Networks of Names – Obtaining Lombardi's Narrative Structures by Combining Visual Analytics and Language Technology

Networks of Names – Gewinnung von Lombardis „Narrative Structures" durch Kombination von Visual Analytics und Sprachtechnologie
Master-Thesis von Artjom Kochtchi
Oktober 2013

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Informatik
Fachgebiet Sprachtechnologie

Networks of Names – Obtaining Lombardi's Narrative Structures by Combining Visual Analytics and Language Technology
Networks of Names – Gewinnung von Lombardis „Narrative Structures" durch Kombination von Visual Analytics und Sprachtechnologie

Vorgelegte Master-Thesis von Artjom Kochtchi

1. Gutachten: Prof. Dr. Chris Biemann
2. Gutachten: Dr. Tatiana von Landesberger

Tag der Einreichung:

Erklärung zur Master-Thesis

Hiermit versichere ich, die vorliegende Master-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 15. Oktober 2013

(A. Kochtchi)

Abstract

Investigating relationships between people and organizations by reading newspapers can be time-consuming and hard to manage for humans due to the large volume of data. We design and develop Networks of Names, an interactive system that provides an accessible way for visual exploration of social networks automatically extracted from newspapers.

We integrate several methods from information visualization, language technology, as well as other fields of computer science, and implement a classifier of non-taxonomic relationships that builds upon user interaction.

By conducting an exploratory user experiment we show that a visualization (compared to a text-based environment) may have an impact on how users label semantic relationships, and that our classifier is capable to produce reasonable results and yield interesting lexico-syntactic patterns.

Zusammenfassung

Das Recherchieren von Beziehungen zwischen Personen und Organisationen kann für Menschen sehr zeitaufwändig und aufgrund der großen Datenmenge schwer zu handhaben sein. Wir entwerfen und entwickeln Networks of Names, ein interaktives System, das eine zugängliche Möglichkeit zur visuellen Exploration automatisch aus Zeitungen extrahierter sozialer Netzwerke bietet.

Dazu integrieren wir verschiedene Methoden aus Informationsvisualisierung, Sprachtechnologie sowie anderen Themenfeldern der Informatik, und implementieren einen Klassifizierer für nicht-taxonomische Beziehungen auf Basis von Benutzerinteraktion.

Mithilfe eines explorativen Benutzerexperiments zeigen wir, dass eine Visualisierung (verglichen mit einer textbasierten Umgebung) einen Einfluss darauf haben könnte, wie Benutzer semantische Beziehungen bennenen, und dass unser Klassifizier in der Lage ist, angemessene Ergebnisse und interessante lexiko-syntatktische Muster zu erzeugen.

Contents

1 Introduction

Networks of Names is a visual interactive system that enables users to explore, investigate, and present relationships between people and organizations of public interest.

In this introductory chapter we explain the motivation and aims of this work (Section 1.1), classify it in terms of common research areas of computer science (Section 1.2), and explore the current state-of-the-art research of related fields (Section 1.3). We summarize our contributions in Section 1.4 and give an overview over this work in Section 1.5.

1.1 Motivation

The idea for Networks of Names originates in the need of individuals to explore and understand the relationships between people and organizations that interact and influence their surroundings, the society, and public policy. A typical example of such relationships is the involvement of politicians with each other as well as with corporations and their representatives. Another, rather different, use case is the understanding of relationships of and interaction between celebrities. Such connections can be of interest to journalists for professional or to private persons in general for political reasons or out of personal interest.

Relationships between people and organizations are numerous and their structure can be complex, so that certain connections can remain hidden or intransparent if not investigated explicitly and closely. Applying methods from computer science to this task could significantly improve its feasibility, simplify access to this kind of information, and enable the possibility for public scrutiny (especially in matters of politics and public policy).

Narrative Structures

The idea of analysing the involvement of public figures bears similarities to the work of *Mark Lombardi*, who in the late 20th century researched political and financial scandals [55]. Lombardi compiled and organized his data from reports by reputable news organizations and managed it in a collection of handwritten cards. Being a conceptional artist, he eventually presented his results in flow diagrams he coined *Narrative Structures*[1], because they could “narrate” the data in his database in a form more accessible and aesthetically pleasing to humans.

Automation

Following Lombardi’s approach, we aim to extract and visualize social network information from newspapers. Unlike Lombardi, however, Networks of Names performs the time-consuming task of collecting, organizing, and visualizing information automatically.

According to the BDZV[2], there are more than 350 daily newspapers published in Germany alone [10]. While reading and understanding single or several of those articles is a common task for humans, processing all newspaper articles of even a single day is clearly unmanageable.

For computers, on the other hand, processing big amounts of data quickly is not a problem in general. However, extracting insightful information and deriving knowledge from newspaper articles, a characteristic human capability, remains a challenging automation task.

Visual Analytics and Language Technology

The respective strengths of humans and computers lead us to an interactive approach that maximises capabilities of the system by combining human and computer capabilities to their best.

This is generally referred to as *visual analytics*, a field that can be defined as “the science of analytical reasoning facilitated by interactive visual interfaces” [58]. Being an interdisciplinary field of research, visual analytics commonly adopts techniques from other fields of computer science.

[1] Examples of Lombardi’s narrative structures can be found at `http://socks-studio.com/2012/08/22/mark-lombardi/`.

[2] Bundesverband Deutscher Zeitungsverleger e. V., `http://www.bdzv.de/`

In our case, we employ several methods from *language technology* to mine and process relationship data from newspapers. The visual analytics approach allows us to exploit user input to improve on algorithmic methods in a way that would be impossible without user interaction.

Our academic interests are 1) to integrate and adapt current state-of-the-art research to make information accessible that would otherwise be obfuscated by the amount of data and its presentation and 2) to create a system that demonstrates the practical capabilities of automatic methods, thereby popularizing their acceptance and use.

1.2 Research Area

The automation at the core of Networks of Names poses a number of algorithmic problems, touching on several fields of computer science.

Algorithmic Problems

In order to acquire data from newspaper texts, we need to engage in several activities from the field of *information retrieval*. More specifically, for extraction of social connections contained in natural language text, relevant problems are *named entity recognition*, i. e. recognizing people and organizations in natural language texts, and *relationship extraction*, i. e. understanding the relationships[3] between those named entities.

The result of the extraction of names and relationships is conceptually a graph, where vertices represent named entities and edges represent type and strength of relationships. *Graph algorithms* are required to enable searching and exploring the graph. Finally, to visualize a graph, vertices and edges need to be layed out in the plane (*graph drawing*).

The resulting network diagram and its presentation can be subjected to several quality criteria, most notably in that it shall support *visual analytics* by providing an interactive interface and facilitate human reasoning about the underlying data.

Additionally, we employ methods for determining *document similarity*, *clustering*, and the discovery and labelling of *non-taxonomic relations* to lower the burden of manual data evaluation on users.

The Visual Analytics Process

This work can be placed in the intersection between information visualization and language technology, with visual analytics of large text corpora as the overarching theme. Figure 1 shows the visual analytics process that illustrates how visual analytics "combines automatic and visual analysis methods with a tight coupling through human interaction in order to gain knowledge from data" [35]. The process describes different "stages" (represented by nodes) and "transitions" (represented by edges).

During the *Data* stage, data from one or several sources are preprocessed to a consistent representation required for the rest of the process. Typical preprocessing tasks include integration of heterogeneous data sources, cleaning and normalization.

The goal is to derive *Knowledge* as a result of the process. This is achieved by an iterative combination of automatic methods and user interaction. *Models* are built by applying data mining methods to the data and refined as a result of user interaction by the analyst. User interaction is facilitated by a *Visualization* that allows the analyst to explore data and influence the model (for instance, by correcting errors, selecting algorithms and tweaking parameters).

Using the *Feedback loop*, derived knowledge can be reused for the *Data* stage of the process.

[3] The term "relationship" is predominantly used to describe actual associations between people, organizations, and so on. On the other hand, "relation" is commonly used to relate concepts or things, including abstractions used in databases and ontologies. For this work, since we deal with relationships between people and organizations on both abstract and concrete levels, both terms can be used interchangeably in many cases.

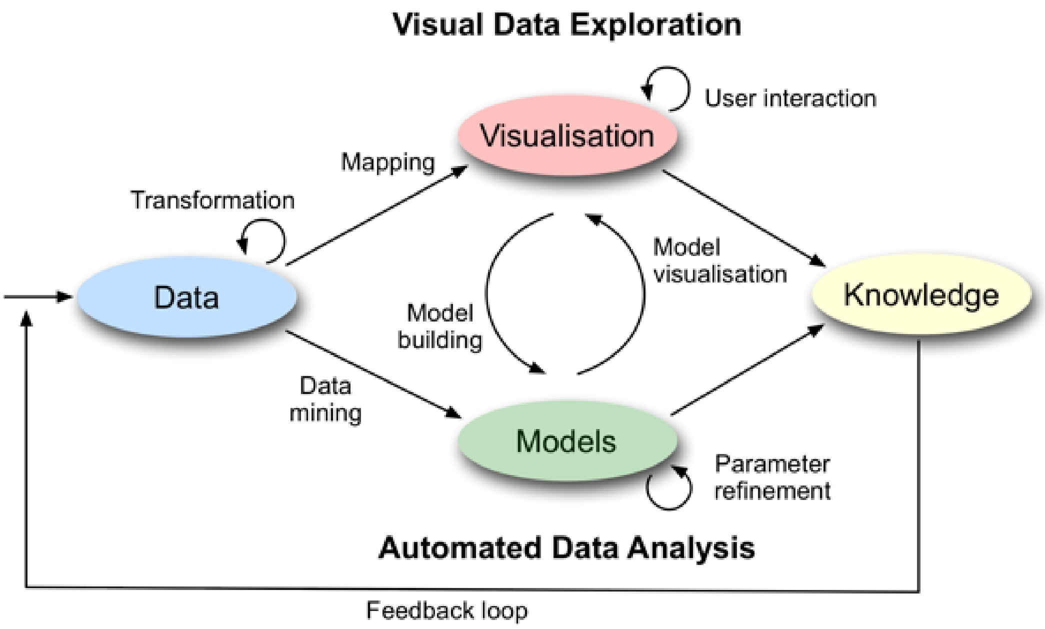


Figure 1: The Visual Analytics Process [35].

Interdisciplinary Approach

As can be seen from the figure, visual analytics is a highly interactive and interdisciplinary process. This high-level overview, however, gives little concrete details on activities like "transformation" and "data mining" or "model visualization" and "user interaction".

Naturally, topics from both language technology and visualization have been the subject of research. However, typically, research related to language technology focuses on developing, improving and evaluating automatic methods (and creating visualization, usually static, only on demand), while works from visualization focus on visualization concepts and user interaction (using already available and preprocessed datasets, without addressing data retrieval and mining).

We believe that a mixed approach could benefit the resulting system as well as both research areas. In this thesis, we focus on both visual data exploration and the automatic extraction and analysis of the data.

1.3 State of the Art

Networks of Names touches mainly on topics and methods from language technology, an overview of which is given in Section 1.3.1 as well as interactive visualization and exploration of large (social) graphs that we collectively dub *visual analytics* and address in Section 1.3.2. The current state of research in those areas is important to assess the possibilities and challenges of contemporary systems.

1.3.1 Language Technology

In the field of language technology, we rely heavily on existing methods and implementations. Our focus is to select and integrate suitable methods to achieve the task at hand. Although language technology is relevant to several parts of this work, in its whole, Networks of Names bears more similarities to visual interactive systems, such as the ones mentioned in Section 2.

For preprocessing, we require a named entity recognizer to discover names of people and organizations in text. Details on this are discussed in the context of our data preprocessing (Section 4.2). During operation, the system employs text similarity metrics and sentence clustering. Details are given in the respective part of our description of the visual interactive system (Section 5.3.3). We exploit the user interaction inherent in a visual interactive system to train and evaluate a classifier for the extraction and labelling of semantic relationships. The classifier and the related research area is discussed in Section 6.

1.3.2 Visual Analytics

Visual analytics is a collective term that describes research dealing with “analytical reasoning facilitated by interactive visual interfaces” [58]. Such interactive visual interfaces are part of computer systems that aim to combine strengths of humans and computers to allow both the user and the computer system to derive results (and thus some form of knowledge) that would be difficult or time-consuming to obtain without the interaction. Thus, the term “visual analytics” usually subsumes other fields, mainly the field of scientific and data visualization, as well as research that is required for the algorithmic parts of the system (e. g., in our case, predominantly related to language technology).

Networks of Names deals with visualization and exploration of a network that describes the relationships between people and organizations. As a result, this work bears similarities to works dedicated to the visualization and exploration of large networks, as well as the more recent field of social network analysis.

Visualization and Exploration of Large Networks

Traditionally, visualizations follow Schneiderman’s visualization mantra “Overview first, zoom and filter, then details-on-demand” [54]. In this approach, a visual interactive system would display a high-level *overview* over the graph (or data in general). With tools to *zoom and filter*, a user can locate and focus on regions of interest.

Keim et al. [36] note in their report on the challenges of visual analytics that datasets become too large and complex for straightforward visualization and present the visual analytics mantra: “Analyse first, show the important, zoom, filter and analyse further, details on demand”. Most notably, “analyse first” expresses that even now many datasets have grown so large that “dumping” them onto the screen would not provide any insightful information to the viewer. It is to be expected that datasets will rapidly continue to grow in size, while display capabilities (e. g. the number of pixels on the screen) and human perception will remain mostly constant. As a result, the ability to zoom and filter would become meaningless without additional analytical tools, including automatic analysis.

Following the insight that top-down approaches such as Schneiderman’s are rendered obsolete in the context of large datasets and focusing on graphs, van Ham and Perer develop a less abstract mantra for visual analytics applications: “Search, Show Context, Expand on Demand” [63]. Here, the user asked to specify a *search* first, to a-priori reduce the amount of relevant information. The system then proceeds to display search results with a *context* in the form of relevant surroundings, i. e. the relevant node and interesting parts of its neighbourhood. From here, the user can interact with the system to explore the data further, by *expanding* the graph into regions that he is interested in.

For searching and exploring networks, both algorithmically and manually by user interaction, a notion of interestingness is crucial. In the context of “Search, Show Context, Expand on Demand”, selecting a result for the initial search and aiding the user with expanding the graph requires the system to pre-assess what might be of interest.

