. It suffers from a clearly non-authoritative 177-in-core that prevents a direct detection of the core/periphery model by the in-core algorithm. The Louvain method detects all of these blogs in the first community of 182 Italian blogs, as depicted in Figure 4.22e.

Table 4.2 shows the identified Italian clusters along with their quality metrics.



(a) original

(b) sparsified

Figure 4.23: GRAMs of the Italian clustering before and after sparsification

Again, it is apparent that only the density ratio makes sense to be considered for this approach, as the other two metrics are not invariant to cluster sizes.

We decide for a threshold of 50, and sparsify the five problematic clusters as described above. Table 4.2 lists the cluster-internal links in the original network, and gives the number of edges removed by random from these clusters. Figure 4.23 shows the GRAMs of the original Italian clustering, and the structure after the sparsification. The five communities have apparently disappeared, just like intended.

The filtered network now still consists of 2,773 nodes as before, but contains only 39,531 edges, opposed to 75,421 in the original network.

4.7.3 Revised A-List Detection

As outlined before, we expect the filtered network to be free of larger anomalies, which prevented a direct A-List detection by the in-core algorithm in the first attempt (see Section 4.4). We are now running the algorithm on the filtered network and evaluate the results just like we did before.

Figure 4.24 shows the CCS of the filtered Italian network. Again, we also plot the results for a corresponding randomly generated network for comparison. Despite the filtering, the network still contains a higher tendency towards a small dense k-in-core

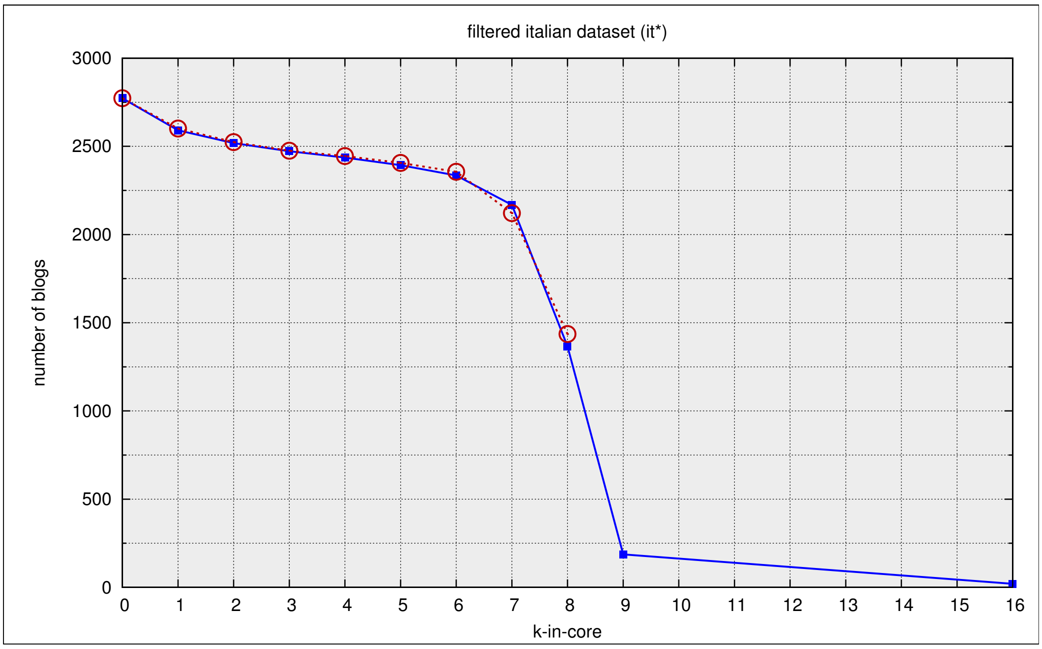


Figure 4.24: In-CCS of the real and the random Italian network after filtering

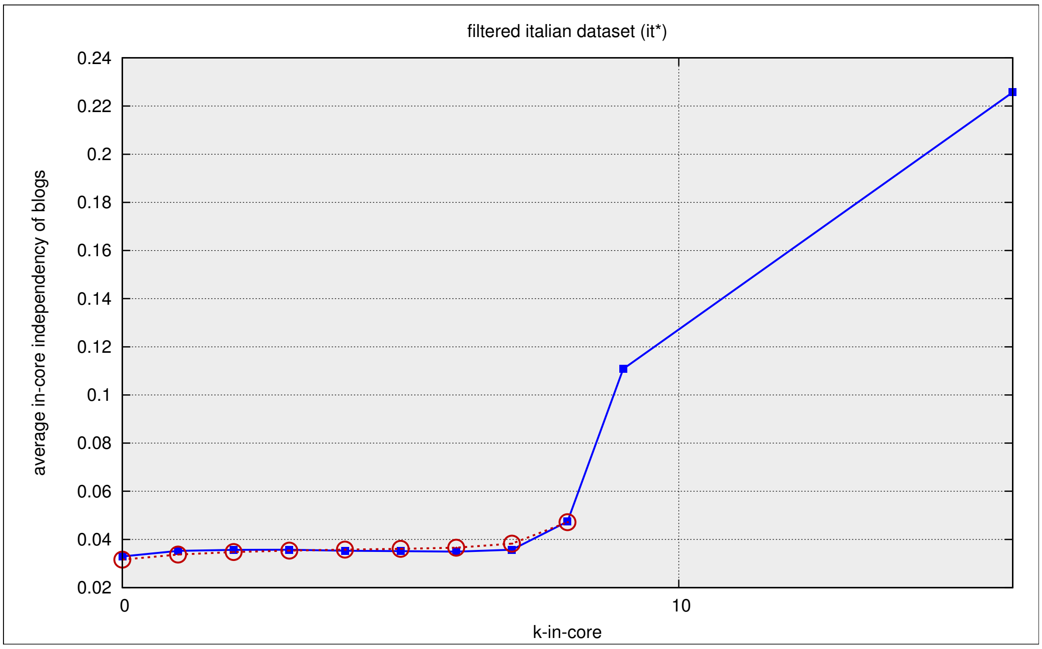
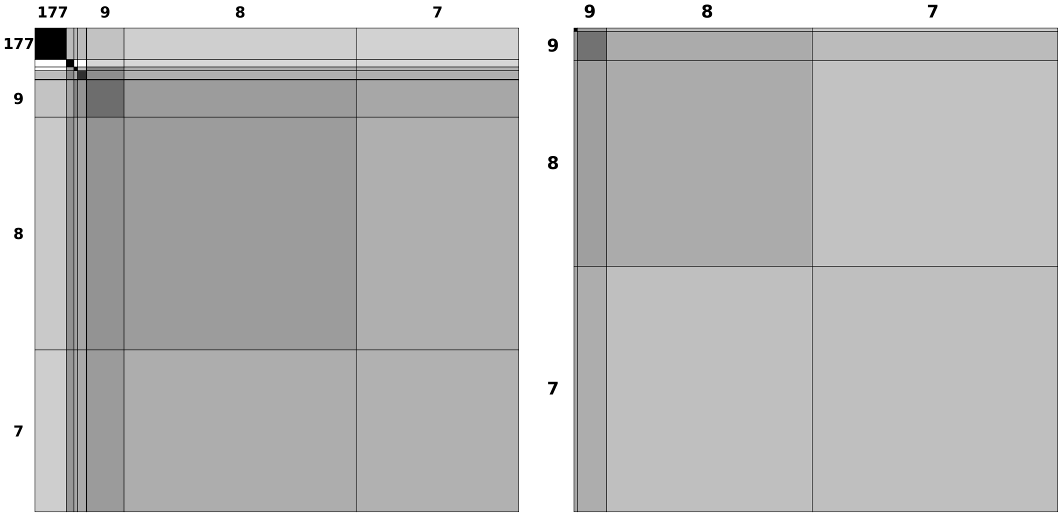


Figure 4.25: Average independencies in Italian in-cores of the filtered network



(a) original

(b) filtered

Figure 4.26: GRAMs of the in-CCS of the original and the filtered Italian network

than the random network does. Figure 4.26 additionally shows the GRAM of the new in-CCS in comparison to the original in-CCS of the Italian network.

Furthermore, the plot of the average independencies of the core members in Figure 4.25 shows that the in-core independencies are constantly increasing. This means that our major indicator for anomalies does not indicate such an anomaly in the filtered dataset anymore.

Therefore, we can assume that the 16-in-core of the filtered Intalian network is indeed a cohesive group with high authority from the rest of the network. The same is true for the larger 9-in-core, which could be interpreted as a wider A-List group, since the 16-in-core is very small with only 20 blogs.

CHAPTER 5

Application in Blog Monitoring

This chapter presents an application of computational SNA of blog authority, which has been implemented in the context of a blog monitoring tool.

5.1 The Social Media Miner Project

The Social Media Miner (SMM) is a research project that was conducted in the Knowledge Management Department at DFKI, the German Research Center for Artificial Intelligence, from December 2008 to November 2010, in cooperation with a media consulting agency. It was funded by the IBB Berlin[1] and co-financed by the EFRE fonds of the European Union.

5.1.1 Monitoring of the Blogosphere

As already discussed in Section 1.3, the blogosphere contains a huge amount of information created by a multitude of sources. According to the "Technorati State of the Blogosphere" (Sobel, 2010) there are at least 900,000 articles published each day, with an upward trend.