In 1998 Furnas transferred the concept of *fisheye lenses*, which distort a view to display some area and its surrounding in greater detail than the rest, to selection of relevant subgraphs in trees [21]. He defines a *degree of interest* (DOI) function to compare interestingness of nodes. This function is a two-component linear combination of an a-priori interest of datapoints and a distance function that is used to discount the function value of datapoints as they move away from the user focus. Applying the approach to various tree structures, Furnas suggests that many datasets have a natural interpretation of the components.

This concept is picked up and built upon by van Ham and Perer for “Search, Show Context, Expand on Demand” [63]: In need of selecting subgraphs for user searches, they extend the concept of DOI by applying it to graphs in general (not just trees). In addition, they introduce a third component for the function that captures special user interest regardless the focal node; a concept suitable for capturing faceted search or other additional search parameters.

Social Network Analysis

In addition to visual analytics research in general, the recent research area of social network analysis, reinforced by the recent popularity of online social networks, offers contributions related to Networks of Names, since we essentially mine, visualize, and analyse a social network from newspapers.

Social networks can be expressed explicitly by friend lists on Facebook or followers on Twitter, or be derived implicitly from social interaction, such as participation in the same discussions online or retweeting each others tweets.

While we do not rely on methods that analyse the graph structure of our network, our choice of algorithms is affected by common properties of social graphs, such as the *small world property* (discussed in Section 4.3). Also, our approach to searches for relationships in the graph, as detailed in Section 5.3.1, conforms to findings from social network analysis research, as the field naturally deals with topics such as the discovery of "good" connections in social graphs [30].

1.4 Contributions

The main contributions of this work are:

1. We develop and deploy *Networks of Names*, an accessible visual interactive system for the exploration and analysis of relationships between people and organizations as portrayed by print media. The system is accessible in the sense that its use requires no special background knowledge. The practical value of the system is that it enables a user to explore a large newspaper corpus visually.

   We explore, adapt and integrate state-of-the-art methods from visualization and language technology to create a usable and useful visualization that is tailored to the properties of our social network graph. In particular, we build upon and extend methods for searching large graphs to produce meaningful results for our use case.

2. By the creation of the visual interactive system, we present capabilities of research results from the fields of visual analytics and language technology, contributing to the popularization of computer science and automatic methods.

3. We utilize user interaction inherent to the system to train and evaluate a classifier for the discovery and labelling of non-taxonomic semantic relationships. As opposed to the common approach, both the training and the evaluation process can be performed continuously instead of prior and posterior to the operation of the system.

   We conduct an exploratory user study and find that, without being predetermined, sentence patterns that in the course of the training and evaluation of our classifier emerge to have high precision correspond to patterns usually employed in related research literature. The study also reveals that the presence of a visualization can have an impact on the kind and quality of labels chosen by users for semantic relationships between named entities.

1.5 Overview

The remainder of this work is organized as follows:

In Section 2, we explore related projects, focusing on visual interactive systems for the exploration of large networks and on knowledge bases. Section 3 then gives a high-level overview over our system, its components, their interaction, and the technologies that we use. Subsequent sections deal with algorithmic and methodical details: Section 4 is devoted to our corpus data, the preprocessor, and the network we extract from the data. Section 5 details the components of the visual interactive system, explaining the visualization, the possibilities of interaction, and the automation involved. The classifier and the challenges tied to its construction are explained separately in Section 6. We evaluate the system and discuss results in Section 7, by detailling a case study, conducting a user experiment, and evaluating our classifier.

Lastly, we give possible directions for future work in Section 8 and conclude this work in Section 9.

2 Related Work

Several visual interactive systems for exploring and analysing networks (of different kinds) have been proposed and developed. Usually, the approaches focus on different aspects of visualization and interaction to address specific domain needs. We discuss such tools in Section 2.1. Knowledge bases constitute another type of system closely related to Networks of Names, because they formalize knowledge about the world and make it accessible not only in machine readable format, but also for exploration by humans, including through visual interfaces. We give an overview of current knowledge bases in Section 2.2.

The exploration of social networks is central to the field of social network analysis. In 2005, Huisman and Van Duijn compile and review a list of tools and methods (including older and non-visual tools) for social network analysis [32]. Only few of those tools are capable of processing large networks and many focus on theoretical analysis by application of different algorithms and statistical methods to data in some predefined format, an approach that assumes mathematical background knowledge and does not allow intuitive data exploration.

A commonly cited free general-purpose tool for the visualization of network data, including large networks and time-dependent data, is Gephi[4] [6]. While it has a wide range of capabilities, due to its general-purpose nature, Gephi shows similarities to complex development environments rather than accessible ("consumer") software, and considerable background knowledge and experience is needed for its use.

2.1 Visual Interactive Systems

In more recent publications, several visual analytics tools for the intuitive exploration and analysis of large network data and social graphs have been presented. Those tools are designed with the visual analytics approach in mind and apart from visualization concepts in general focus on the application of automatic methods to specific aspects of graph exploration and analysis, or on specific data types and domains.

Apolo

Following the general tendency of visual analytics to move away from top-down visual exploration, Chau et al. follow a bottom-up approach for sensemaking in large network data. With Apolo [13], the authors create a visual interactive system that encourages users to dive into network data by "starting small" and gradually building an understanding of the network structure. The system incorporates a machine learning technique called "belief propagation" that evaluates user interaction and attempts predictions of what other data might be interesting for the user based upon his action. The authors apply Apolo to a scenario in which users explore a large citation network and show that the system can actively benefit user understanding.

FacetAtlas

FacetAtlas [11] implements the possibility of "multifaceted visualization of rich text corpora", addressing the need to explore and understand cross-document and inter-corpus relationships. The work aims to provide means to organize and view information retrieval results and is thus complementary to traditional keyword-based search engines. The system organizes such search results by extracting concepts present in the data, organizing them into topics ("facets") and forming clusters that are connected by internal (in-facet) and external (cross-facet) relationships. The authors demonstrate the capability of the system by applying it to healthcare use cases and show that FacetAtlas is capable to correctly cluster and visually convey the dependence of deseases, symptoms and treatments.

[4] http://gephi.org/

Guess Who?

With their serious social network game Guess Who? [27] Guy et al. take on the problem of the quality of social network data. Focusing on internal enterprise social networks, they use gamification to encourage members of the network to complete and improve the dataset. In GuessWho?, users can create relationships and tags by interaction with the system. The authors show that deploying the system can improve the overall quality of the data significantly, yielding valid and diverse sets of both, relationships and tags. The system is designed to encourage users to add data that is novel and valid: While entering new data is rewarded, evaluation mechanisms build into the game ensure that user can not enter invalid information to gain an advantage.

PivotPaths

PivotPaths [17] proposes "strolling" as a method to navigate heterogeneous information spaces, such as journalists and their newspaper articles in relation to publication category and date. The research focus is set to visual design that encourages casual exploration of data under different facets without the need for domain knowledge or familiarity with specialized search engines. The user interface of PivotPaths aims to avoid abrupt changes between different subsets of data that is typical for current systems (like web-based movie databases or citation networks). Instead, colouring is used to conform to the user's attention focus, while animations help the user to preserve his mental model across transitions.

SaNDVis

Perer et al. focus on relationship discovery in the enterprise. Relationship discovery, unlike information discovery in general, is aimed at searching and finding people, as opposed to documents. The authors present SaNDVis [46], the visual analytics interface of the system (SaND and SaNDGraph are the components responsible for data mining and model building). The system automatically extracts and displays relationship data derived from documents present in the enterprise. The visual interface allows users to view the resulting social graph and query it for certain facets such as area of expertise.

They Rule

They Rule[5] provides a simple interactive network visualization of US companies, their director's boards and interconnections. It is possible to search for people and companies and interactively expand and layout the resulting network graph. The data is retrieved from LittleSis (see Section 2.2).

2.2 Knowledge Bases

Knowledge bases aim to provide structured data on facts about the world. Such facts include data on people, organizations and their relationships and is therefore of interest for Networks of Names and relationship extraction in general.

In contrast to visual interactive systems, which focus on data visualization and a human-computer interface, knowledge bases are usually designed to be machine readable.

Freebase

Freebase[6] is a large knowledge base that contains a diverse range of entity types and relationships between them. Originally created by Metaweb Technologies, the company and service were acquired by Google in 2010. Most data is crawled and extracted automatically from web sources, such as Wikipedia. Users have the possibility to add new and edit existing entries. All relationships fit into a predefined list of types, including basic information on the entity, awards, authorship of books, appearances in media, and web presence. Data can be accessed through a URL-based API under a creative commons license (CC BY 2.5).

[5] http://theyrule.net/

[6] http://freebase.com/

LittleSis

LittleSis[7] is a knowledge base by the Public Accountability Initiative, a US non-profit organization. The database aims to provide transparency to "the key relationships of politicians, business leaders, lobbyists, financiers, and their affiliated institutions". The data is compiled and updated mostly manually and partly automatically from government documents, news articles and other similar publications. The project focuses exclusively on American public figures, organizations and institutions. While the facts on entities contain basic information similar to other knowledge bases, the focus is on memberships and political positions, money transactions and other domain-specific relationships. Data from LittleSis can be accessed automatically through a URL-based API and is offered under a creative commons license (CC BY-SA 3.0 US).

Yago

Yago is a knowledge base created by the Max Planck Institute for Computer Science. Its contents are derived from Wikipedia, WordNet[8] (a lexical database), and GeoNames[9] (a database of geographical entities such as countries, cities, and landmarks along with some of their characteristic attributes). The knowledge base contains a large variety of object types, including people and organizations. Yago provides a SPARQL Interface and an interface that allows to query SPARQL-like using natural language (with queries such as "Beatles' wives born before Woodstock and near London"). In addition, there are web interfaces for the composition of such queries. Alternatively, the knowledge base can be browsed in a text-based format or through a visual view that displays an entity in the centre and its attributes and relations in star-form around them. This interface has rudimentary capabilities of user interaction, most notably the possibility to mark facts as correct or false. Data from Yago is licensed under a creative commons license (CC BY 3.0).

Wikidata

Wikidata[10] is a project of the Wikimedia Foundation and is a free and collaborative knowledge base. It contains editable facts on entities, such as people's date and place of birth, citizenship, and family relations, as well as links to different language versions of the respective Wikipedia articles on that entity. Furthermore, the data is used to populate Wikipedia's infoboxes and is expected to make this kind of information more coherent and complete over time. Data from Wikidata can be retrieved automatically by a URL-based API and is offered without copyright claims (CC0).

[7] http://littlesis.org/

[8] http://wordnet.princeton.edu/

[9] http://www.geonames.org/

[10] http://wikidata.org/

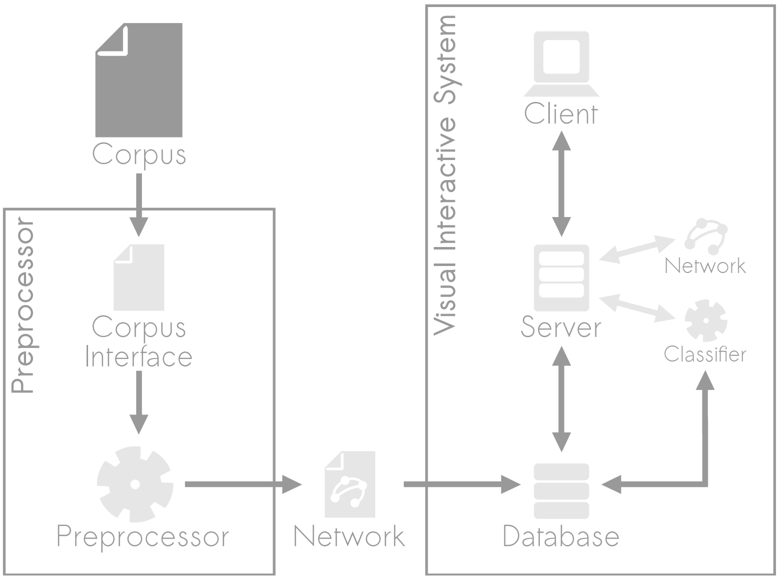


Figure 2: Overview over the components and data flow in Networks of Names

3 Overview of the System

This section gives a high-level overview over the components of Networks of Names and the technology used for their realization. Section 3.1 presents the main components and explains how they interact. Section 3.2 contains information on the technology, such as programming languages and frameworks.

3.1 Components and their Interaction

A graphical representation of the architecture is depicted in Figure 2. The system can roughly be divided into two major parts: The preprocessor, which is responsible for the conversion of a text corpus into a social network, and the visual interactive system, which provides visualization of and interaction with the network.

The design decision to separate the two parts and execute them sequentially stems from our initial situation, in which we have a large corpus, but no existing network yet. In future work, when both an initial network and new text documents exist, the preprocessor and the visual interactive system could be refactored to interact iteratively and more closely in order to extend the existing work (as opposed to creating it in the first place).

Preprocessor

The preprocessor runs prior to the operation of the visual interactive system and is responsible for the extraction of a social network from a natural language text corpus made up of sentences.

The corpus is accessed through an interface that is used to abstract from the actual corpus representation. In our case, this is used to combine several corpora into a single large corpus, but could in the future be employed to provide access to different data sources in arbitrary format (as long as they contain natural language sentences and their sources). The interface exposes a data structure that can be used by the preprocessor to sequentially step through the corpus data.

The preprocessor core itself converts sentences into a social graph by extracting entity names and their relationships from the sentences and performing data cleaning[11] to improve the quality of the network. This process is detailed in Section 4.

[11] While conceptually data cleaning is part of the preprocessing steps, for practical reasons, it is currently implemented as a part of the visual interactive system, because this allows for testing and adjustment of cleaning criteria during development without the need to re-run the preprocessor.

The preprocessor produces a tab-separated file-based representation of the network, including the source data obtained from the corpus.

The application is implemented using a client-server architecture and uses a database for storage. We choose to implement Networks of Names as a web application in order to make it more accessible, and web applications pose no requirements on the client except for the presence of a sufficiently recent browser.

Database

The database contains the source data (sentences and sources), our derived social graph (entities and relationships, along with their attributes) as well as data created and used by the classifier (see Section 6). The data can be loaded into the database from the file-based representation created by the preprocessor.

Server

The server is the backend part of the application and interfaces with the database for storage. On startup, the server connects to the database and loads a graph representation of entities and relationships into memory. This in-memory representation is static and remains loaded through its lifecycle. It is used to answer common user queries and perform graph algorithms. Queries for additional data from the database are performed on demand, as it becomes necessary during operation. The network itself is never changed as a result of the operation of the system. Thus, neither the in-memory representation nor the relational database representation of the graph need to be updated.

Apart from this server-owned data, i. e. the in-memory representation of the network, the server is stateless, i. e. it maintains no session state with respect to clients. Instead, for every request, the client provides all information that is needed for the server to respond.

Tasks that require the presence of the global graph structure, access to the database, or that are computationally demanding, are performed by the server. Such tasks include answering user searches and subsequent expansion of the graph, and the clustering of source sentences by similarity. Details on these tasks are discussed in Section 5.3.

Initial client access is served by returning the HTML structure, style and scripts that form the frontend. All other relevant methods that handle requests are exposed for asynchronous access by the client.

If a client performs interaction that is used to train the classifier, the server fires background tasks that train and subsequently apply the classifier. Since this is a potentially long-running task, the client does not have to wait for the classifier to complete its work. The classifier writes computation result back to the database. Since classification results do not affect the in-memory graph representation, no further work is needed to make them available to the server. The inner workings of the classifier are documented in Section 6.

Client

The frontend part of the application (client) is executed by the browser running on the user's machine. When first accessing the service, the frontend application is served to the client.

The client maintains user-specific application state, i. e. current search parameters and visualization state. It is responsible for creating the visualization, handle user interaction and communicating with the server to obtain data directly or indirectly requested by the user.

The user interface of the frontend (see Figure 3) consists of controls to create new custom searches, open example searches, export the current visualization state and clear the current visualization. The visualization is rendered in the middle of the page. At the bottom, additional information on usage and known issues, and a legend, is available. Details on visualization and user interaction are given in Sections 5.1 and 5.2, respectively.

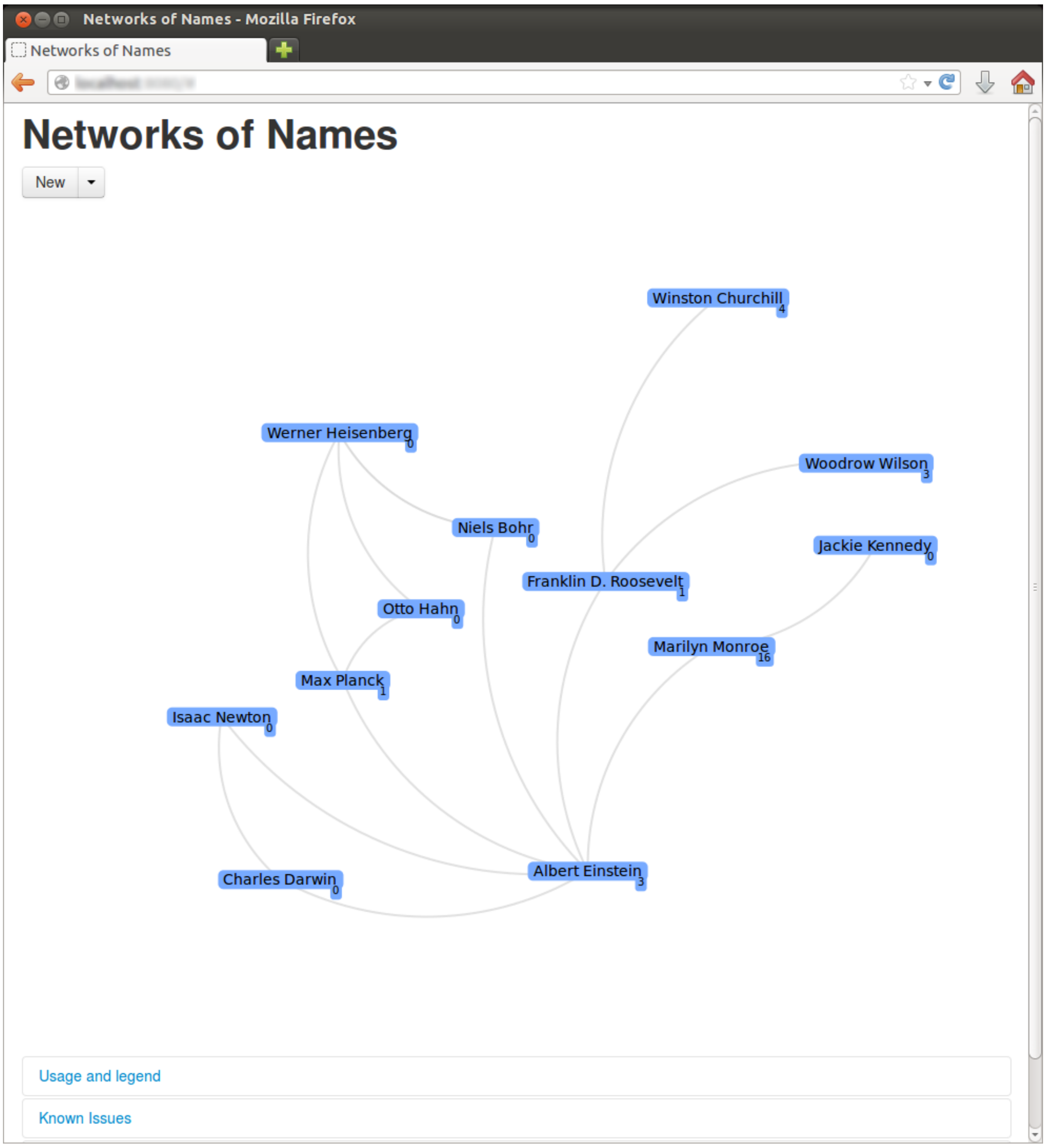


Figure 3: Networks of Names frontend user interface

3.2 Technology

In this section, we briefly describe the programming languages and frameworks used in the architecture of Networks of Names, along with a reasoning for their selection. This section focuses on main components and omits libraries included for specific, minor tasks. Such libraries are referenced in other parts of this work where they are relevant.

Preprocessor

The preprocessor is implemented in Scala and executes some UNIX command line tools such as `sed` and `sort` (and is platform-dependent as a result). For the extraction of entities, we employ the Stanford Named Entity Recognized, which is discussed further in Section 4.2.

Intermediate and final results are stored in tab-separated files. This choice was made, because text files allow fast line-by-line processing and permit the usage of command line tools for tasks like search/replace or sorting. This also avoids the overhead induced by the usage of a database when changes to data happen frequently.

Scala[12] is a statically typed, object-oriented and functional programming language that is compiled to Java Byte Code and executed by the Java Virtual Machine. Scala was chosen for personal preference, and because it seamlessly interacts with existing Java libraries. In addition, especially for tasks that contain large portions of data processing, functional programming languages can have significant advantages for code expressiveness and conciseness, which results in easier maintenance.

Visual Interactive System

The visual interactive system is a diverse system that requires different technologies on the back- and frontend. In the frontend, the visualization itself requires additional tools, because the high degree of interactivity is otherwise hard to achieve "from scratch" and within the scope of this work.

Database

We obtain a network and the source data from preprocessing. This data is imported into a Postgre-SQL database into tables for entities, relationships, sentences, and sources along with two tables representing many-to-many relationships from relationships to sentences and from sentences to sources, respectively. We create indexes for the unique ids and for entity names (as we intend to search for entity names).

Several reasons influence our choice of a SQL database:

1. Relational database formats are flexible enough to allow screening of and changes to the data during development where the exact form of the network is not yet fixed.

2. Databases like Postgre-SQL are heavily optimized and well-documented.

3. SQL-based databases allow for both, conveniently loading frequently used subsets of data into memory and performantly querying the data for specific datasets.

4. The Leipzig Corpora Collection that we use as our corpus is offered in a relational format.

This allows us to store other data using the same persistence solution. In addition to the relational graph representation, our database contains tables to maintain data created by user interaction and the automatic relationship classifier (Section 6) as well as auxiliary data required for the user study and evaluation (Section 7.2).

[12] `http://www.scala-lang.org/`

Server

Like the preprocessor, the server is implemented in Scala. As a framework for the construction of web applications, we use Play. For the in-memory representation of the network, we use the JUNG framework.

Play[13] is a framework for development of web applications in Scala. Play is free software released under under the Apache 2 License.

The framework includes several features required for the creation of Scala applications in general and web applications in particular. Most notably, it provides dependency management, file structure, easy routing, a template system as well as some other useful components.

We select Play for personal preference and because the framework itself is developed in Scala and thus offers a solution tailored to Scala programming paradigms.

The **Java Universal Network/Graph Framework**[14] (JUNG) is a popular software library for the representation, visualization and analysis of graphs [44]. JUNG is free software written in Java and available under the BSD license.

The core library provides graph representations for several types of graphs, including directed and undirected as well as multigraphs. JUNG's implementation of graphs contains methods that are frequently required in graph processing, most notably the retrieval of neighbours and the extraction of subgraphs induced by a set of nodes or edges. The library includes a number of algorithms, such as Dijkstra' shortest paths algorithm, and algorithms for calculation of graph metrics.

The visualization component of JUNG (realized for Java's Swing framework) is not employed in Networks of Names, because we rely on a web frontend, but was used during the prototyping stage.

Client

The client (in line with current web development standards) is implemented in JavaScript directly or CoffeeScript[15], a JavaScript alternative that aims to simplify and clarify JavaScript syntax, and is compiled to JavaScript sources by the Play framework. Pages served to the client are represented in HTML5 and styled by CSS3, relying on Bootstrap[16] for common modern frontend components. The visualization is created using D3.js, a framework for the visualization of data in the browser.

Data Driven Documents[17] (D3.js) is a JavaScript framework "for manipulating documents based on data". "Documents" refers to any kind of structured data that can be accessed through the *Document Object Model*[18] (DOM) interface, as defined by the *World Wide Web Consortium* (W3C), i. e. (X)HTML or XML documents. Most commonly, SVG is used to render the underlying data.

As a framework, D3.js does not introduce completely novel vocabulary or interfaces. Instead, it employs existing, non-proprietary, and flexible data representation and technologies. For that, relevant parts of the network are represented in JSON format and provided as the underlying data to the visualization, while the visualization itself is created using SVG and CSS3. D3.js provides the "glue" to transform the former into the latter. In addition, data elements remain linked to the visualization, so that changes to the data can be defined to result in changes in the visualization.

For common tasks in DOM manipulation, namely selection and transformation, D3.js provides convenience methods following the functional programming paradigm and chaining. This allows for declarative programming of the visualization. DOM properties can be set to functions instead of static values, allowing the visualization to easily depend on data properties and thereby react to changes in the data.

Alongside functions for DOM manipulation, D3.js also contains implementation of common tasks that arise in the context of data visualization. For instance, it is possible to bind click handlers and enable

[13] http://www.playframework.com

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

[15] http://coffeescript.org/

[16] http://getbootstrap.com/2.3.2/

[17] http://d3js.org/

[18] http://www.w3.org/DOM/

panning and zooming the document. For graph visualizations, D3.js contains a configurable force layout algorithm.

D3.js is free software, distributed under a BSD-style license. It is under active development and well documented, including documentation of core features, as well as numerous and diverse tutorials and examples.

The framework proves to have good performance. Rendering and animating graphs with over 50 nodes, along with links and labels, is not a problem. The technology is dependent on modern browsers, but all reasonably current browser versions support fast execution of JavaScript and implement large portions of the SVG specification. D3.js visualizations can be found in New York Times publications, and, according to Wikipedia, are also employed by a number of other projects[19].

[19] http://en.wikipedia.org/wiki/D3.js

4 Mining the Network

In this chapter we describe our data sources, preprocessing steps, and the properties of the resulting network. This corresponds to the stage *Data* and transition *Transformation* from the visual analytics process shown in Figure 1 on page 8. From the data, we derive a social network graph and its respective vertex and edge properties (*Data Mining* of *Models* in the visual analytics process).

Section 4.1 discusses requirements for our corpus, chances and challenges of obtaining data directly from online publications of newspapers, and the Leipzig Corpora Collection that we use in Networks of Names. Section 4.2 explains how we extract and process data from the corpus. Section 4.3 discusses the properties of the resulting network.

4.1 The Newspaper Corpus

The objective of Networks of Names is to visualize people, organizations and their relationships as represented by print media. To achieve this goal, we need to collect and process newspaper contents.

To keep the required effort within the scope of this thesis and due to personal preference and knowledge, we focus on German newspaper articles and defer the adaptation and integration of tools to handle different source languages to future work. To meet the goals of a visual analytics tool that allows the exploration of the underlying data, we require that elements of the visualization be traceable to data from which they originate. Lastly, relationships between people and organizations change over time. This is naturally represented by newspapers and newspaper articles published on different dates. In order to provide a tool that is of interest to people who want to explore contemporary events, the corpus should contain sufficiently recent data, span over several years of recent history and be expandable by sources that are published after its initial creation.

To summarize, a corpus for Networks of Names should have the following properties:

1. It should consist of text from newspaper articles.
2. It should be recent and span over several years.
3. It should be expandable by new data.
4. It should be diverse and large enough to be a fair representation of newspaper articles of that time.
5. Although it is not a general design restriction, for the scope of this work, the corpus language should be German.

Newspaper APIs

Nowadays, nearly every newspaper has an online version. Online newspapers are naturally more suitable for machine processing than printed newspapers.

However, information in newspapers, online and offline, follows no absolute criteria for form and type of content, Different layouts, inconsistent and incomplete markup, and diversity of content presented on a single page makes it difficult to automatically parse contents and separate them by type with confidence (i. e. headline, teaser text, article content, advertisement, unrelated text, . . .).

To mitigate this problems, some newspapers have created online APIs. However, the existence of such APIs is a very new tendency, making them few in number and experimental in nature. Existing APIs are unstandardised, often incomplete and lack comprehensive documentation. For example, Zeit Online[20] provides a beta API for developers[21]. While it is a powerful tool, results returned by it do not contain the article text. To extract that, we face the same problems that are associated with parsing newspaper articles.

[20] http://www.zeit.de/

[21] http://developer.zeit.de/

Leipzig Corpora Collection

Under the name "Leipzig Corpora Collection"[22] the Leipzig University department for Natural Language Processing maintains a rich collection of various language corpora. The collection contains a significant fraction of corpora extracted from newspapers. For German online newspapers, as of this date, corpora ranging the timeframe from 1995 to 2010 are available publicly and are composed of millions of sentences.

For the newspaper corpora, the creators automatically crawl and sample online newspaper articles for sentences. The sentences are then, also automatically, cleaned and stored together with publication date and source URL. Additionally, the corpora contain precomputed statistical information and metadata that is not relevant to nor used in Networks of Names. Details on this process can be found in [49].