Whenever the question arises how a product, a brand, a personality, an institution, a technology or some other specific entity is perceived by the public, the blogosphere is a good source of information. In this project, such an entity is defined as a *domain*. These specific domains usually interest professionals in marketing and PR

[1] `http://www.ibb.de/`

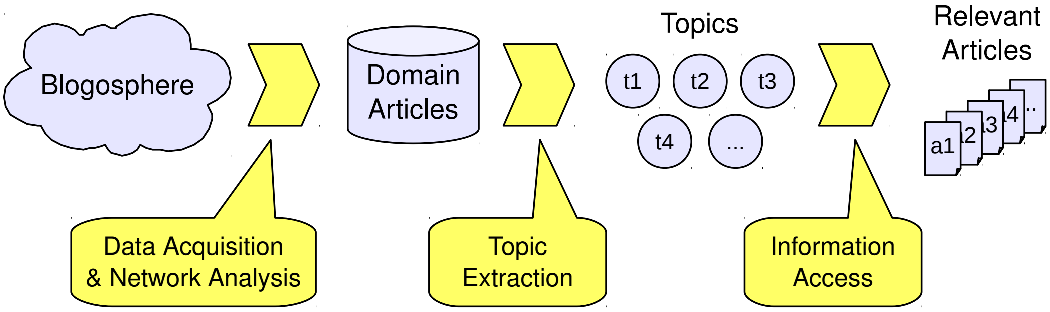


Figure 5.1: SMM workflow

businesses the most, opposed to the broader interests of sociologists and blogosphere researchers.

Modern search services offer a rich set of tools to monitor or track the blogosphere, but the analysis with respect to a specific domain is very limited. For example, Icerocket Blog Trends[2] can plot the number of articles per day for a specific query. It plots a static, non-interactive curve, but there is neither an explanation of this curve nor access to further information. It has to be post-processed manually with different tools by the market researcher.

From our experiences we know that there is a strong demand for business oriented social media monitoring, with the ultimate goal to make better decisions thanks to better information. That demand cannot be served by search services yet, thus the project wanted to create a blogosphere-specific methodology to bootstrap such business intelligence systems.

5.1.2 Goals

In this chapter we pursue three concrete goals to enable domain-specific blogosphere monitoring, which will then enable business intelligence applications. These applications can then perform clustering, trend detection, information extraction, sentiment analysis, or other content-based mining technologies on top of this data.

Figure 5.1 shows the workflow realised in the SMM project. The collected articles are post-processed by a topic clustering component, which gives a chronological overview of the activities inside a domain for a given timeframe. The information access per topic is then supported by a relevance ranking of the articles.

[2] `http://trend.icerocket.com/`

The focus of this chapter is limited to describing the foundational social network analysis and mining aspects. We will justify all of our decisions, and provide empirical evidence where possible.

Data Aggregation

As a first goal, we try to aggregate as many articles of the domain as possible. Kumar et al. (2005) have shown that in blogspace information evolves in bursts. This has been successfully modeled by Goetz et al. (2009). In consequence, there is a repeater effect for information, and the more articles we have at hand, the better the extent of this effect can be observed and exploited in textual processing methods. A selection of relevant articles can still be made afterwards, when presenting results to the user.

Authority Measurement

In order to enable this selection, it is our second goal to derive a meaningful measure of social authority, based on links among blogs and articles. The more articles we have at hand, the better the interconnectivity between them. And the more accurate the social authority derived from these links, the better the filtering and ranking that can be presented to the user in the end.

Time Sensitivity

Third, we will enable the approach to principally work over very long time periods of monitoring. Therefore, we need a metric of *attention* for articles, that can find the "hot" articles and blogs in our evolving domain at any given point of time.

Furthermore, we want to have a good and relatively stable overview of the opinion-leading blogs in a specific domain after a longer period of observation. This could be called the domain specific A-List.

5.2 Crawling Domain-Specific Blog Articles

In order to find blog articles of our domains, we define the keywords for an appropriate search query and aggregate the search results from multiple blog search services. That way. we do not have to set up a complete search engine infrastructure by ourselves, and we can reach more articles than a single search service can provide, as our experiments will show.

5.2.1 Existing Experiences

An indicator for the hypothesis that search engines obviously have very different indexes, is given by Herring et al. (2005), who noticed huge differences when comparing different Top 100 lists with each other.

In a preliminary experiment Wortmann (2009) manually analysed the quality and reach of five popular blog search services to validate this hypothesis. These services were Technorati[3], Google Blogsearch[4], Bloglines[5], Icerocket[6] and BlogPulse[7]. The domain of this test was represented by the keyword “Henrietta Hughes”, which unequivocally refers to an event on February 10th, 2009, when this homeless person talked to US president Barrack Obama. The event had a noticeable impact on broadcast media, as well as on social media, especially the blogosphere.

None of the services delivered more than 50% of all the articles found, and concerning the validity of the search results, there was a number of non-blog articles and pages not even mentioning the lady’s name. Google Blogsearch had a comparatively high false positive rate of 50%, and consequently, we left this service out of the final aggregation component. With these experiences, we implemented a number of heuristics to detect non-blogs, based on the URL, meta data and the site content, in order to filter out as many of the invalid results as possible.

5.2.2 The Aggregation Component

For our analyses, we need the URL of each blog article along with the date of publication, the title and the textual content. As the methodology is intended to monitor a domain over a very long period of time, the crawler is implemented as a permanently running service that regularly queries the search services for the latest articles, and adds these to the dataset.

All search services allow to return the query results unfiltered and sorted by date, enabling us to quickly fetch all the latest results. Each search result is listed with the notion of the article’s age. In a second step, each result is validated and, if a feed entry is available on the blog site, the more accurate date and the textual content is saved from it.

[3] http://www.technorati.com/

[4] http://blogsearch.google.com/

[5] http://www.bloglines.com/

[6] http://www.icerocket.com/

[7] http://www.blogpulse.com/

#	domain	articles	article links	blogs	blog links
1	Android G1	2,511	416	1,319	460
2	VW Golf	1,328	99	806	136
3	Toyota Hybrid Car	2,719	138	1,521	246
4	Angela Merkel	2,150	103	1,415	1,057
5	Robbie Williams	3,595	84	2,253	517
6	Fraunhofer	348	10	289	23
7	Google Wave	15,836	2,017	10,594	4,793

Table 5.1: Overview and characteristics of the example domains

Another important aspect of our datasets is the link structure among these articles. We want to track all links, where the textual content of an article is citing another blog article in the domain. These links are used later as a social assessment of the authority of articles, as widely known from PageRank (Page et al., 1998) and similar algorithms.

We impose some requirements on these article links, in order to include only expressive ones. First of all, links between articles on the same blog are ignored, since their expressiveness of authority is doubtful at best. These often appear in a "Related Articles" section at the end of an article. Links from articles that contain dozens of references are also ignored, as these are usually spam articles trying to manipulate PageRank and other ranking algorithms.

In a next step, we extract the underlying blog URLs out of the article URLs and gain a second type of data, the blogs. We then collect the blogroll links between these blogs, according to our method presented in Chapter 3. They will serve as supplementary authority indicators in the following network analyses.

5.2.3 Example Data

We have chosen a number of different domains, from products over services up to personalities, to test our methodology on them. All seven domains have been observed during October 2009, and the data is available on the author's homepage[8] as a zipped MySQL dump file. Table 5.1 lists the seven domains along with the number of articles, blogs and links.

[8] http://www.dfki.uni-kl.de/~obradovic/data

Based on this data, we have analysed the performance of the four search engines that we used. Figure 5.2 depicts each search engine with two values. The left blue bar denotes the percentage of articles of the aggregated set that was found via this engine, the right red bar denotes the percentage of articles of the aggregated set that was found only via this engine. For our datasets, none of the search engines was able to find more than 50% of all articles, but each one contributed a significant share of articles that was not known to any of the other three engines.

This is in principle what we had expected and why we have chosen a meta search approach, but the extent of the effect was not foreseen. It becomes more apparent when looking at the ratios of articles based on the number of engines they were found in. Figure 5.3 plots this data and reveals that only 1.5% of all articles were found by all four search engines, the remaining 98.5% were unknown to at least one of the engines, and nearly 70% of the articles were found only via one engine.

With this characteristic number, which we call the *appearances* of an article, we have another independent measure of article popularity available. Later, Figure 5.6 will reveal that there is a high correlation between the number of appearances of an article and its number of citations.

5.3 Determining Social Authorities

Social authority can be defined as a metric of centrality, importance or relevance induced by inbound links in social networks. There are many different metrics for authority in the field of SNA, which are all based on graph algorithms.

5.3.1 Authority Values

In this chapter we do not focus on a specific metric for the measurement of authority. The presented methodology is intentionally designed to work with an abstract authority metric, with some constraining assumptions. We assume to have an abstract authority function *auth* returning normalised authority values for a given node.

$$auth : V \to [0,1] \tag{5.1}$$

An important property of this function for our reasoning in this chapter is the direct dependency on the indegree of a node, as defined below.