4.2 Preprocessing Steps

Starting with data from the Leipzig Corpora Collection, several preprocessing steps are perform to obtain the data that is eventually used by Networks of Names.

First, named entities and relationships between them are extracted from the sentences. The datasets are then aggregated and assigned unique identifiers. Last, the data is imported into a database for persistence and querying.

4.2.1 Extraction of Named Entities

Recognizing people and organization in natural language text is a classification problem referred to as *named entity recognition* (NER).

NER has been studied for a wide variety of language, domains, and types of entities. Most interestingly for this work, German as a language, the journalistic domain, and both people and organizations as entity types are among the best studied. While research in NER is ongoing, named entity recognizers of solid performance can be found for several application domains. Since the focus of this work does not lie in the improvement of NER methods, we rely on existing methods and some additional data cleaning to find and extract named entities and their types from text.

For a survey of methods in named entity recognition, see [41].

The Stanford NER

In Networks of Names, we employ the *Stanford Names Entity Recognizer*[23] [20] (Stanford NER) published by the Stanford Natural Language Processing Group[24]. Specifically, we use the *German NER* by Faruqui and Padó [18].

The Stanford NER is a *Conditional Random Field* (CRF) Classifier. CRFs, a probabilistic framework introduced first by Lafferty et al. in 2001 [37], are the state-of-the-art method for classification of sequences. Common sequence classification tasks include part-of-speech tagging and other similar problems from computational linguistics as well as DNA tagging in bioinformatics. The method bears similarities to *Hidden Markov Models*, but instead of joint probabilities, it defines conditional probabilities $p(\boldsymbol{Y}|\boldsymbol{x})$ over label sequences $\boldsymbol{Y}$, given an observation sequence $\boldsymbol{x}$. A suitable label sequence $\boldsymbol{y}_\star$ for a new observation sequence $\boldsymbol{x}_\star$ is then selected by maximizing $p(\boldsymbol{y}_\star|\boldsymbol{x}_\star)$. For a brief introduction to CRFs see [66], for a more detailed tutorial see [57].

The classifier performs tokenization of sentences and classification of the elements of the tokenized sequences into classes of entities. For the German NER those classes are: `Person`, `Organization`, `Location`, and `Misc`. The Stanford NER for English supports other classes as well. In Networks of Names, we make use of the classes `Person` and `Organization` only.

[22] http://corpora.informatik.uni-leipzig.de/

[23] http://nlp.stanford.edu/software/CRF-NER.shtml

[24] http://nlp.stanford.edu/

For the German NER, two models are available. One is trained on the Huge German Corpus[25] (HGC), the other on the deWaC[26] corpus. We selected the former, because it is trained on newswire text and thus suggests to be a good fit for our application and because, from a quick glance, it seemed to outperform the deWac-trained classifier on our sample newspaper sentences.

Usually, named entity recognizers use a *BIO scheme* for labels [50], with B standing for "beginning" of a labelled sequence, I for the "inside" of a sequence and O for "outside", referring to unlabelled tokens. Thus, the first token of a multi-part person's name would be labelled `B-PER` and all other parts would receive the label `I-PER`. Tokens that are not part of names would uniformly be labelled `O`.

A peculiarity of the the German NER using any of the two pre-trained models (in which it differs from the Stanford NER for English) is that it does not label entity boundaries. Instead, all tokens that are part of a name are assigned the "inside" variant of a label, while the "beginning" variant is unused. As a result, every uninterrupted sequence of tokens that have been assigned some label that is not `O` are treated as one (single) entity. In practice, there are cases where two names appear next to each other and are incorrectly recognized as one entity. However, since such cases are sufficiently rare, we do not pursue a solution to that problem.

Our Approach

We apply the Stanford NER to each sentence from the corpus and extract all names (and their types) annotated to be a person or organization.

If a name appears with different types (i. e. sometimes classified as `Person` and sometimes as `Organization`), we assume the more frequent type to be correct and overwrite the less frequent type by it. The frequency of each name is then counted and multiple appearances are aggregated into one.

After extraction, we perform several data cleaning steps discussed in Section 4.2.3 to improve the overall quality of results.

Disambiguation of Entity Names

In natural language text, the same proper name may refer to different real-world entities. Likewise, the same entity may be referenced by different names and variations thereof (as well as pronouns and descriptive phrases). A named entity recognizer is capable of detecting appearances of proper names in text and determining their types. However, given an instance of a name, it cannot decide what entity the name refers to, or whether two instances refer to the same or to different entities.

For instance, the name *George Bush* may refer the the 43rd president of the United States, or his father, the 41st president of the United States, as well as a number of other people bearing the same name. Additionally, the same person can be referred to by alternative variations of his name (such as *George W. Bush*, i. e. including an abbreviation of his middle name), just his surname (*Bush*), or the genitive form of any of those possibilities (e. g. *Bush's*). Similarly, organizations are often referred to by different names. For instance, George Bush's political party, the *Republican Party* of the United States, is also commonly called the *GOP* (as in "Grand Old Party") or simply *Republicans*.

The problem of linking instances of names to entities they refer to is called – depending on the point of view and focus of the work – *named entity disambiguation*, *named entity resolution*, or *named entity normalization* (see, for instance, [29, 53]), or more generally *record linkage* [19]. The respective entities can be identified by a URI[27], such as a link to their Wikipedia page or some other knowledge base. For instance, in the English Wikipedia, George Bush junior unambiguously corresponds to `http://en.wikipedia.org/wiki/George_W._Bush`, while George Bush senior is denoted by `http://en.wikipedia.org/wiki/George_H._W._Bush`, and URIs for other namesakes typically contain the relevant domain in addition to their name, such as in `http://en.wikipedia.org/wiki/George_Bush_(NASCAR)` for a race driver named George Bush.

[25] http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/hgc.en.html

[26] http://trac.sketchengine.co.uk/wiki/Corpora/DeWaC

[27] Unique Resource Identifier

Usually, disambiguation is achieved by maintaining dictionaries of name variations and misspellings, by comparison of a document a name appears in (e. g. text, paragraph or sentence) to documents where the corresponding entity is already known (e. g. the Wikipedia article), or – especially for people and organizations – by comparison of social relations implied in the text to known social relations of entities (that can be retrieved from their respective Wikipedia pages or other knowledge bases).

We do not implement a disambiguation mechanism within the scope of this work, but note that it would constitute a useful addition, because in the social network that results from our extraction, cases do occur where one entity is represented by several vertices (with different names, variations, and misspellings) or multiple entities are represented by just one vertex, because they share the same name. In practice, the first case problem can be be avoided by working with one vertex and ignoring/removing other vertices representing the same entity, while the second can be counteracted by ignoring/removing the relationships of the "wrong" entity, and keeping only the relevant ones.

4.2.2 Extraction of Relationships

Relationship or *relation extraction* is the task of detection and classification of mentions of semantic relationships between entities in natural language text.

Relationship extraction is a fundamental problem of language processing and knowledge discovery, because the understanding of semantic relationships is crucial for understanding the content of spoken or written language and to derive knowledge about the world. As a result, the topic has been subject of numerous and diverse research [3, 52, 42].

Our Approach

During the preprocessing step, we opt to proceed without complex methods for relationship extraction and employ a very simple and broad definition of relationships: We regard two entities to share a connection if they appear together in a sentence.

The main reasons for this is our focus on user interaction. Specifically:

1. We want users to be able to classify relationships. That is, while we are interested in the detection of relationships, we do not want to focus on their automatic classification during preprocessing, and thus possibly prevent valid user decisions. Instead, we aim to train and evaluate a classifier based on manual labelling by user interaction. The purpose of this is to study what results and result quality can be achieved by simple methods in conjunction with human capabilities in this context.

2. Most methods for relationship extraction classify relationships into a set of predefined relations. However, in Networks of Names we deliberately choose to allow for various kinds of relationships and leave the final decision to the user. That is, instead of a predefined set of relations, we rely on a user-created *folksonomy*.

3. An additional factor is that, differently from named entity recognition, no single definitive method exists in the field of relationship extraction yet. Selecting some of the methods available would drive the focus of this work in a certain direction.

Thus, during preprocessing, we do not investigate semantics of relationships further. Instead, we engineer a classifier based on user interaction. Details on the challenges of non-taxonomic relationship extraction and labelling, as well as our classifier, are discussed in Section 6.

Once all relationships are extracted, their frequency per pair of entities is counted and multiple appearances are aggregated into one.

Like entities, relationships are also cleaned to remove likely low-quality data. Our data cleaning steps are described in the following section. Relationships are affected by both relationship and entity cleaning, because the removal of entities leads to the removal of their relationships.

4.2.3 Data Cleaning

Fully automated processing is prone to occasional errors and imprecision that lead to junk output. Using such output as input for subsequent processing yields no meaningful results. To prevent this, we perform data cleaning on several occasions before passing the data to preprocessing steps.

Leipzig Corpora Collection

Some sentences in the Leipzig Corpora Collection contain common encoding errors related to German special characters (for instance Ã¶ instead of ö), different types of dashes and quotes, bullet points, and other symbols.

Such cases have an adverse impact on results returned by the Stanford Tokenizer (prior to named entity recognition). Therefore, before running the preprocessor, we perform a search/replace of "corrupt" symbols combinations, that we assume to represent others, by the respective "correct" symbols. We remove sentences that contain unrecognisable characters or ambiguous character combinations altogether, because this randomly samples the dataset without any substantial influence on the size or expressiveness of the original sample.

Named Entities

We employ several heuristics to clean the set of entities recognized by the Stanford NER:

- We remove named entities that appear only once in the complete dataset, since they are likely to be false positives.
- Entities that contain braces, quotes, or commas are removed, since this is extremely uncommon for names and hence likely the result of incorrect tokenization or classification.
- We consider entities with names that consist of two characters only to be likely false positives. This is not strictly true, but very common to have negative implication on the quality of the dataset in practice. Typical cases are general abbreviations that reference a specific named entity in one sentence, but different entities across sentences. Examples include abbreviations that are part of sport club names (like "FC", "SC", "AS"), abbreviations identifying the type of institution (like "TU" for "Technische Universität" (university of technology), or "AG" for "Aktiengesellschaft" (stock company)), stock symbols or parts thereof (like "C" for Citigroup and previously Crysler, or "KO" for Coca Cola in the US stock market), abbreviations that reference affiliation (like "US" in "US Senate"), and abbreviations that are ambiguous ("BA" for "Bundesanstalt für Arbeit" (federal labour office), "British Airways", "Berufsakademie" (university of cooperative education) and several other meanings). Some two-character names, especially for organizations, commonly reference a specific entity rather unambiguously: "VW" for "Volkswagen", "EU" for "Europäische Union" (European Union) and several other names of organizations or institutions. We whitelist those names and do not remove them.
- We remove news agencies, because their occurrence in sentences rarely describes their relationships, but instead denotes the origin of the news. Excluded entities include "dpa" and "Reuters".
- We remove entities that are neither people nor organizations, but are frequently classified as such by the named entity recognizer. A very frequent example is "Euro".
- We remove entities with names that are single common first names. Such entities are highly ambiguous and create connections in the graph that are mostly meaningless.
- We remove entities that have only one relationship. This heuristic has rather low precision, but in practice helps remove misspellings and keep overall number of entities low for performance.

Relationships

Relationship of entities that are removed due to entity cleaning, are also removed. Apart from that:

- We remove relationships that appear only once in the complete dataset, as we find such relationships to be mostly random occurrences instead of meaningful connections. Especially for our larger dataset of 70 million sentences, this cleaning step helps considerably reduce the number of relationships, resulting in better performance and less visual clutter overall.

4.3 Properties of the Network

From the preprocessing steps we obtain the social *graph* or *network* to be used by the visual interactive system. Vertices of the network represent entities, i. e. people or organizations, and undirected edges represent relationships between them. We choose the network to be undirected, because at this stage we do not know anything about semantics of the relationships, and thus cannot assign directions to them.

We acquire the network from combined corpora taken from the Leipzig Corpora Collection (LCC). For development, we use a corpus of 16 million sentences spanning the years of 1995 to 2010 (one million per year). For the operation of the system, including the user study and evaluation, we use the largest set of newspaper sentences in German (which is the language with the highest number of sentences in the LCC) available from the LCC, amounting to 70 million sentences for the same timespan (ten million for the years 2007 to 2010, respectively, and one to three million for each of the other years). Though the LCC has been used for the study of networks and their properties [8], to our knowledge this is the first application of it to extract a social network. Although we focus on German within the scope of this work, this is no general design limitation, and the work could be adapted to include other languages in the future.

At this point, i. e. before any relationships (and therefore edges) are labelled, we find the social graph to already contain interesting subgraphs, such as the connections of the Darmstadt University of Technology (Figure 4) or the vicinity of Charles Darwin (Figure 5).

Metrics

For the development of our system, including the selection and parametrization of algorithms, we are sometimes required to judge the methods by how “well” they work on our network, and thus based on subjective criteria such as aesthetics. We do not exclude the possibility that our approach might also work on similar networks or be unsuitable for networks with characteristics different from ours. Therefore, we give the properties of our network as a list of metrics commonly used to characterize networks in Tables 1 and 2. The first table shows properties of the smaller graph used during development (and thus relevant for the design and evaluation of methods during development). This graph is extracted from 16 million sentences. The second shows properties of the larger graph used for the user study and evaluation (and thus relevant for results presented in Section 7). This graph is extracted from 70 million sentences and is roughly four times larger than the the graph we use in development, judging by the numbers of vertices and edges. We give the details on both networks to allow a comparison for similarities and differences between them.

The *number of vertices* and *edges* corresponds to the number of extracted entities and relationships, after data cleaning. The *average degree*[28] is the average number of incident edges for each vertex; this denotes the average number of relationships one entity is involved in. Since the graph is undirected, every edge is counted as incident to two vertices. The *average shortest path length* is the average length of the shortest path between any two vertices. The *diameter* is the length of the longest among all shortest paths.

[28] The vertex degrees in our network are distributed following a power law (or possibly some other similar distribution). We list the average degree for completeness, but note that in itself it is not a very expressive metric, as vertices that have considerably lower and higher degrees than the average are very common and not evenly distributed. We reference this metric to point out that vertices with very high degrees, i. e. degrees that vastly exceed the average, exist.

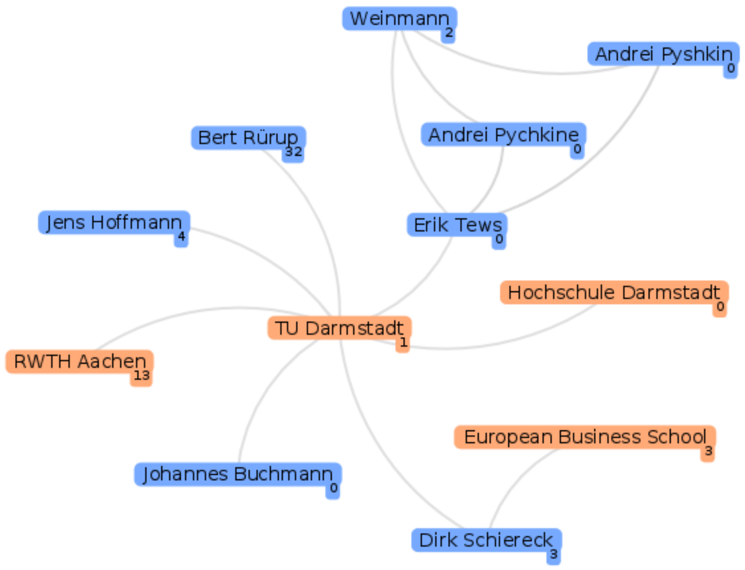


Figure 4: Unlabelled relationships of TU Darmstadt in the social network extracted from 70 million newspaper sentences taken from the Leipzig Corpora Collection.

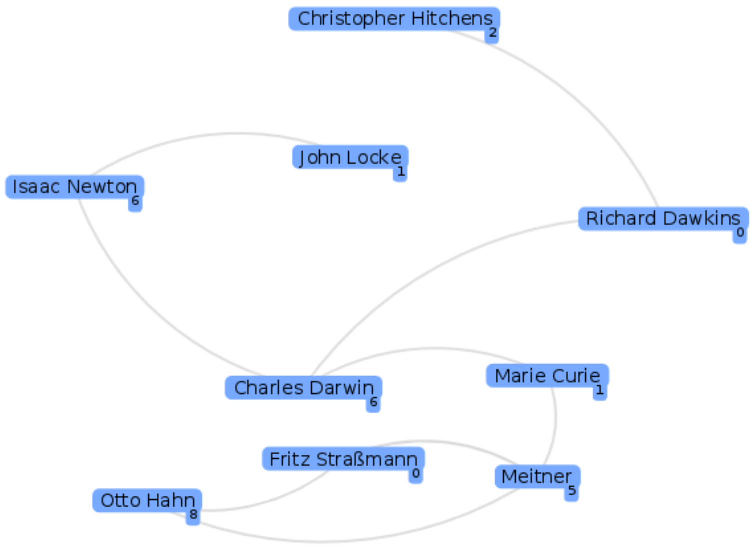


Figure 5: Unlabelled relationships of Charles Darwin in the social network extracted from 70 million newspaper sentences taken from the Leipzig Corpora Collection.

Property	Value
Number of vertices	12,460
Number of edges	41,397
Average degree	6.64
Average shortest path length	4.63
Diameter	21
Network clustering coefficient	0.47
Power-law exponent	2.62

Table 1: Properties of the network extracted from 16 million sentences (used during development).

Property	Value
Number of vertices	47,939
Number of edges	184,053
Average degree	7.68
Average shortest path length	4.56
Diameter	20
Network clustering coefficient	0.46
Power-law exponent	2.51

Table 2: Properties of the network extracted from 70 million sentences (used for user study and evaluation).

The *network clustering coefficient* is the average of *local clustering coefficients* of vertices in the graph. Local clustering coefficients describe how close vertex neighbourhoods are to forming a complete subgraph (*clique*). The value is given by the number of edges present in the neighbourhood divided by the maximum number of edges theoretically possible (if the neighbourhood were complete). Thus, it ranges between 0 (neighbours are unconnected) and 1 (the neighbourhood is complete).

The *power-law exponent* explains the distribution of vertex degrees, in that for an exponent α and (sufficiently large) degrees d, the probability of a vertex having degree d follows a power law $P(d) \propto d^{-\alpha}$. We calculate the power-law fit following the methods described in [15], using an implementation available from the paper's companion website[29]. The resulting power-law exponents are valid for $x_{\min} = 29$ (i. e. for degrees greater or equal 29) and are statistically significant with p-values[30] of 0.51 and 0.64, respectively. The resulting power law is shown graphically in Figure 6.

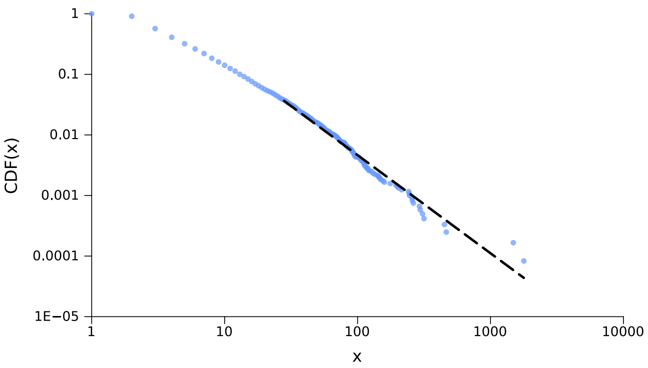
Scale-free networks

If the distribution of vertex degrees in a network follows a power law, the network is called *scale-free* [5], referring to the absence of a scale in form of a typical vertex degree. Instead, power-law distributions are heavy-tailed and imply that while most vertices have low degree, vertices with a degree that vastly exceeds the average are relatively common, with no bound on the deviation from the average.

This property is usually conjectured for social networks like ours (and networks from a number of other fields), and attributed to network growth over time and *preferential attachment*, i. e. the tendency

[29] `http://tuvalu.santafe.edu/~aaronc/powerlaws/`

[30] The p-values are obtained from a *goodness-of-fit* test presented in [15]. The authors quantify the plausibility of the hypothesized power-law model by measuring the distance between the empirically observed dataset (i. e. the actual vertex degree distribution) and synthetic datasets generated from the model. Values close to 0 denote low plausibility, while values close to 1 indicate that differences can be completely attributed to statistical fluctuations. Models with p-values above some threshold, commonly 0.05 or (more conservatively) 0.1, express that there is a statistically significant probability of the observed empirical dataset to be a result of the model.



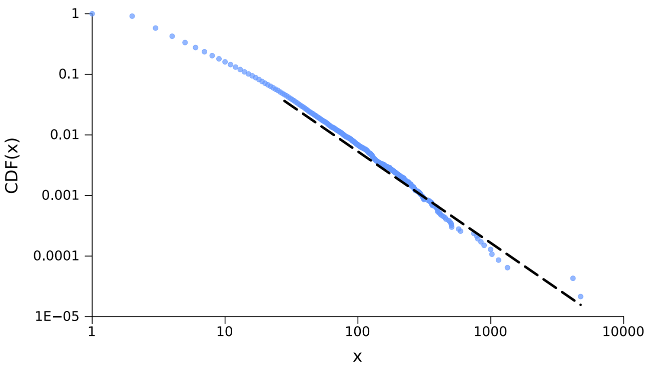


Figure 6: Complementary cumulative distribution functions[31] CDF(x) of vertex degrees (blue dots) and their maximum-likelihood power-law fits (black dashed lines) on a log-log-scale, for the development network derived from 16 million sentences (left) and production network derived from 70 million sentences (right).

of new vertices to form connections to existing high-degree vertices (as opposed to forming connections randomly). Mathematically, however, the presence of power-law distribution has been recently shown to be neither obvious nor easy to prove [15] due to its similarity to other relevant distributions in empirically observed data, such as exponential or log-normal. Following the methods from [15], our network does, however, exhibit properties that favour this assumption.

The main characteristic of scale-free networks is thus the presence of high-degree vertices, generally referred to as “hubs”. In social networks, including our network, such hubs are usually celebrities and politicians, multi-national corporations, famous sports clubs, and political parties. Hubs heavily contribute to graph connectivity.

Small-world networks

Networks that are scale-free are usually also *small-world networks*, where paths between any two vertices are very short and the diameter small.

This characteristic has been found to be natural for social networks [25] and can be explained by the presence of a *community structure*, where vertices form subsets that are densely connected internally, but have only lose connections to other such subsets, and the existence of hubs, that due to their high degree tend to have connections into a large number of communities. It follows that paths between vertices are short within communities due to their density, and between communities due to the presence of hubs connecting the communities.

Furthermore, and for the same reasons, the diameter in small-world networks tends not to scale with the size of the network [16]. In our networks, the effect of this property can be observed by comparing the metrics of our smaller development and larger production graph: Despite varying in size by factor 4, both networks have nearly identical diameters of 21 and 20, respectively.

In practice, this property means that usually there exists a path between any two entities and the shortest path between them is short regardless the overall networks size.

[31] The complementary cumulative distribution (CDF) is given by $CDF(x) = \mathrm{P}(X \geq x)$, i. e. the fraction of vertices having a degree greater or equal a given degree x. Plotting power laws using the complementary CDF is a common technique and results in visually smoother graphs than plotting the frequency distribution directly.

5 The Visual Interactive System

This chapter deals with concepts of the visual interactive system that allows user to explore and analyse the data. Three topics are central for the design and development of Networks of Names:

- Visualization of the underlying data in a way that supports and reinforces human cognition. This includes the mapping of data elements to visual elements on screen as well as algorithms directly related to the visualization.
- An interface for human-computer interaction. That is, means for the user to interact with the system in order to drive the exploration and analysis process.
- Algorithms for the automation of tasks that are challenging for humans.

Sections 5.1, 5.2 and 5.3 discuss how these aspects are addressed in Networks of Names. The system also employs a classifier that is trained by user interaction and applied during the operation of the software. The classifier is addressed in detail in Section 6.

5.1 Visualization

The core part of the Networks of Names frontend is the visualization of the the underlying network as a *node-link diagram* [59]. A node-link diagram is a common visual representation for graphs that draws vertices of the graph as circles or some similar shapes (*nodes*), and edges as lines of some shape, e. g. straight or elliptic (*links*). Nodes may contain additional information about the corresponding vertex, such as the name of the entity or other attribute information. Links are commonly equipped with arrows to signal the edge direction and labelled with text to specify edge semantics.

An alternative visual representation for networks is matrix-based. However, node-link diagrams are less abstract and believed to be more intuitive [22] and is thus more suited for our goals.

5.1.1 Visual Mappings

The assignment of data properties to their visual representation on screen is referred to as a *visual mapping* [12]. To be suitable for the task, mappings need to represent the data correctly (i. e. not misrepresent it) and convey sufficient information contained in the data to allow the user to understand it, draw conclusions, and make decisions. At the same time, the amount of information in the visualization must be low enough to be manageable by human perception.

Entities

Entities of the social graph are mapped to nodes of the node-link diagram (see Figure 7). A node contains the entity name as a label in the centre of the node and the entity type encoded by colour. In addition, because the visualization shows only subgraphs of the entire network, each node contains a number that indicates the number of neighbours that currently do not appear in the visualization.

For the encoding of the entity type, the colour choices are blue for people and orange for organizations. This choice is not overly important for the overall design of the implementation. It avoids the rather frequent case of red-green colour blindness of potential users, but does not exclude the possibility of users not being able to differentiate between colour encoding altogether. A common solution to this problem is to use more than one visual mapping for one piece of information (in this case the entity type), such as an additional symbol or a different node shape. However, we refrain from doing so for two reasons: First, this information is “nice to have” for the assessment of a social structure and its presentation, but not crucial for the exploration of the graph. Second, the information is already naturally encoded in entity names, because users usually have sufficient human knowledge about the world to accurately differentiate between people’s names and organization’s names.

Nodes, regardless of entity type, appear in three visual states:

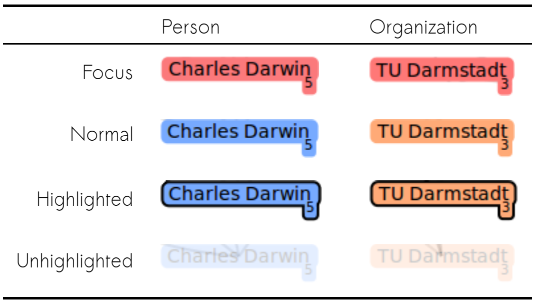


Figure 7: Visual states of nodes.

- “Focus”: Right after the user’s initial search, the node matching the user’s search term directly is highlighted in red. The focus state transitions into the “normal” state within the first few seconds after the node appears.

  The focus state is meant to allow the user to identify the direct hit to his search term quickly in the resulting visualization. The deviating colour of a single (target) element allows humans to quickly spot the element among other (distractor) elements [67].

- “Normal” for the usual representation of nodes.

  This state displays the node without any special highlighting. This is how nodes are normally represented when the user is not interacting with the visualization in any way.

- “Highlighted”: If the user hovers a node, the node is a neighbour of a hovered node, or incident to a hovered link, it appears in the highlighted state.

  The highlighted state is meant as a way for users to visually highlight a part of the graph they are currently investigating. Highlighted nodes are represented with a black stroke around them. There is no difference to the normal state in terms of colour, saturation or opacity. If some part of the graph is highlighted, the rest of the graph becomes “unhighlighted” (see Figures 9 and 10).

- “Unhighlighted”: If the user is highlighting parts of the graph and a node is not part of the highlighted subgraph, it becomes “unhighlighted”.

  Unhighlighted nodes appear with reduced opacity.

Relationships

Relationships between entities are mapped to links of the node-link diagram (see Figure 8). A link that connects two nodes indicates an existing relationship between them. If the relationship is labelled, the label appears as text on the link.

Links are drawn as elliptical arcs, because this corresponds to Lombardi’s drawings, which rely predominantly on curves instead of straight lines (although all our ellipses follow the same arc, while Lombardi was more flexible). In practice, this also helps the layout assume more convex faces, which is an aesthetic criterion usually conjectured for node-link diagrams [7]. Link colour is set to be grey. Although black would be a valid alternative, the high contrast to the background tends to draw attention away from the graph overall towards the lines. Instead, black is used to highlight links (and nodes). Link opacity is set to correspond to the significance value of the link (for more on the significance metric, see Section 5.3.2), with a minimum value to prevent links of low significance to become close to invisible.

Analogously to nodes, links appear in different visual states: “Normal”, “highlighted”, and “unhighlighted” (see Figures 9 and 10 for practical examples).