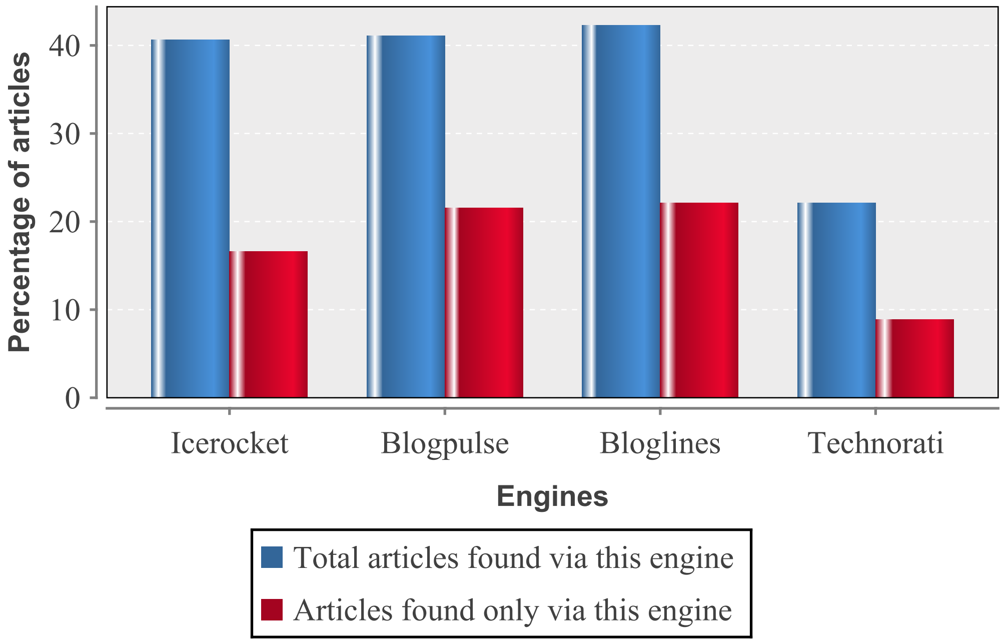


Figure 5.2: Performance comparison of the selected blog search engines

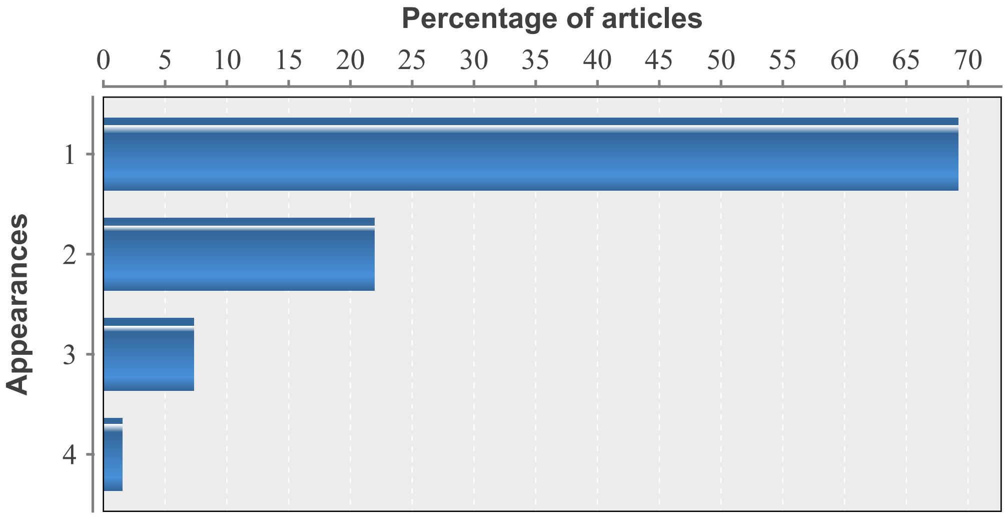


Figure 5.3: Article ratios based on appearances

$$\forall v \in V\,(indeg(v) > 0 \leftrightarrow auth(v) > 0) \tag{5.2}$$

All popular authority metrics like the undamped PageRank by Page et al. (1998), HITS by Kleinberg (1998) or the more blog-specific iRank by Adar et al. (2004b) comply to this condition and can be safely used with our methodology.

5.3.2 Networks from Data Aggregation

In the example data that we aggregated we have two separate social networks, the article network $G_{articles}$ with citation links and the blog network G_{blogs} with blogroll links, as defined below.

$$G_{articles} = (V_{articles}, E_{articles}) \tag{5.3}$$

$$G_{blogs} = (V_{blogs}, E_{blogs}) \tag{5.4}$$

There also exist links between articles and blogs due to the containment of each article in a specific blog. This is a *two-mode network* on its own (see Wasserman et al., 1994, pp. 39f.). Looking at all three networks at once, we have a construct which we decide to call a *hybrid network*, which is the starting point for our analyses. A simple example of such a network is given in Figure 5.4.

5.3.3 Original Article Authority

Using the plain network $G_{articles}$, we can compute the authority values for articles from this network. We define $auth_{article}(v)$ to be the *original article authority*, as derived from $G_{articles}$. However, the datasets show that articles are very sparsely connected in specific domains (see Table 5.1), and therefore we decide to use a more sophisticated method for calculating social authorities, which will give us more articles with non-zero authority values in the end.

For the determination of our social authorities we use a mutually dependent measure. The authority of an article depends on the authority of its blog, and the authority of a blog depends on the authorities of its articles. We present the derivation of the two measures in the following sections. We will use the original article authority later, to compare if the final social authority of articles indeed gives less non-zero authority values than the original article authority does.

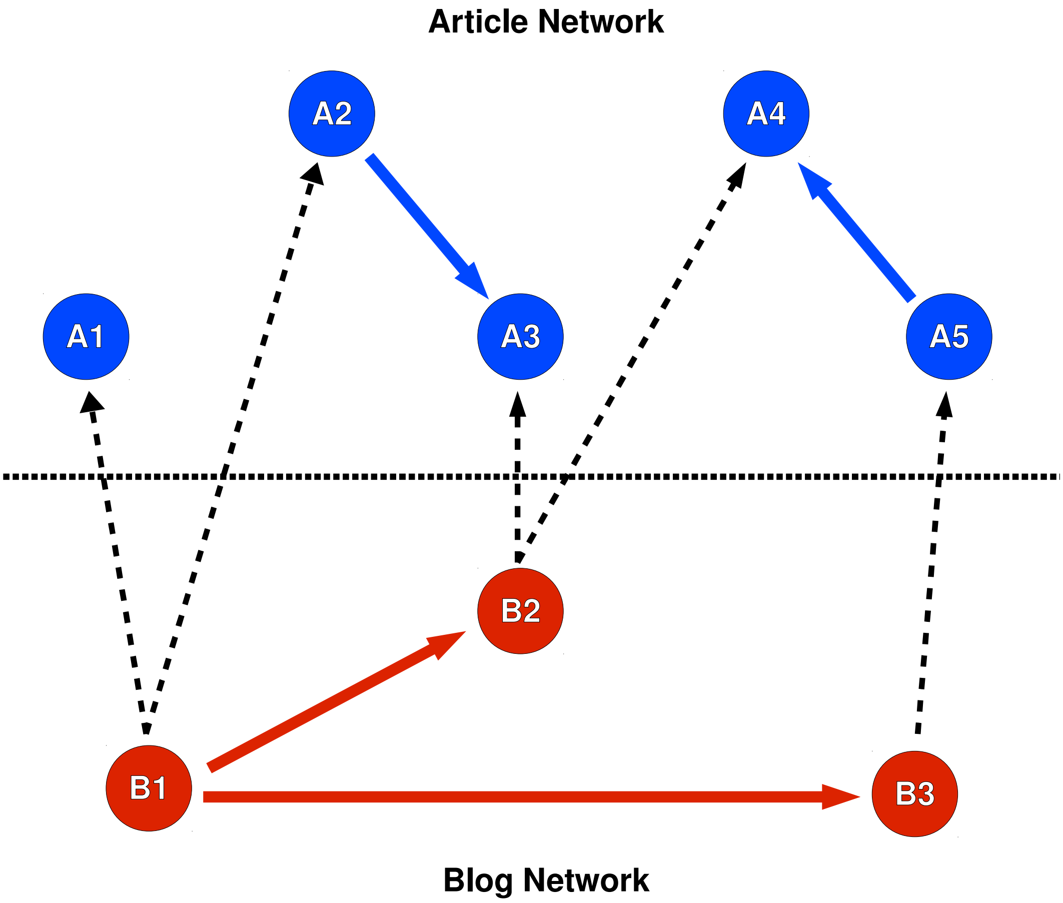


Figure 5.4: Example of a hybrid article/blog-network

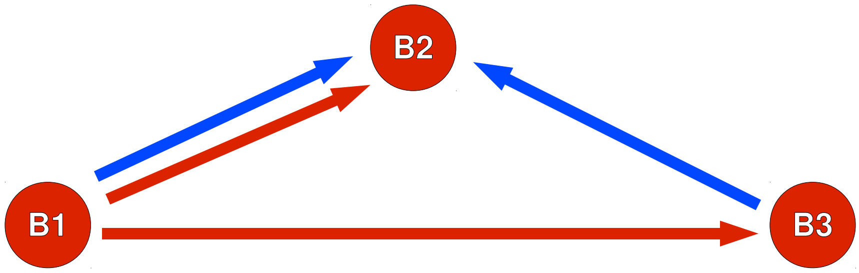


Figure 5.5: Blog multi graph derived from the hybrid network example

5.3.4 Blog Authority

To realise these mutually dependent metrics, we first map the article links into the blog network. This is possible with a function returning the hosting blog for a given article.

$$blog : V_{articles} \to V_{blogs} \tag{5.5}$$

So we can map each egde $(a_1, a_2) \in E_{articles}$ from the article network to an edge $(blog(a_1), blog(a_2))$ in the blog network with another function.

$$map : E_{articles} \to (V_{blogs} \times V_{blogs}) \tag{5.6}$$

As we have excluded links between articles of the same blog in the data aggregation, this cannot introduce loops in the new graph. However, this can introduce parallel edges, and hence turns our blog network into a *multi-graph* G_{multi}, i. e., a graph with multiple sets of differently typed or coloured edges (see Wasserman et al., 1994, pp. 145f.).

$$G_{multi} = (V_{blogs}, E_{blogs}, \{map(e), e \in E_{articles}\}) \tag{5.7}$$

Figure 5.5 illustrates the resulting multi-graph G_{multi} for the example hybrid network from Figure 5.4.

In order to compute the authorities of blogs with standard algorithms, which are not designed to operate on multi-graphs, we have to perform one last transformation, the unification of parallel edges.

All multi-edges are transformed to normal weighted edges, with a weight equivalent to the number of original edges in the multi-edge. This results in a weighted directed network, which is the most complex form that can be analysed by standard algorithms without major modifications. In the example multi-graph from Figure 5.5, the multi-edge $(B1, B2)$ would be transformed to an edge with a weight of 2, while the remaining two edges have a weight of 1 each.

As a result from this, we assume to have an authority function $auth_{blog}$, derived from the multi-graph transformed in such a way.