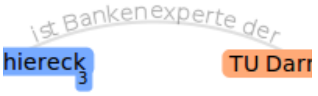
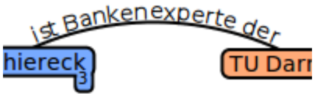
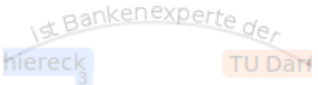
	Link
Normal	
Highlighted	
Unhighlighted	

Figure 8: Visual states of links.

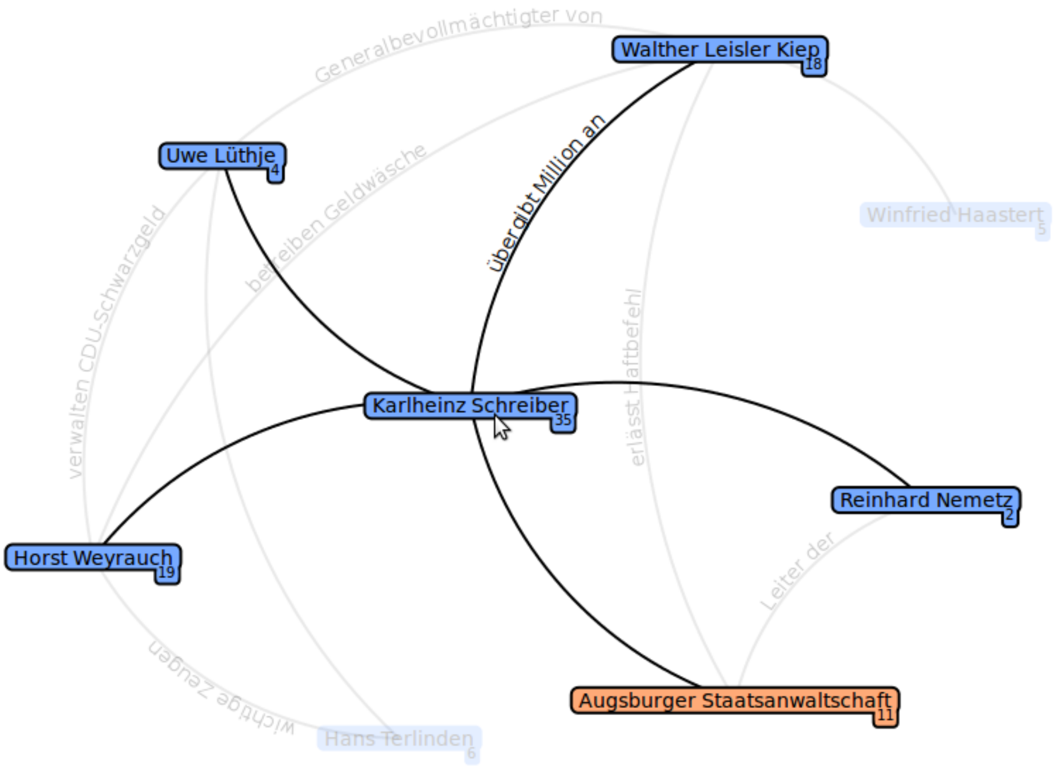


Figure 9: Visualization with highlighted node.

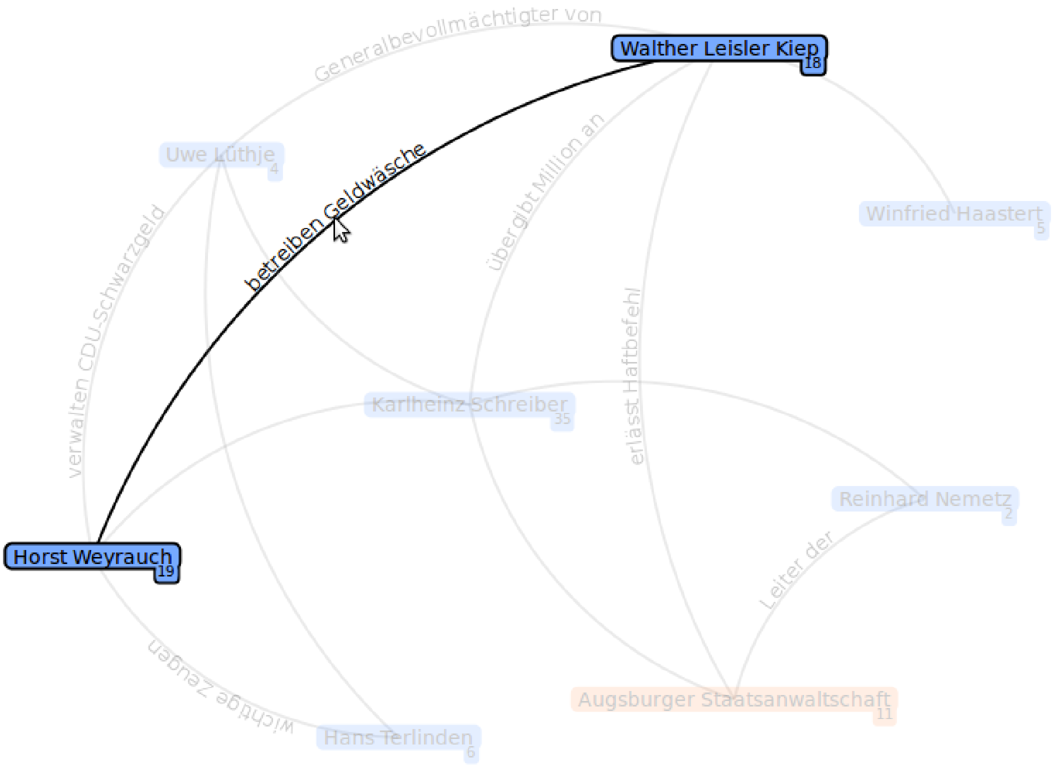


Figure 10: Visualization with highlighted link.

5.1.2 Layout

For the layout, we rely on the force-directed layout that is included in D3.js[32]. The layout can be configured through a number of parameters. We determine good values for the parameters by experimentation and arrive at good results for a `linkDistance` of 300 (the value that determines the eventual distance between two connected nodes) and a high `charge` of -1000 (the strength of the repulsion between nodes). We leave the remaining values at their defaults.

This values help to capture visualization several heuristics for graphs [7], namely the even distribution of nodes on screen and a uniform edge length. While the algorithm performs reasonable to prevent node overlapping most of the time, the overlapping of edges is comparatively common. In many cases, this problem can be corrected by user interaction. However, in the presence of several highly connected nodes, the elimination of edge crossings is not always possible.

The layout algorithm is executed for the first few seconds after nodes appear. This allows for nodes to settle into their approximate final positions (a complete equilibrium may never be reached in force-directed layouts). The layout is then stopped, but can still be changed by manually dragging nodes.

In order for the algorithm to run, nodes need to be assigned initial positions on the screen. One possibility is to assign all of them the same position, e. g. at the corner of the visualization or in its centre. With a high repulsion between nodes, however, this leads to an extreme initial outward motion, which makes the layout process turbulent and long. An alternative is to distribute nodes randomly. This prevents problems that arise with positioning all nodes at the same point. However, randomness has the effect that even the repetition of identical searches produces completely different layouts, which counteracts users' expectations and can slow down their ability to perceive the information drawn onto the screen.

As a consequence, we create a solution where entity names are hashed onto (x, y) coordinates. Doing this, every node is placed onto the same position every time it appears on screen. This results in identical searches arriving at nearly identical layouts (small fluctuations between runs can occur due to implementation details of the layout algorithm) and similar searches to show at least similar distributions of nodes (although the combination of initial nodes has significant impact on the layout process). The effect is shown in Figures 11 and 12 on page 35. The first figure shows a graph of the neighbourhood of the Darmstadt University of Technology, the second figure the neighbourhood of Dirk Schiereck, a professor at the university. Both subgraphs share several nodes. The initial deterministic positioning of nodes causes several similarities in the layout of the two graphs: Although details differ, in both cases, "TU Darmstadt", "Dirk Schiereck" and the "European Business School" appear at the bottom and in similar layouts. In both graphs, "Johannes Buchmann" appears above "TU Darmstadt" towards the left, while "Erik Tews" is aligned towards the right.

5.2 Interaction

Networks of Names is designed to be a tool for the general public. We explicitly aim to make information accessible that is publicly available, but hard to process in its entirety for a single person. For that reason, no special background knowledge should be necessary to operate the tool, and we usually select good defaults and ease of use over customization.

As a result, the user interaction in Networks of Names focuses not on the configuration and parametrization of the tool, but on decisions related to the exploration and analysis process itself.

Conducting Searches

By selecting *New* from the menu, the user can conduct custom searches. The modal dialogue for entering search parameters is shown in Figure 13. Two possibilities exist for that (organized in the tabs "Relationship" and "Name"): Entering two names lets the user search for a connection between two entities (details on how this is done algorithmically are discussed in Section 5.3), while entering

[32] `http://github.com/mbostock/d3/wiki/Force-Layout`

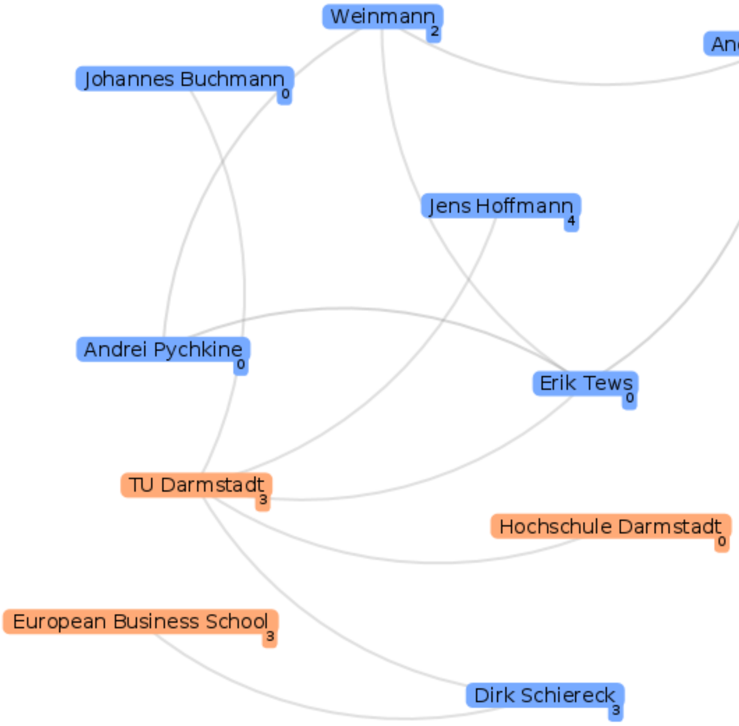


Figure 11: Initial layout for the search for *TU Darmstadt*, the Darmstadt University of Technology.

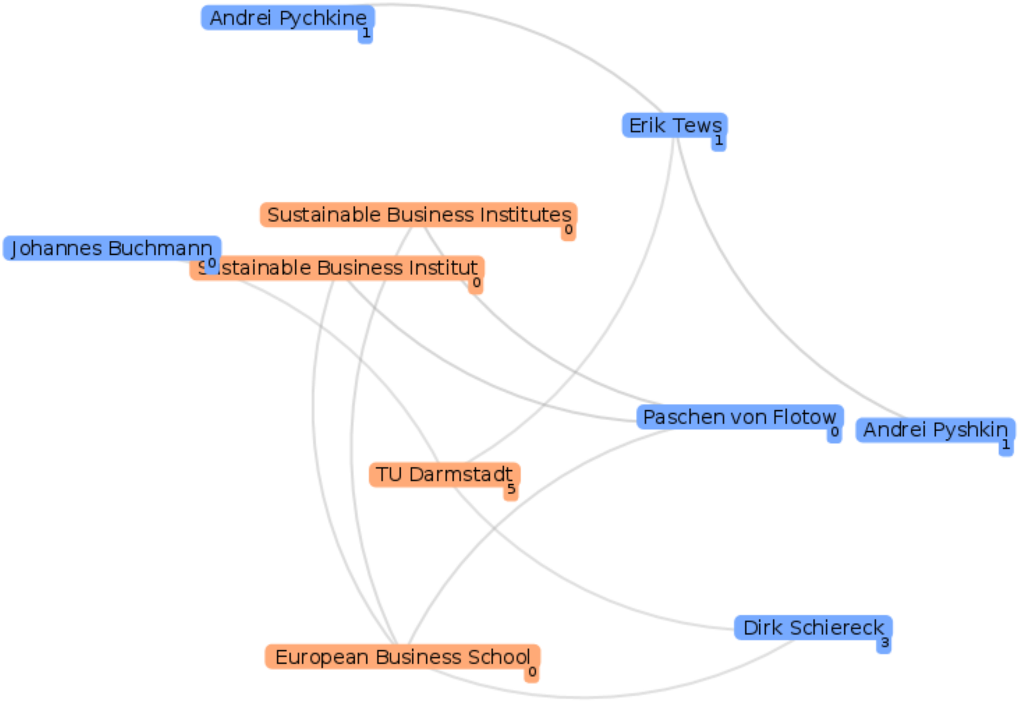
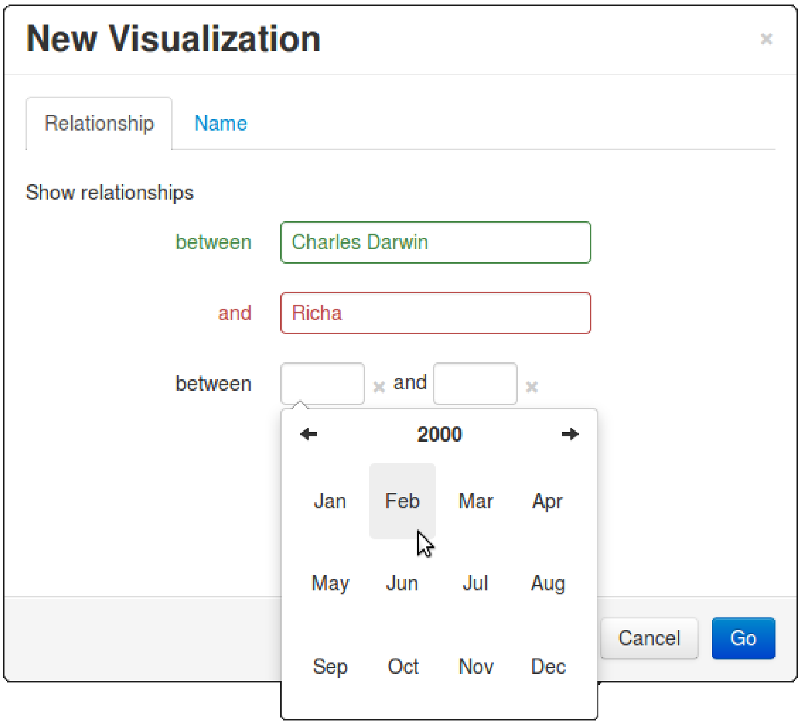


Figure 12: Initial layout for the search for *Dirk Schiereck*, a professor at the Darmstadt University of Technology.