5.3.5 Combined Article Authority

We calculate the final article authority by combining two factors. The first one is the original article authority $auth_{article}$, as described in Section 5.3.3. The second factor

#	domain	$auth_{article} > 0$		$auth_{comb} > 0$		increase
1	Android G1	145	6%	510	20%	3.5
2	VW Golf	48	4%	165	12%	3.4
3	Toyota Hybrid Car	99	4%	343	13%	3.5
4	Angela Merkel	73	3%	670	31%	9.2
5	Robbie Williams	64	2%	663	18%	10.4
6	Fraunhofer	9	3%	20	6%	2.2
7	Google Wave	664	4%	2,920	18%	4.4

Table 5.2: Comparison of authoritative articles per domain

is the authority of the blog the article was published in, using the function $auth_{blog}$, as described in Secion 5.3.4. Additionally, we need a function $auth_{comb}$ that returns the final *combined authority* value in the interval $[0,1]$ for a given article a. In the simplest form, such a function looks as follows.

$$auth_{comb}(a) = \frac{auth_{article}(a) + auth_{blog}(blog(a))}{2} \tag{5.8}$$

Any other form of combination can be used with this methodology, but the suitability depends on the exact requirements of the final application.

With this procedure for the derivation of the combined article authority, we achieve to compute meaningful authority values for substantively more articles than by using the original article authority. We provide some empirical evidence for both claims in the following sections, i. e., for the increase of non-zero authoritative articles, and for the meaningfulness of the new measure.

5.3.6 Increase of Authoritative Articles

Table 5.2 lists the number of authoritative articles per domain for both metrics, when using the original article authority, and when using the combined article authority metric. Along with the absolute numbers we also provide the percentages with respect to all articles contained in the domain dataset. Based on these two numbers we present the increase factor, calculated as the number of authoritative articles using $auth_{comb}$ divided by the number of authoritative articles using $auth_{article}$.

The increase achieved by this method is between 2.2 and 10.4 in our example domains. It directly depends on the structure of the hybrid blog/article network. The better the blogs are connected and the more articles a blog contains on average, the higher the increase. What we cannot explain yet is the impact of the domain on that structure. In the domains number 2 and 3, which both deal with cars, we have, despite different sizes, a highly similar structure, and thus a nearly identical increase factor. This could be generally true for car domains, or coincidence, at least it calls for further investigation.

5.3.7 Evaluation of Combined Article Authority

We justified our combined authority measure from a theoretical network perspective, proposing that a blog's authority also influences an article's authority. We are able to cross-check it with the authorities expected from the number of appearances of an article in the different search engines (see Section 5.2.3). Figure 5.6 plots for each class of appearances the percentage of articles with that number of appearances, that have a non-zero authority value. The red squares joined by a red line refer to the original article authority measure $auth_{article}$, the blue circles joined by a blue line refer to the combined article authority measure $auth_{comb}$.

The original authority of an article is obviously highly correlated to its appearances (red line), the more appearances an article has, the higher the probability to have a non-zero authority. We can also see that our combined authority measure does not only increase the number of articles with authority, but does so in a highly consistent way with respect to the appearances. There is the same correlation to the number of appearances (blue line), which is a strong indicator for the meaningfulness of our method.

5.4 Including the Time Dimension

Since it is our third goal to monitor specific domains over a long period of time, we have to consider the time dimension as well. In SNA, *dynamics* is usually interpreted as evolving networks, in which new nodes and edges are added over time (Berger-Wolf & Saia, 2006; Skyrms & Pemantle, 2000). The intent is to identify patterns in this behaviour.

Both of our original networks are evolving networks as well, but for business intelligence we are not interested in patterns of behaviour in the first place. We are more interested in a measurement of *attention*, that reveals which articles are cited most often at a certain point of time.

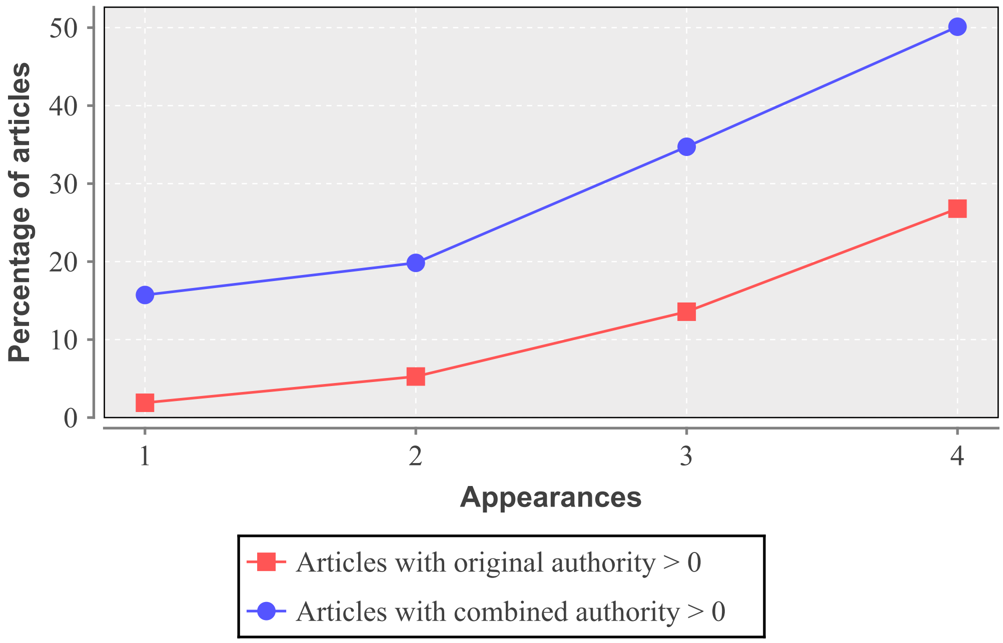


Figure 5.6: Original and combined article authorities based on appearances

The blog network with its blogroll links remains a static network in that case. Blogroll links do not change often, a regular update of each blog along with an update of the network is enough.

However, the article network is not only evolving, but a highly *time-sensitive* network. Each article has a timestamp, and a link between two articles is characterised by the time difference between its two end points.

During the monitoring of a domain, new articles are constantly added, new links are discovered and old links lose expressiveness for measuring the current attention. For example, an article that has been referenced a hundred times three months ago is not as relevant for the current situation of the domain as an article that has been cited twenty times in the last 48 hours.

In contrast, we have seen articles being referenced during our observation, which were published six months ago. Thus, these still get a good share of attention months after their publication, and this turns them to be relevant for the current point of time.

These different cases make clear that it is not enough to consider the articles of the last n days only, but that we need a more sophisticated measure instead to reflect the current attention an article receives.

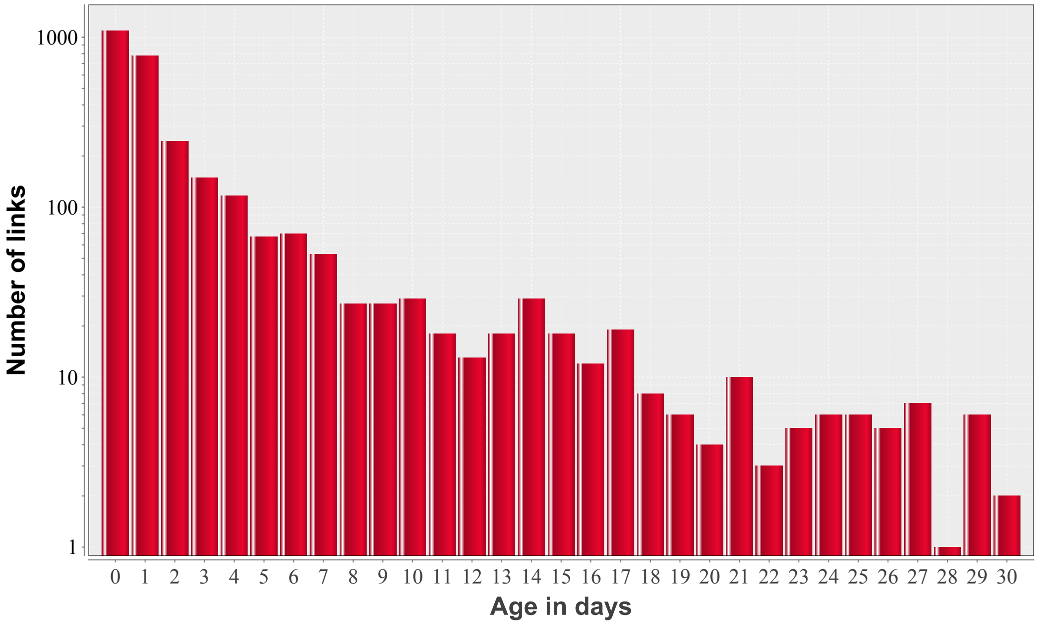


Figure 5.7: Distribution of link ages

5.4.1 Ages of Links

To analyse this phenomenon, we first look at the occurring time differences of links in our example datasets.

We first introduce some notations to handle this properly. Assume the current point of time is t_{now}. Given a function $time(a)$ that returns the point of time an article a was published at, and a subtraction operator that returns the time difference between two points of time, we can define a function *age* for a directed edge from article a_s to article a_t as follows.

$$age((a_s,a_t)) = time(a_s) - time(a_t) \tag{5.9}$$

Figure 5.7 illustrates the ages of the links found in our example datasets, rounded down to full days. Using a log-scale for the number of links of a certain age, we can observe that the vast majority of links to an article is set right after publication, but there are still a number of links set several days after publication. So there is good reason to respect this time difference when monitoring a specific domain over a long period of time.

5.4.2 A Time-Sensitive Network Model

Consequently we extend our methodology to consider the age of links for the determination of an article's attention. This will allow articles to have high attention values, even if they were published long time ago. We choose an approach of link decay realised via edge weights.

We can define a time-sensitive weight function for an edge $e = (a_s, a_t)$, which can be implemented in various ways. For simplicity, we present an example with a linear decay that is parameterisable with a maximum lifetime of Δt_{max} for an edge. The resulting weight function looks as follows.

$$weight((a_s, a_t)) = 1 - min\left(\frac{t_{now} - time(a_s)}{\Delta t_{max}}, 1\right) \tag{5.10}$$

With this weight function, a time-sensitive attention can be computed exactly like in a simple static weighted network. For the time-sensitive network, we define the indegree of a node a at the point of time t_{now} as the sum of the weights of all incoming links as follows.

$$indegree(a) = \sum_{s \in pre(a)} weight\,((s, a)) \tag{5.11}$$

Attention for Articles

Figure 5.8 illustrates the resulting effect for two articles. We have chosen two popular articles from domain number 7, which both have 31 incoming links in the static article network. The first one was published on the first day of October, the second one on the ninth day. With t_{now} moving from day 1 to day 31 we plot the current indegree of the articles with Δt_{max} set to 10 days.

While the two articles had the same indegree in the static network, it is now visible how the attention is spread over time. There are articles that receive a lot of attention for a short period of time, and articles that receive less attention, but for a longer period of time.

Thanks to a model based on a standard weighted directed network, we can calculate the attention of an article with any standard algorithm that is based on indegrees. We assume to have a metric $att(a)$ that returns the attention of an article for the current point of time calculated with a standard authority algorithm based on the indegrees of the time-sensitive network.

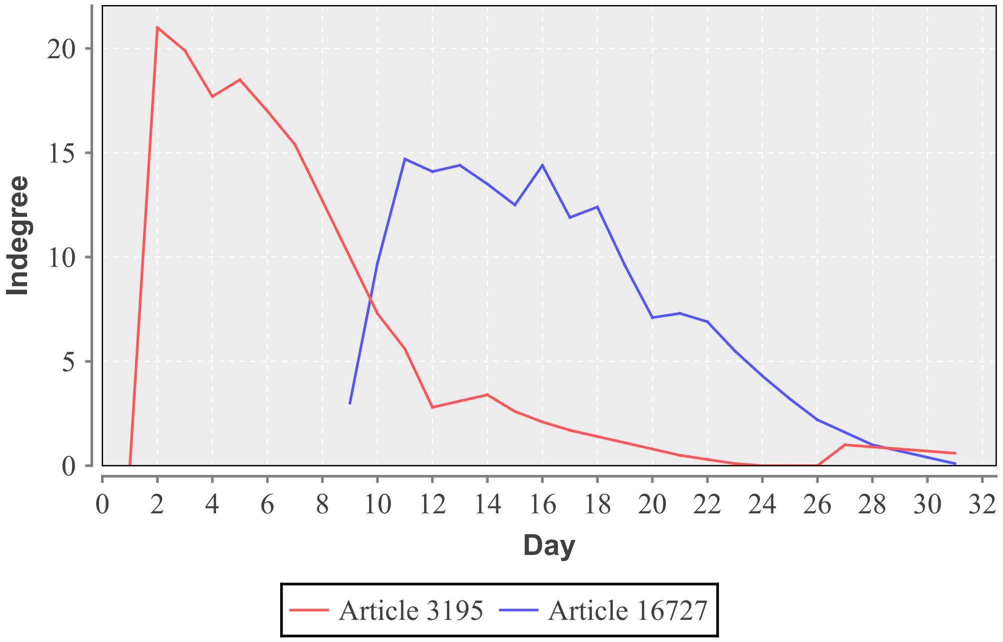


Figure 5.8: Indegrees over time for two selected articles

Attention for Blogs

With this model at hand, we can also provide an attention metric for blogs. Using the same mapping as for the calculation of blog authorities, we can construct a time-sensitive blog network. The fusion of multi-edges has to be done by adding up the weights of the mapped edges. Time-insensitive blogroll links have to be omitted for attention calculation. In this resulting weighted network, we can calculate attention values in the same way as done for the articles.

5.4.3 Time-Sensitive Relevance

With the new dimension of attention, the selection and ranking of presumably relevant articles at a certain point of time can be performed with a combination of article authority and attention. With authority only, we had to rely on articles around the given point of time to make a time-sensitive selection. Combined with attention, we can now consider the whole dataset and an according scoring function will find the currently relevant articles independently from their date of publication. In the simplest form, such a scoring function looks as follows.

$$relevance(a) = att(a) \cdot auth_{comb}(a) \tag{5.12}$$

Having the blog attention metric and the blog authority metric, these two can be combined to a time-sensitive relevance metric for blogs in the same way as done for the articles.

5.4.4 Enabling Retrospection

With the extensions from the last section, we are now capable to monitor blog article relevances over long periods of time. But currently, the calculation of metrics always refers to the current point of time t_{now}. Often it is interesting to retrieve metrics or make calculations for points of time in the past, especially when there is a demand for a comparison of the current state with states in the past.

We therefore extend our network structure with retrospection capabilities. This means that for any given point of time from the past, we want to enable all network calculations. In other words, we want the network to be easily revertable to any point of time t_{net} in a single instance. Duplicating network structures with snapshots and the like is considered too expensive and not expected to scale well.

We define the network structure being valid at a point of time t_{net} as follows.

$$G(t_{net}) = (V(t_{net}), E(t_{net})) \tag{5.13}$$

$$V(t_{net}) = \{a \in V \mid time(a) \leq t_{net}\} \tag{5.14}$$

$$E(t_{net}) = \{(s,t) \in E \mid time(s) \leq t_{net}\} \tag{5.15}$$

Such a network structure can be easily incorporated into a network data structure with a time attribute for the network. We have to override some basic methods to respect this attribute as defined in the formulas 5.14 and 5.15. These basic methods are the default network methods for getting nodes and edges, the node methods for getting incoming and outgoing edges, and the edge method for getting its weight.

With these changes, all subsequent methods based on these basic methods will behave in the correct way without further modifications. This is no problem in modern object oriented languages, and we implemented this in plugins for the Perl `SNA::Network` package located at CPAN[9].

[9] `http://www.cpan.org/`

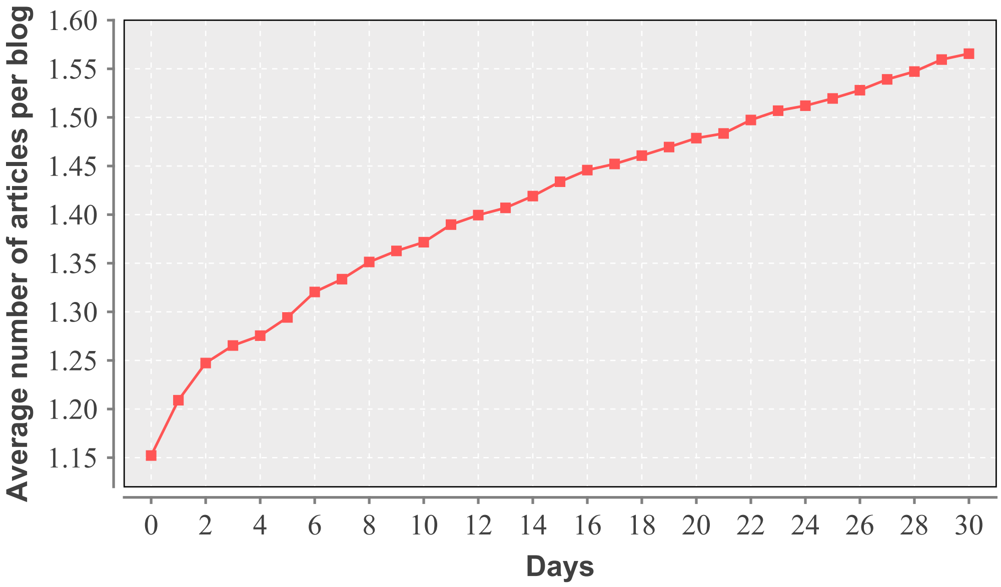


Figure 5.9: Average number of articles per blog over time

5.4.5 The Evolution of Domain Blogs over Time

While the blog articles are being aggregated over time, we see articles published in previously unknown blogs, as well as articles published in blogs already known from previous articles of the domain. To get an idea of this relation, Figure 5.9 plots the daily updated average number of articles per blog over all of our seven domains.

After a very steep increase in the first days, when most blogs of a domain are found with their first article, the curve is becoming less steep over time, which means that we see more and more articles published by the same blog.

Domain-Specific A-Lists

In consequence, this leads us to the idea that after some time of observation, the opinion-leading group of blogs for this specific domain should emerge in the structure of the blogroll network. This fact is of very high relevance in the given context of media monitoring for marketing or business intelligence, since it gives the user a hint where the blogspace of interest can be influenced in the most effective way. Such an influence could be the placement of advertisings, the distribution of comments, incentives for featured articles, and so on.

#	**domain**	$\lvert V_{conn}\rvert$	$\lvert V_{GC}\rvert$	$E[k_{max}]$	k_{max}	*members*
1	Android G1	279	231	1.00 ± 0.00	4	6
2	VW Golf	128	55	0.90 ± 0.31	2	10
3	Toyota Hybrid Car	215	103	0.73 ± 0.45	3	4
4	Angela Merkel	462	429	1.03 ± 0.18	3	20
5	Robbie Williams	385	194	1.00 ± 0.00	2	23
6	Fraunhofer	26	8	0.20 ± 0.41	0	8
7	Google Wave	3,066	2,603	1.00 ± 0.00	6	7

Table 5.3: Emergence of k-in-cores in the blog networks per domain

In order to detect the opinion-leading groups for a domain, we use the method of identifying *k-in-cores* presented in Chapter 4. For all of our blog networks we observe the emergence of a *giant component* after some time, as expected according to Molloy & Reed (1998). This is a weakly connected component that contains the majority of nodes in a graph, while the rest of the nodes is either isolated or connected in multiple small weakly connected components.