New Visualization

Show relationships

between Charles Darwin

and Richa

between and

Figure 13: The *New* dialogue.

only one name searches for a single entity and its relationships. The input fields are equipped with an autocomplete feature. If a name entered into the field exists in the database, the field turns green (seen in the first input field in the figure), otherwise it turns red (seen in the second input field in the figure). In addition to entering names, the user can limit the timeframe by selecting a year and a month from a calendar. Clicking on "Go" conducts the search and causes the system to submit the search request and subsequently visualize the result.

Interacting with the Layout

The user has several state-of-the-art possibilities to affect the layout and adjust the viewport:

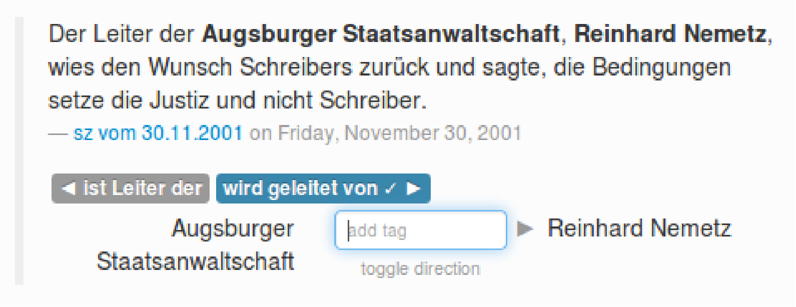
- Pan and zoom by clicking and dragging empty areas or turning the mouse wheel, respectively.
- Drag and drop nodes by performing a drag gesture on them (click on the node, drag to the intended position, and release the mouse button).

Interacting with Nodes

Apart from moving nodes around, the user can summon a context menu on nodes, by performing a right click. The context menu contains three actions:

- *Expand More* to show up to five additional neighbours of this node (if they exist). New nodes are expanded in order of significance. Nodes that were previously removed by the user, are not expanded by this action.
- *Expand...* to view a list of neighbours and possibly expand some of them explicitly. This action can also be used to re-expand nodes that were previously removed.
- *Remove node* to remove a node and its incident links from the visualization.

If new nodes are expanded, they are layed out using the force layout algorithm, while nodes that were already present remain at their positions.



Der Leiter der **Augsburger Staatsanwaltschaft**, **Reinhard Nemetz**, wies den Wunsch Schreibers zurück und sagte, die Bedingungen setze die Justiz und nicht Schreiber.

— sz vom 30.11.2001 on Friday, November 30, 2001

Figure 14: A sentence in the sources view (“The head of the Augsburg prosecution, Reinhard Nemetz, rejected Schreiber’s wish and remarked that the judiciary, not Schreiber, would be setting terms.”), with tags (“is head of”, “is lead by”) for the relationship between “Augsburger Staatsanwaltschaft” (Augsburg prosecution) and Reinhard Nemetz and the tagging widget.

Expanding and removing nodes is used to navigate the graph. In addition, the removal of nodes can be utilized to hide nodes that contain artefacts from the automatic extraction, such as only parts of a name or misspellings.

Interacting with Links

Left clicking on a link brings up the sources view that can be seen in Figure 15 on page 45. Performing a right click on a link opens a context menu that contains three actions:

- *Select label* lets the user select one of the labels associated with the relationship from a submenu. In this submenu the user can also choose to remove labels if he thinks that they are certainly incorrect.
- *Hide label* lets the user have no label displayed on the link.
- *Remove edge* removes the link from the visualization.

Interaction with links affects and is affected by the creation and validation of tags (interactions that happens within the sources view). Specifically, selecting a label triggers marking all tags of that relationship that bear the same label as correct. Analogously, removing a label (from the *Select label* submenu) triggers all tags with this label to be marked incorrect.

Creating and Validating Tags

In the sources view, the user can review sentences from which a relationship was extracted. The sources are grouped by similarity into clusters and sorted chronologically (more on how clustering is implemented follows in Section 5.3.3).

Every sentence is accompanied by its source and publication date, as well as a tagging widget and possibly tags that have already been entered by other users or were automatically created by the system. An example is shown in Figure 14.

A tag contains four pieces of information: First, a label that acts as a name or description of the relationship. Second, the direction of the relationship. A direction can have three different values, namely left-to-right (represented by an arrow next to the label pointing to the right), right-to-left (represented by an arrow next to the label pointing to the left) or undirected (represented by the presence of both arrows next to the label). Third, whether it was entered by a user or created automatically. User-created labels are blue, while automatically created labels are grey. Fourth, a checkmark next to the label that denotes whether the label has been accepted as correct by a user or not. For user-created tags, the checkmark is always present.

Users can right click on a tag to open a context menu. It contains three possible actions:

- *Set this label* to make a label appear on the corresponding link in the visualization.

- *Confirm label* to accept the tag as correct.

  This action is used to accept auto-created labels as correct. If a user selects this option, the system adds a checkmark to the corresponding tag and notifies the server to persist the user input. Additionally, the system extrapolates the users intent to all other tags in the same relationship that bear the same label.

- *Remove label* to reject a tag.

  This action is used to mark auto-created labels as wrong. Selecting the option removes the tag from the interface and notifies the server to persist that change. Additionally, the system extrapolates the users intent to all other tags in the same relationship that bear the same label.

Creating tags can be achieved using the widget under each sentence. For that, a user selects the direction of the tag by repeatedly clicking “toggle direction”, enters a label and presses Enter.

For the user, the creation of tags serves the purpose of generating labels that can be used on links of the visualized graph. The system, on the other hand, has the possibility to utilize user-created tags to learn sentence patterns, thereby training the classifier to recognize and auto-tag similar relationships (Section 6 contains more on how the classifier works).

Accepting and rejecting tags is a way to evaluate classifier performance. Since relationships in Networks of Names can have arbitrary semantics (decided by the user), we have little or no possibility to automatically validate classifier performance. Users can, however, decide whether a tag created by the classifier is correct or not. This user input, once performed sufficiently often, can be used to calculate a precision for the classifier (evaluation results on the performance of the classifier are given in Section 7.3).

5.3 Automatic Analysis

Several tasks in Networks of Names are performed by algorithms, with little or not user interaction. This approach is taken when a computer is better suited to perform a certain task than a human, or the amount of data is too large for a human to handle. The most computationally intensive task in Networks of Names is the processing of corpus data into a social network that serves as the basis for the visualization. This was discussed in Section 4.2. This section focuses on tasks that are performed during the use of the system. Such tasks are usually triggered by certain user actions, but then performed without the need for the user to understand or configure them in detail.

We have established in Section 1.3.2 that the visualization mantra suitable for our use case of large networks is “Search, Show Context, Expand on Demand” [63]. This mantra relies heavily on automation, since at least searching and (initial) showing context cannot be achieved without computer assistance. Our approach to these problems is discussed in Sections 5.3.1 and 5.3.2.

Another frequent user activity that requires algorithmic aid is viewing the sources that constitute the origin of a relationship. For frequent relationships, the number of sources can be large and thus difficult for humans to assess. To reduce the text load on users, we cluster similar sentences and display only a few representants of every cluster. Details are given in Section 5.3.3.

5.3.1 Search

Searching is typically performed by the user entering a search term and the system replying with a list of matching results. However, in general, other possibilities of searching large databases exist.

Methods of Searching

The possibility of *faceted search* has been suggested by van Ham and Perer [63] and others, where (possibly in addition to keyword search) the user interest is expressed in relation to attribute information available for the vertices and edges of the network. For instance, a user could require only edges that represent relations of a certain kind (such as politics, sports, or fashion), or only vertices above a given frequency threshold.

Yet another approach is to search the network for structural properties instead of keywords or semantics. Networks have been shown to have recurring and statistically significant sub-graphs called *motifs* [65]. Motifs have been suggested to imply certain properties about their elements, though contrary opinions exist (see, for instance, [33]). In social graphs, motifs can provide insights about the interdependence of entities. A popular example is a star motif, where one entity is connected to a number of neighbours, while the neighbours are not interconnected. This structure suggests a central role of the well-connected entity, which is responsible for the cohesion of its neighbourhood, and could thus exhibit special properties, such as the capability to influence all its neighbours or disconnect them.

Lastly, although for large networks it is not possible to reasonably show a complete overview, for certain use cases, a simplified visual overview may be a viable option, highlighting selected features of the graph or the underlying data. In [63], van Ham and Perer give a scatterplot of specific aspects of the data as an example, although this example bears little connection to our (or their) use case. A possible example for such an alternative simplified overview of large networks is given by Yang et al. in [68]. Their system *PIWI* uses tag clouds to visualize communities in a network, with small matrix plots to denote their neighbourhood relationships.

Searching Entities in Networks of Names

In Networks of Names, searching for entity names by directly entering the name is an obvious choice. This corresponds to a simple keyword search.

Once the user enters the entity names and submits a form, the name is used to perform a database query for the vertex' ID in the graph. For this use case, the database is indexed to allow fast access by entity names.

Searching Relationships in Networks of Names

Since we create and visualize a network of social relationships, it seems natural to support the search of relationships between two people and/or organizations.

For this, a user can enter two entity names. The system retrieves the unique IDs of both entities analogously to the search for one entity. The system is now required to select a suitable path that connects the two entities. Once such a path is found, the two initial vertices and the vertices on the path between them constitute an initial result to the user query.

Selecting a “good” path between two vertices has no obvious solution, however. It has been suggested that simply selecting the shortest path between two entities in a social graph might not produce the best results [30]. The reason of this is the limited expressiveness of path length in social graphs: Due to the small-world property, all distances between two entities are commonly very short, often even just one or two hops. As a consequence of the overall shortness of distances between any two entities, it is common that several shortest paths of equal length exist. In this case, it is impossible to use path lengths to select one path over another. Furthermore, the mere length of a path does not express the strength of the tie; a longer, but stronger connection may be better to describe the relationship.

We therefore define a measure for significance of relationships (we use *normalized pointwise-mutual information*, introduced in Section 5.3.2 below), and construct *maximum-capacity paths*, i. e. paths where the capacity of the minimum-strength edge is maximal, setting edge capacities to their significance values.

For that, we use a modified Dijkstra algorithm that keeps track of the capacity (instead of length) of paths. Although other, and also more efficient, algorithms exist to solve that problem [48, 38], the Dijkstra implementation is concise and proves efficient enough in practice. Since we find the plain version of this algorithm to produce rather long paths, we implement it to round capacity values to two decimal points. This prevents small differences in capacity to have an impact on results. Lastly, we configure the algorithm to select shorter paths, given the same capacity.

Timeframe

We implement faceted search in that the user can select a timeframe of interest. When answering user requests, the timeframe is respected by creating a filtered view onto the graph, using timestamps associated with every relationship – and thus the edges in the graph. The filtered view is a subgraph of the complete graph, induced by all eligible edges.

5.3.2 Show Context

During the Search step, the users enters a search term that results in a focal vertex y. However, just the focal vertex is not enough to constitute a meaningful result. Apart from the focal vertex, the user needs to receive a context that puts the vertex into perspective. For that, an interesting subgraph needs to be extracted.

More formally, given a user search for vertex y and a large graph G, we want to extract a connected subgraph F of G that contains y, has size n and maximal total interestingness. The computation of F should be sufficiently fast and n should be a suitable size for transmission to the client and visualization.

Degree of Interest for Vertices

Such a measure of interestingness is usually referred to as a *degree of interest* (DOI) value. While Furnas introduced DOI as a measure for trees [21], van Ham and Perer generalize the notion of DOI to a formula that applies to graphs in general (i. e. not only trees) [63]. They define DOI as

$$\mathrm{DOI}(x|y,z) = \alpha \cdot \mathrm{API}(x) + \beta \cdot \mathrm{UI}(x,z) + \gamma \cdot \mathrm{D}(x,y) \tag{1}$$

where x is the vertex of which the DOI is to be evaluated, y the focal vertex, and z a user-selected facet; API defines an *a-priori interest* of the vertex, UI the *user interest* in regard to the search facet, and D the *distance* from the focal vertex; the factors α, β, and γ can be used to weigh the components.

To adapt to the specifics of general graphs, van Ham and Perer expand their definition of DOI further: In trees, DOI would typically be defined in a way that DOI(x) is always smaller than DOI(r) iff x is a descendant, i. e. in a subtree of, r. However, in general graphs, will have several local maxima with respect to DOI values. As a result, methods that work on trees cannot be applied to general graphs. For instance, thresholding the DOI value would yield a large number of disconnected subgraphs (as opposed to cutting off subtrees) and applying local search algorithm (like the one presented below) to the resulting forest would, for the lack of connectivity, not be able to extract meaningful subgraphs.

To counteract this problem, van Ham and Perer modify the API function to "diffuse" the interestingness of vertices over the graph, so that not only the vertex' intrinsic interestingness, but also the interestingness of neighbouring (possibly more interesting) vertices, is taken into account. They define

$$\mathrm{API}_{\mathrm{diff}}(x) = \max(\mathrm{API}(x), \delta \cdot \max(n \in \mathrm{neighbors}(x) : 1/\mathrm{EI}(e,x,n) \cdot \mathrm{API}_{\mathrm{diff}}(n))) \tag{2}$$

where EI > 0 defines an edge disinterest function that is higher for less interesting edges and used to reduce the flow of interestingness over low-interestingness edges, while the factor δ controls how strong diffusion is overall. The intuition is to replace low DOI values by a fraction of high DOI values of neighbouring vertices, so that a local algorithm is driven towards high DOI vertices even if it has to pass low DOI vertices.

The introduced complexity by "diffusing" API values through the graph can be counteracted by the fact that API does not depend on user-defined and search-specific parameters and can thus be precomputed.