Table 5.3 lists the number of weakly connected nodes V_{conn} in the domain's blog network opposed to the number of nodes in the giant component $|V_{GC}|$. Futhermore it lists the highest value k_{max} for the detected *k-in-core*, which is a cohesive subgroup in which each member receives at least k incoming links from the other members of the *k-in-core*. The number of members is also listed in the table.

Assessing the Emerging Cores

We evaluate this by comparing the resulting k_{max} value with the expected value $E[k_{max}]$ from 30 randomly generated networks, which is also given in Table 5.3. These were generated based on the degree distribution of the blog network for each domain, as described in Section 2.3. We only look at the k_{max} value in this case, disregarding the complete properties of the In-CCS, because of the extreme sparsity of the networks in question, which leads to hardly visible sequences.

The emergence of unexpectedly high k value in our networks is a significant indicator for the presence of an authoritative subgroup according to the *A-List* theory, as outlined in Section 1.3. Defining a threshold Δt_{max} for active blogs, this method can constantly provide the end user with a list of the most influential blogs for the domain.

Looking at our largest example dataset, the “Google Wave” domain, we have a 6-in-core with 7 members. Remembering that the CCS is a nested measure, we look at the 5-in-core with 20 members, and find all the famous technology blogs in there, especially Engadget[10] and TechCrunch[11] for example, which confirms very clearly that this method is working very well in this case.

5.5 The Final Tool

The aggregation component and the authority/relevance measurements described in this chapter have been implemented and combined with a textual topic-clustering component by Schirru et al. (2010) and a sentiment analysis component by Pimenta et al. (2010). The result is the prototype of the SMM project, a web-based graphical interface realising the architecture presented in Section 5.1.

Domain Overview

Figure 5.10 shows the starting screen for the observation of the German star fashion designer “Karl Lagerfeld”. The upper part plots the volume of articles aggregated during the observation, as described in Section 5.2.

The lower part shows the detected topics in the selected time interval. This includes a list of the ten most characteristic keywords of the topic, the volume of the topic and the overall sentiment in the topic. When highlighting it, a key phrase of the topic along with a list of detected semantic entities is displayed in a popup window.

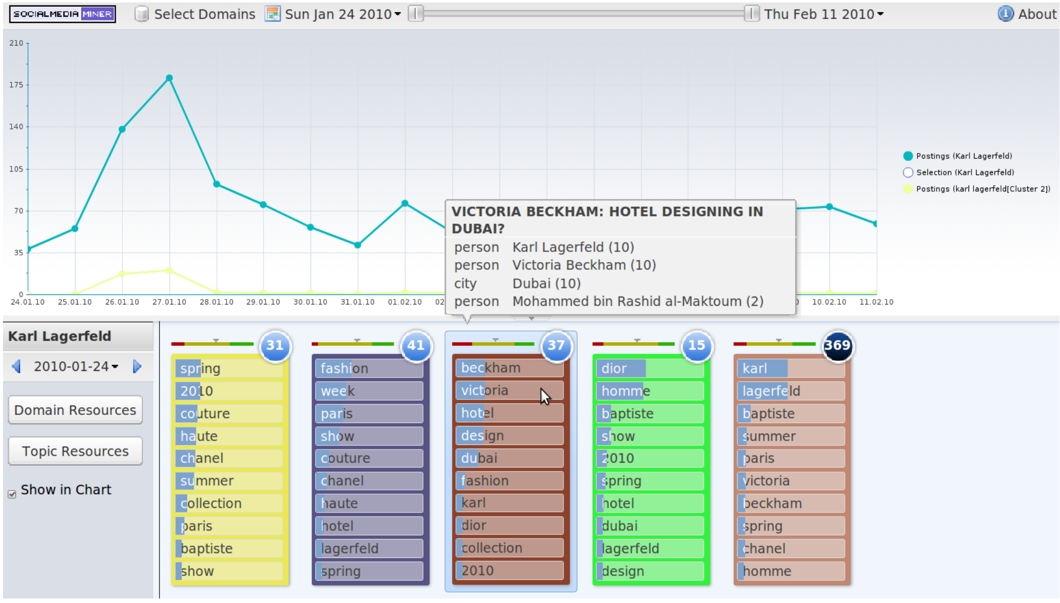
Articles Overview per Topic

When accessing a topic, for example the Dubai design hotel project planned together with Victoria Beckham, the interface lists all blog articles relevant to the topic ranked by authority as shown in Figure 5.11. Authority values have been computed using a variant of the HITS algorithm (Kleinberg, 1998), globally normalised to rounded numbers between 0 and 100. Thanks to the combined article authority, we can list several really authoritative articles at the top of each topic in all cases.

Here the user can select the articles of interest from the left side, and see a thumbnail of the article page, some meta data and the full text on the right side.

[10] http://www.engadget.com

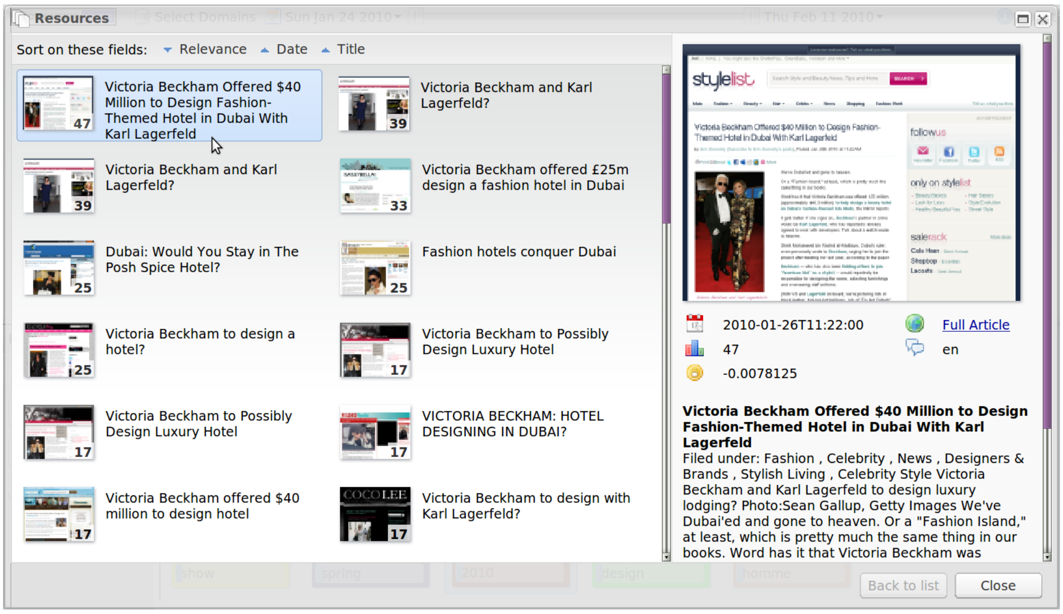
[11] http://www.techcrunch.com



VICTORIA BECKHAM: HOTEL DESIGNING IN DUBAI?

person	Karl Lagerfeld (10)
person	Victoria Beckham (10)
city	Dubai (10)
person	Mohammed bin Rashid al-Maktoum (2)

Figure 5.10: SMM main view for the domain “Karl Lagerfeld”



Victoria Beckham Offered $40 Million to Design Fashion-Themed Hotel in Dubai With Karl Lagerfeld 47	Victoria Beckham and Karl Lagerfeld? 39
Victoria Beckham and Karl Lagerfeld? 39	Victoria Beckham offered £25m design a fashion hotel in Dubai 33
Dubai: Would You Stay in The Posh Spice Hotel? 25	Fashion hotels conquer Dubai 25
Victoria Beckham to design a hotel? 25	Victoria Beckham to Possibly Design Luxury Hotel 17
Victoria Beckham to Possibly Design Luxury Hotel 17	VICTORIA BECKHAM: HOTEL DESIGNING IN DUBAI? 17
Victoria Beckham offered $40 million to design hotel 17	Victoria Beckham to design with Karl Lagerfeld? 17

2010-01-26T11:22:00	Full Article
47	en
-0.0078125	

Victoria Beckham Offered $40 Million to Design Fashion-Themed Hotel in Dubai With Karl Lagerfeld
Filed under: Fashion , Celebrity , News , Designers & Brands , Stylish Living , Celebrity Style Victoria Beckham and Karl Lagerfeld to design luxury lodging? Photo:Sean Gallup, Getty Images We've Dubai'ed and gone to heaven. Or a "Fashion Island," at least, which is pretty much the same thing in our books. Word has it that Victoria Beckham was

Figure 5.11: Article list for the topic around the “Dubai design hotel”

CHAPTER 6

Conclusion

We conclude this thesis by summarising the important findings of the previous chapters and their relations among each other. We then discuss the implications of our research for the scientific field and its applicability, as well as problems that remained open. Finally, we present some thoughts about potential future work that is related to this research, or questions that are raised by open problems.

6.1 Summary

After a detailed explanation of the two foundational concepts of this thesis in Chapter 1, the blogosphere and the scientific field of SNA, we presenred the central methods for evaluating our work in Chapter 2, namely the evaluation method by comparison with random networks, and GRAMs.

In Chapter 3 we presented our blog datasets that we used for the A-List detection analyses. By having similar datasets of six different languages, we gained the opportunity to cross-check our later results, which clearly benefits the reliability of the later findings.

In Chapter 4, the main chapter of this thesis, we engaged our first research question, how to reliably detect the elite group of A-List blogs. Based on the literature, we decided to adhere to the core/periphery model by Borgatti & Everett, and used a suitable variant of Seidman's robust concept of *k*-cores to approximate it efficiently. This approximation has been implemented with the scalable in-core algorithm. Applying this algorithm, we instantly accomplished very good results for two of our six datasets.

A critical analysis of the other results revealed that there were still some open issues with large highly cohesive non-authoritative subgroups in these four datasets. In an attempt to work around this problem, we extensively studied the usage of existing community identification algorithms for our datasets, and suggested a first approach to filter the networks using this knowledge about community structure. This approach was experimentally applied to the Italian dataset, and provided good results for the A-List detection according to the core/periphery model.

In Chapter 5, we investigated our second research question, where the measurement of authority and relevance of blogs is required in a practical scenario. In the SMM project, a monitoring application has been developed, which intelligently aggregates blog articles for different domains, and enables the user to access relevant articles of current hot topics. We showed that our meta search is extremely effective for achieving a good coverage, and that our derivation of combined authority from article citations and blogroll entries is effective, sound, and scalable with respect to a long observation time.

Furthermore, using the knowledge from Chapter 4, we were able to quickly identify the most important blogs for a domain after an initial period of observation.

6.2 Discussion

Despite the relatively good results, there are some issues that remain problematic, and would need further investigation, if possible at all.

The blog datasets of different languages sampled in Chapter 3 are the starting point for all the analyses conducted in Chapter 4. So every shortcoming here directly affects the results there, and indeed, we have an important shortcoming here to be noticed. The sampled blogs are only a small excerpt of the language's blogosphere, containing almost certainly all the authoritative blogs, but not the huge long tail. This long tail however is important for the final judgement of the quality of the detected A-List. We are convinced that our dataset is complete enough for strong statements here, but especially Section 4.7 revealed, that it becomes at least very difficult to find the right parameters for sparsification, if not even impossible to fix the problematic dataset without a large enough share of the long tail.

It also needs to be considered that our goal is strictly limited to match the structural core/periphery model of Borgatti & Everett. Thus we depend on its soundness. Having shown the perfect correlation between the formal A-List characteristics from the literature and the definition of the core/periphery model, we are convinced that this decision is scientifically sound. But it cannot be guaranteed that the structurally detected cores indeed match with the real A-Lists. This could be engaged by a thorough qualitative social evaluation of the blogosphere, but even this result would be an uncertain qualitative one.

More or less the same applies to the application in the SMM project. We adhered to the rich findings of related work and followed the recommended methodology of the field, but a final proof for the correctness of the measured relevancies cannot be given, just like described above. In this case however, we have some positive feedback from project partners and customers, who were very satisfied with the results.

6.3 Outlook

The results of this thesis can be directly applied to blogosphere analysis, and already have been, as outlined in Section 5.5. When trying to generalise the results, the transferability to similarly structured data and problems is certainly given. But these are highly specific problem solutions, required only in social media applications, which additionally need a good parameterisation for some steps. That is why we do not expect a big impact here.

The general scientifc impact is much more interesting in our opinion. First, the evaluation of measured network results or behaviour is always a crucial point. In this thesis we have evaluated our network structures by comparison with random networks, which were generated by latest state-of-the-art MCMC algorithms. The research around random networks is often conducted by mathematicians and physicists, and there are hardly examples where this is practically applied. We demonstrated a very useful application of random network generation, and hope that this will inspire other researchers to apply the same methodology in the future, since the insights gained here have been very substantial.

Second, we introduced the visualisation method with GRAMs in Section 2.4. This has been an enormous help in understanding large partitioned networks, not only in the apparent context of community identification, but also for the judgement of more subtle partitionings like a CCS. Especially our open problem of cluster quality measurement was easy to solve with this way of thinking, as presented in Section 4.6.2. The method is relatively easy to implement and very scalable thanks to the parameterisation possibilities. We hope to see some more usage of it in future SNA research.

Acronyms

API Application Programming Interface 27

CCS Core Collapse Sequence 34

GRAM Group Adjacency Matrix

HTTP Hypertext Transfer Protocol 26

MCMC Markov Chain Monte Carlo iii

OSN Online Social Network 3

SMM Social Media Miner 67

SNA Social Network Analysis iii

URL Uniform Resource Locator 26

WWW World Wide Web 3

Bibliography

Adamic, L. A. & Glance, N. (2005). The political blogosphere and the 2004 u.s. election: divided they blog. In *Proceedings of the 3rd International Workshop on Link Discovery (LinkKDD)* (pp. 36–43). 7

Adar, E., Zhang, L., Adamic, L., & Lukose, R. (2004a). Implicit structure and the dynamics of blogspace. In *Workshop on the Weblogging Ecosystem*. 7

Adar, E., Zhang, L., Adamic, L. A., & Lukose, R. M. (2004b). Implicit structure and the dynamics of blogspace. In *Workshop on the Weblogging Ecosystem, WWW2004* New York, NY. 74

Alon, U. (2007). Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6), 450–461. 14

Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley, 1st edition. 56

Bansal, N. & Koudas, N. (2007). Searching the blogosphere. In *Proceedings of the 10th International Workshop on Web and. Databases, WebDB 2007* Beijing, China. 7

Barabási, A.-L. (2003). *Linked - how everything is connected to everything else and what it means for business, science, and everyday life*. Plume. 2

Barabasi, A. L. & Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286, 509–512. 15, 37

Batagelj, V. & Brandes, U. (2005). Efficient generation of large random networks. *Physical Review E*, 71(3), 036113. 18

Batagelj, V. & Zaversnik, M. (2002). Generalized cores. *CoRR*, cs.DS/0202039. 37

Batagelj, V. & Zaversnik, M. (2003). An o(m) algorithm for cores decomposition of networks. *CoRR*, cs.DS/0310049. 37

Berger-Wolf, T. Y. & Saia, J. (2006). A framework for analysis of dynamic social networks. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 523–528). New York, NY, USA: ACM. 78

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), P10008+. 55

Blood, R. (2002). *The Weblog Handbook: Practical Advice on Creating and Maintaining Your Blog*. Perseus Books. 4, 7

Bollobas, B. (1985). *Random Graphs*. London: Academic Press. 17

Borgatti, S. P. & Everett, M. G. (1999). Models of core/periphery structures. *Social Networks*, 21, 375–395. i, 33, 42, 89, 90

Branckaute, F. (2010). State of the blogosphere in 2010. http://www.blogherald.com/2010/09/20/state-of-the-blogosphere-in-2010/. 6

Brandes, U. & Erlebach, T. (2005). *Network Analysis: Methodological Foundations*. Springer. 4

Chau, M. & XU, J. (2007). Mining communities and their relationships in blogs: A study of online hate groups. *International Journal of Human-Computer Studies*, 65(1), 57–70. 7

Chin, A. & Chignell, M. (2006). A social hypertext model for finding community in blogs. In *Proceedings of the seventeenth conference on Hypertext and hypermedia*, HYPERTEXT '06 (pp. 11–22). New York, NY, USA: ACM. 7

Delwiche, A. (2005). Agenda-setting, opinion leadership, and the world of web logs. *First Monday*, 10(12). 7

Doreian, P. & Woodard, K. L. (1992). Fixed list versus snowball selection of social networks. *Social Science Research*, 21(2), 216 – 233. 26

Doreian, P. & Woodard, K. L. (1994). Defining and locating cores and boundaries of social networks. *Social Networks*, 16(4), 267 – 293. 34, 35

Dunbar, R. (1993). Coevolution of neocortex size, group size and language in humans. *Behavioral and Brain Sciences*, 16(4), 681–735. 14

Erdos, P. & Renyi, A. (1959). On random graphs. *Publ. Math. Debrecen*, 6, 290. 16, 17

Faloutsos, M., Faloutsos, P., & Faloutsos, C. (1999). On power-law relationships of the internet topology. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication* (pp. 251–262).: ACM. 15

Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3-5), 75 – 174. 53, 54

Freeman, L. C. (2004). *The Development of Social Network Analysis: A Study in the Sociology of Science*. Empirical Press. 2

Goetz, M., Leskovec, J., Mcglohon, M., & Faloutsos, C. (2009). Modeling blog dynamics. In *International Conference on Weblogs and Social Media*. 69

Gruhl, D., Guha, R., Liben-Nowell, D., & Tomkins, A. (2004). Information diffusion through blogspace. In *WWW '04: Proceedings of the 13th international conference on World Wide Web* (pp. 491–501). New York, NY, USA: ACM Press. 7

Herring, S. C., Kouper, I., Paolillo, J. C., Scheidt, L. A., Tyworth, M., Welsch, P., Wright, E., & Yu, N. (2005). Conversations in the blogosphere: An analysis "from the bottom up". In *Proceedings of the 38th HICSS* (pp. 107.2).: IEEE. 7, 9, 56, 70

Herring, S. C., Scheidt, L., Bonus, S., & Wright, E. (2004). Bridging the gap: A genre analysis of weblogs. In *Proceedings of the 37th Hawaii International Conference on System Sciences*. 4

Kleinberg, J. M. (1998). Authoritative Sources in a Hyperlinked Environment. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms* (pp. 668–677).: AAAI Press. 74, 86

Krishnamurthy, S. (2002). *The Multidimensionality of Blog Conversations: The Virtual Enactment of September 11*, volume 3. Internet Research 3.0. 5, 101

Kumar, R., Novak, J., Raghavan, P., & Tomkins, A. (2004). Structure and evolution of blogspace. *Communications of the ACM*, 47, 35–39. 7

Kumar, R., Novak, J., Raghavan, P., & Tomkins, A. (2005). On the bursty evolution of blogspace. *World Wide Web*, 8(2), 159–178. 7, 69

Leskovec, J., Lang, K. J., Dasgupta, A., & Mahoney, M. W. (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1), 29–123. 54, 59

Marlow, C. (2004). Audience, structure and authority in the weblog community. In *Proceedings of the International Communication Association Conference*. 7

Milo, R., Kashtan, N., Itzkovitz, S., Newman, M. E. J., & Alon, U. (2003). On the uniform generation of random graphs with prescribed degree sequences. *Arxiv preprint cond-mat/0312028*. 18, 19, 20, 21

Molloy, M. & Reed, B. (1998). The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7, 295–305. 85

Newman, M., Watts, D., & Strogatz, S. (2002). Random graph models of social networks. *Proceedings of the National Academy of Sciences USA*, 99, 2566–2572. 17

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45, 167–256. 4, 13, 15, 18, 53

Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3), 036104+. 54

O'Reilly, T. (2005). What is web 2.0. design patterns and business models for the next generation of software. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html. 3, 6

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). *The PageRank Citation Ranking: Bringing Order to the Web*. Technical report, Stanford University. 3, 71, 74

Park, D. (2004). From many, a few: Intellectual authority and strategic positioning in the coverage of, and self-descriptions of, the "big four" weblogs. In *Proceedings of the International Communication Association Conference*. 7

Pimenta, F., Obradovic, D., Schirru, R., Baumann, S., & Dengel, A. (2010). Automatic sentiment monitoring of specific topics in the blogosphere. In *Workshop on Dynamic Networks and Knowledge Discovery (DyNaK 2010)*. 86

Rueger, C. (2010). *Community Identification in International Weblogs*. Master thesis, University of Kaiserslautern. 55

Schirru, R., Obradovic, D., Baumann, S., & Wortmann, P. (2010). Domain-specific identification of topics and trends in the blogosphere. In P. Perner (Ed.), *Advances in Data Mining. Applications and Theoretical Aspects. Industrial Conference on Data Mining (ICDM-10)*, volume 6171 of *LNAI* (pp. 490–504).: Springer. 86

Scott, J. (2000). *Social Network Analysis: A Handbook*. SAGE Publications. 2

Seidman, S. B. (1983). Network structure and minimum degree. *Social Networks*, 5, 269–287. i, 34

Shirky, C. (2003). Power laws, weblogs, and inequality. http://shirky.com/writings/powerlaw_weblog.html. 7, 15, 28

Skyrms, B. & Pemantle, R. (2000). A dynamic model of social network formation. *Proceedings of the Natinal Academy of Sciences, USA.*, 97(16), 9340–9346. 78

Snijders, T. (1991). Enumeration and simulation methods for 0-1 matrices with given marginals. *Psychometrika*, 56(3), 397–417. 20

Sobel, J. (2010). State of the blogosphere 2010. http://technorati.com/blogging/article/state-of-the-blogosphere-2010-introduction/. 6, 67

Tricas, F., Ruiz, V., & Merelo, J. J. (2003). Do we live in an small world? measuring the spanish–speaking blogosphere. In *Proceedings of the BlogTalk Conference*. 15

Ulicny, B. & Baclawski, K. (2007). New metrics for newsblog credibility. In *In Proceedings International Conference on Weblogs and Social Media* Colorado, USA. 7

Viger, F. & Latapy, M. (2005). Efficient and simple generation of random simple connected graphs with prescribed degree sequence. In *Proceedings of the 11th international conference on Computing and Combinatorics*, volume 3595 of *LNCS* (pp. 440–449).: Springer. 19, 20, 21

Wasserman, S., Faust, K., & Iacobucci, D. (1994). *Social Network Analysis : Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press. 4, 74, 76

Wasserman, S. & Robins, G. L. (2005). An introduction to random graphs, dependence graphs, and p*. In P. J. Carrington, J. Scott, & S. Wasserman (Eds.), *Models and methods in social network analysis* (pp. 148–161). Cambridge University Press. 17

Watts, D. & Strogatz, S. (1998). Collective dynamics of small-world networks. *Nature*, (393), 440–442. 16, 17

Wortmann, P. (2009). *Topic-Based Blog Article Search for Trend Detection*. Project thesis, University of Kaiserslautern. 70

Zhou, Y. & Davis, J. (2006). Community discovery and analysis in blogspace. In *Proceedings of the 15th international conference on World Wide Web* (pp. 1017–1018).: ACM. 7, 56

List of Figures

List of Tables

Publications by the Author

The following list gives a chronological overview of accepted peer-reviewed scientific publications directly related to this thesis, which are authored or substantially co-authored by the author of this thesis.

1. Darko Obradović, Stephan Baumann. “Identifying and Analysing Germany’s Top Blogs”. In *Proceedings of the 31st German Conference on Artificial Intelligence (KI 2008)*, Kaiserslautern, Germany, pp. 111–118, Springer, September 2008.

2. Darko Obradović, Stephan Baumann. “A Journey to the Core of the Blogosphere”. In *Proceedings of the International Conference on Advances in Social Network Analysis and Mining (ASONAM 2009)*, Athens, Greece, pp. 1–6, IEEE, July 2009.
   (2nd Best Paper Award)

3. Darko Obradović, Rafael Schirru, Stephan Baumann, Andreas Dengel. “Social Media Miner – Automatische Erkennung von Trends im Web 2.0” (in German). In *DOK.magazin*, no. 2-10, pp. 76–78, good source publishing, June 2010.

4. Darko Obradović, Stephan Baumann. “A Journey to the Core of the Blogosphere” (extended version). In *From Sociology to Computing in Social Networks*, Nasrullah Memon, Reda Alhajj (Eds.), Lecture Notes in Social Networks (LNSN), vol. 1, pp. 25–43, Springer, July 2010.

5. Rafael Schirru, Darko Obradović, Stephan Baumann, Peter Wortmann. “Domain-Specific Identification of Topics and Trends in the Blogosphere”. In *Proceedings of the 10th Industrial Conference on Data Mining (ICDM 2010)*, Berlin, Germany, pp. 490–504, Springer, July 2010.

6. Darko Obradović, Stephan Baumann, Andreas Dengel. “A Social Network Analysis and Mining Methodology for the Monitoring of Specific Domains in the Blogosphere”. In *Proceedings of the International Conference on Advances in Social Network Analysis and Mining (ASONAM 2010)*, Odense, Denmark, pp. 1–8, IEEE, August 2010.
   (1st Best Paper Award)

7. Fernanda Pimenta, Darko Obradović, Rafael Schirru, Stephan Baumann, Andreas Dengel. “Automatic Sentiment Monitoring of Specific Topics in the Blogosphere”. *Workshop on Dynamic Networks and Knowledge Discovery (DyNaK 2010)*, Barcelona, Spain, published online, September 2010.

8. Darko Obradović, Wolfgang Schlauch. “Zuverlässige und Schnelle Erzeugung von Zufallsnetzwerken für Evaluationszwecke” (in German). In *Proceedings of the Young Researcher Symposium 2011 (YRS 2011)*, Kaiserslautern, Germany, Center for Mathematical and Computational Modelling, University of Kaiserslautern, February 2011.

9. Darko Obradović, Christoph Rueger, Andreas Dengel. “Core/Periphery Structure versus Clustering in International Weblogs”. In *Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN 2011)*, Salamanca, Spain, pp. 1–6, IEEE, October 2011.

10. Darko Obradović, Fernanda Pimenta, Andreas Dengel. “Mining Shared Social Media Links to Support Clustering of Blog Articles”. In *Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN 2011)*, Salamanca, Spain, pp. 181–184, IEEE, October 2011.

11. Darko Obradović. “Weblogs im Internationalen Vergleich – Meinungsführer und Gruppenbildung” (in German). In *Knoten und Kanten 2.0 – Soziale Netzwerkanalyse in Medienforschung und Kulturanthropologie*, Markus Gamper, Linda Reschke, Michael Schönhuth (Eds.), pp. 163–184, transcript, April 2012.

12. Darko Obradović, Stephan Baumann, Andreas Dengel. “A Social Network Analysis and Mining Methodology for the Monitoring of Specific Domains in the Blogosphere” (extended version). *Social Network Analysis and Mining*, Springer, accepted for publication.

Curriculum Vitae

Personal

Name	Darko Obradović
Date of Birth	November 28th 1980
Place of Birth	Kaiserslautern, Germany
Nationality	Croatian
Marital Status	married, no children
Address	DFKI GmbH Trippstadter Straße 122 67663 Kaiserslautern Germany
Phone	+49 (631) 20575 1510
E-Mail	`darko.obradovic@dfki.de`
WWW	`http://www.dfki.de/~obradovic`

Languages

native	German, Croatian
fluent	English, French, Spanish
basic	Italian, Latinum

Education

2007-2012 Doctoral student at the German Research Center for Artificial Intelligence in Kaiserslautern, Germany under supervision of Prof. Dr. Prof. h.c. Andreas Dengel, finished with a Dr. rer. nat. (corresponds to a Ph.D.), Grade “magna cum laude”

07/2010 Participant at the Lipari School on Computational Complex Systems “Social Networks” by the Jacob T. Schwartz International School for Scientific Research, lectured by the Profs. C. Faloutsos, R. Kumar, D. Helbig and A. Barrat

2000-2006 Computer Science studies at University of Kaiserslautern with emphasis on Software Engineering and Artificial Intelligence, finished with a Dipl.-Inf. (corresponds to M.A. Sc.), Grade 1.7 (max. 1.0)

1991-2000 Gymnasium an der Burgstraße (grammar school) in Kaiserslautern, Germany, with emphasis on Mathematics, Politics and French, finished with Abitur (A-Levels), Grade 1.5 (max. 1.0)

06/1999 Invited participant at the Summer School “Mathematical Modelling” of the Technomathematics Group of the University of Kaiserslautern at Pfalzakademie Lambrecht

1987-1991 Grundschule Schillerschule (primary school) in Kaiserslautern, Germany

Work Experience

since 04/2007 Researcher at the German Research Center for Artificial Intelligence (DFKI), department of Knowledge Management, Kaiserslautern, Germany

2001-2006 University of Kaiserslautern, Faculty for Computer Science, teaching assistant for lectures in software development

Awards & Prizes

08/2010 1st Best Paper Award at ASONAM 2010 conference

07/2009 2nd Best Paper Award at ASONAM 2009 conference

10/2004 Best rated teaching assistant in summer term 2004 of the Faculty for Computer Science of the University of Kaiserslautern

02/2000 Special prize of the VR Bank Südpfalz at “Jugend Forscht” (Youth Researchers) regional competition in Mathematics/Computer Science

02/1999 2nd place at “Jugend Forscht” (Youth Researchers) regional competition in Mathematics/Computer Science