Original Algorithm

Using the degree of interest measure for graphs, it is possible to answer user queries by selecting a suitable subgraph of maximum interestingness. A naive approach would be to compute the total degree of interest of all subgraphs of some fixed size n that contain the user's focal vertex y. Per definition, the graph with the highest total DOI would be the most interesting graph. This approach proves infeasible for large graphs. As a result, van Ham and Perer suggest a greedy and more efficient algorithm that we reuse in Networks of Names. The algorithm does not necessarily produce an optimal solution, but is suitable to compute good solutions quickly. In the context of an already fuzzy definition of "interestingness" in itself, non-optimality is acceptable, and a greedy algorithm is a good trade-off between speed and optimality:

Using a priority queue of candidates that initially contains y, the algorithm pops the candidate with the highest DOI, includes it into the selected subgraph, and adds all its immediate neighbours to the queue. This is repeated n times or until there are no candidates left and the resulting subgraph, that is the graph induced by all selected vertices, is returned.

Degree of Interest for Edges

Given the idea of a DOI measure and an algorithm that utilizes it to calculate a context for a search, it is left to define the function API, UI, and D.

We treat and implement the selected timeframe as a hard constraint, i. e. we simply filter sources that do not respect it. Since this is our only facet, we do not require a soft parameter and define $\beta = 0$ and $UI = 0$.

The a-priori measure API is the core part of the DOI measure and is claimed to often have natural interpretations. In our graph, it is tempting to use the entity frequency (i. e. how often it is mentioned in the corpus). The distance from the focal vertex can also be easily interpreted as a shortest path distance in the graph.

Thus, following Equation 1, a possible definition for some vertex x and the focal vertex y would be

$$\begin{aligned} \mathrm{API}(x) &= \mathrm{frequency}(x) \\ \mathrm{D}(x,y) &= -(0.5^{\mathrm{d}(x,y)} \cdot \mathrm{frequency}(x)) \end{aligned}$$

where $\mathrm{frequency}(x)$ denotes the frequency of the entity represented by the vertex x and $\mathrm{d}(x,y)$ is the shortest path distance between x and y. That means that a vertex' a-priori interest is its plain frequency in the dataset. In the definition of D we reduce the a-priori interest by fractions of itself, because the frequency value is arbitrarily high, so that subtracting fixed values would have much larger impact on low-frequency vertices and nearly none on high-frequency vertices.

Setting $\alpha = \beta = \gamma = 1$, $\delta = 0.5$ and leaving the edge disinterest function EI aside for simplicity (we use a value derived from pointwise mutual information discussed below), the definition of DOI is complete.

However, in practice, we find this model to produce very uniform search results regardless the search: A few entities have extremely high frequencies, while many entities have comparatively low frequencies. This discrepancy can be attributed to the small-world property discussed in Section 4.3 and has an adverse impact on result quality of the greedy expansion algorithm for both high- and low-frequency entity searches. High-frequency entities include influential politicians like Angela Merkel as well as large organizations like the European Union, international corporations, central banks, national telecommunication companies, or political parties. Those entities usually also have connections between each other and form complete high-frequency clusters that tend to be expanded "as a whole", although from a user perspective, such connections are not "interesting" per se. For searches of low-frequency entities, such high-level entities and their clusters form "honeypots" in the graph that "lure" the greedy expansion algorithm: it tends to find such "honeypots" quickly and proceeds to expand their supposedly "interesting" neighbourhood instead of focusing on the original search. Thus, searches for vertices within a high-frequency cluster and searches for vertices in the immediate vicinity of such a cluster produce nearly identical results, which seems inappropriate for the exploration of our network.

This effect can be observed frequently to a degree that any two searches, given both entities are found in the same connected component of the graph, produce heavily overlapping results. We assume two main reasons for this:

- Due to the small-world property, the expansion algorithm can find high-frequency clusters starting from any focal vertex.
- The diffusion of API values adds to the problem by creating additional “lures” for the expansion algorithm, thereby acting against the original intention to promote nearby “interesting” vertices.

Our attempts to mitigate this effect by adjusting parameters, by logarithmically flattening frequencies, or by not diffusing API values did not have significant impact on the overall problem.

We therefore changed the focus from the interestingness of vertices to the interestingness of edges, aiming for a measure that is capable of expressing the significance of a connection between two vertices by a value independent of the entity’s or relationship’s actual frequency values. Such a measure could be used to expand a subgraph around the focal vertex by assessing the interestingness of incident edges instead of adjacent vertices.

Based on the definition in Equation 1, we adapt the definition of a degree of interest function for vertices to define a degree of interest measure for edges:

$$\mathrm{DOI}_{\mathrm{edge}}(\{u,v\}|y,z) = \alpha \cdot \mathrm{API}_{\mathrm{edge}}(\{u,v\}) + \beta \cdot \mathrm{UI}_{\mathrm{edge}}(\{u,v\},z) + \gamma \cdot \mathrm{D}_{\mathrm{edge}}(\{u,v\},y) \tag{3}$$

where $\{u,v\}$ is the edge of which the DOI is to be evaluated, y is the focal vertex, and z a user-selected facet; API defines an *a-priori interest* of the edge, UI the *user interest* in regard to the search facet, and D the *distance* of the edge from the focal vertex; the factors α, β, and γ can be used to weigh the components.

Pointwise Mutual Information

To define the components of our edge-oriented degree of interest, we require a notion of the interestingness of edges. The advantage over a measure based on vertices is the higher amount of properties of an edge, namely the properties of the edge itself and the properties of two incident vertices, that can be used to calculate it. In our network, we have frequencies of vertices, i. e. how often entities appear in the underlying data, as well as frequencies of edges, i. e. how often two entities co-occur.

Borrowing from information theory, we express the a-priori interest of edges by calculating the *pointwise mutual information* (PMI). Given two random variables X and Y, and two possible outcomes $X = x$ and $Y = y$, PMI describes whether x and y are independent (PMI is zero), positively correlated (PMI is positive), or negatively correlated (PMI is negative).

For our use case, we interpret “$X = x$” as an event that an entity x is seen in a sentence, and “$X = x$ and $Y = y$” as an event that the entities x and y co-occur, i. e. appear together in a sentence and thus share a relationship. Transferred to graphs, PMI of an edge is defined as

$$\mathrm{pmi}(\{u,v\}) = \log\left(\frac{f_{\{u,v\}}}{f_u \cdot f_v}\right)$$

where $\{u,v\}$ is the (undirected) edge between vertices u and v, $f_{\{uv\}}$ is the relative frequency of the relationship represented by that edge (which is the frequency of the vertices co-occurring), and f_w for $w \in \{u,v\}$ the relative frequency of the entities represented by the respective vertices.

The upper bound of pmi depends on the respective frequencies. Due to this lack of a fixed upper bound, it is not possible to tell from the value of pmi how far it is from perfect independence or correlation. To make the values comparable, we use the *normalized pointwise mutual information* [9], defined in our context as:

$$\mathrm{npmi}(\{u,v\}) = \frac{\mathrm{pmi}(\{u,v\})}{-\log(f_{\{u,v\}})}$$

As opposed to pmi, the evaluation of npmi results in values between -1 and 1, with -1 indicating that two elements never co-occur (hence, -1 is a lower bound and never actually reached), 0 indicating independence, and 1 complete co-occurrence.

For easier handling, we define a npmi variant that produces positive results only, by changing the interval of npmi from $[-1,1]$ to $[0,1]$:

$$\mathrm{npmi}^{+}(\{u,v\}) = \frac{\mathrm{npmi}(\{u,v\})+1}{2}$$

It is now left to define the elements of $\mathrm{DOI}_{\mathrm{edge}}$. Again, we treat the search facet as a hard constrain and define

$$\begin{aligned} \mathrm{API}(\{u,v\}) &= \mathrm{npmi}^{+}(\{u,v\}) \\ \mathrm{D}(\{u,v\},y) &= -(0.5^{\mathrm{d}(\{u,v\},y)} \cdot \mathrm{npmi}^{+}(\{u,v\})) \end{aligned}$$

where $\mathrm{d}(\{u,v\},y)$ denotes the shortest path distance of the edge $\{u,v\}$ to the focal vertex y and should be read as $\min(\mathrm{d}(u,y),\mathrm{d}(v,y))$. We weigh the elements equally by setting $\alpha = \beta = \gamma = 1$.

Our Algorithm

According to our focus on edges, instead of vertices, we adapt the original algorithm from [63] to expand by degree of interest of edges instead of vertices:

The set of selected vertices S initially contains the focal vertex y. We maintain a priority queue Q ordered by our edge-DOI that contains all eligible edges, initially filled with all edges incident to y. In every step, the algorithm pops the edge with highest DOI from the queue and adds the endpoint that is not yet in S to S. All incident edges of the selected vertex are then added to Q. This is repeated until S reaches the desired size n or until no candidates are left. The resulting subgraph, i. e. the subgraph induced by all vertices in S, is returned.

Low-frequency bias

PMI exhibits a bias towards low-frequency relationships [9]. As a result, our algorithm tends to expand low-frequency high-correlation relationships before similar high-frequency relationships. To a degree, it is possible to counteract, but not completely remove, this problem, for instance by defining:

$$\mathrm{pmi}_2(\{u,v\}) = \log\left(\frac{(f_{\{u,v\}})^2}{f_u \cdot f_v}\right)$$

$$\mathrm{npmi}_2(\{u,v\}) = \frac{\mathrm{pmi}_2(\{u,v\})}{-\log((f_{\{u,v\}})^2)}$$

We choose, however, not to over-optimize the measure at this point, partly because it yields good results in most cases, and ultimately, it is not arbitrable by statistical measures and with complete confidence which relationships are the most "interesting".

Hubs

We have established in Section 4.3 that our network contains hubs, i. e. vertices of very high degree. Regardless of whether we use DOI for vertices or edges, low-quality expansion can be caused by the presence of such hubs. Once expanded, they introduce a large number of candidates for further expansion. Some of the candidates are likely to have high PMI values with the hub and draw the expansion away from the original search.

An example of hubs are political parties. They tend to have high degrees because they share relationships with all or most of their members. However, relationships of people by mere membership in the same, possibly large, political party are usually not as interesting as direction relationships or relationships via other people.

To favour direct connections over connections via hubs, we stop expansion at high-degree vertices. Once such a vertices has been selected, its neighbours are not added as candidates for expansion.

Summary and Discussion

By redefining and adapting the original degree of interest function and algorithm from [63] to operate on edges instead of vertices and calculating pointwise mutual information to determine the interestingness of edges, we are able to generate appropriate and subjectively "interesting" subgraphs as context for a user search.

While van Ham and Perer successfully apply their vertex-based DOI measure for a legal citation network, it proves inappropriate in our social network graph of public figures and organizations. We thus conclude that the focus on edges or vertices may be appropriate to assess the interestingness of subgraphs depending on graph structure, semantics and use case. In this matter, van Ham and Perer [63], but also others [40], incorporate a notion of global interestingness into their approach. In our case, however, globally interesting nodes are nodes such as "Angela Merkel" that due to the small-world property are close to users' initial searches regardless the details of the search. Thus, for our usecase, where user conduct seaching for specific parts of the graph, guiding them towards globally interesting parts seems counterproductive.

For similar reasons, we refrain from "diffusing" DOI values through the graph, because this method draws attention away from the focal vertex, which in the search for social connection around that vertex appears to reduce subjective quality of results.

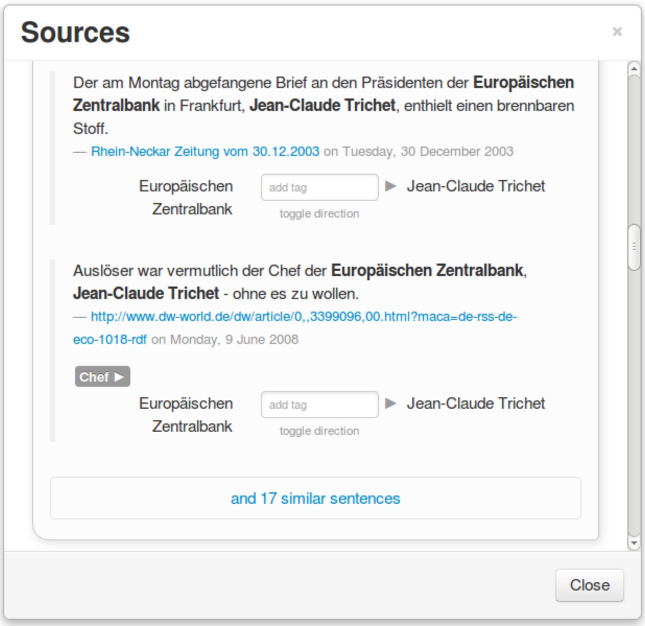
Not diffusing DOI values, and due to the fact that pointwise mutual information can be efficiently calculated as it is needed, we arrive at a solution that allows fast and efficient expansion without the need to precompute and store parts of our degree of interest measure.

5.3.3 Clustering of Sources

The user has the ability to view sentences, along with their dates and sources, from which relationships in the network originate, by selecting the respecting edge in the visualization. Since high-frequency relationships are attributed to hundreds, up to thousands of sentences, we sample sentences and cluster the resulting sample by similarity, displaying only a few sentences to represent each cluster and hiding the rest.

This way, we reduce the load of text on the user and practice a kind of result set diversification by displaying sentences that are likely to deal with different topics (for details on result set diversification, see, for instance, [64]).

For each cluster, three sentences are shown and the rest is hidden. An example can be seen in Figure 15. The three representants are selected as follows: The first (not seen in the figure) is the earliest source for the cluster. If a cluster coherently and completely represents a semantic relationship between two entities, this source marks its first occurrence. The second is selected by searching for a sentence that contains one, two or three words between the names of the two entities. By this, we attempt to prominently present sentences that could result in good sentence patterns for the classifier. The third sentence is ideally a sentence that has been assigned an automatic tag that was not yet confirmed or



Der am Montag abgefangene Brief an den Präsidenten der **Europäischen Zentralbank** in Frankfurt, **Jean-Claude Trichet**, enthielt einen brennbaren Stoff.
— Rhein-Neckar Zeitung vom 30.12.2003 on Tuesday, 30 December 2003

Auslöser war vermutlich der Chef der **Europäischen Zentralbank**, **Jean-Claude Trichet** - ohne es zu wollen.
— http://www.dw-world.de/dw/article/0,,3399096,00.html?maca=de-rss-de-eco-1018-rdf on Monday, 9 June 2008

Figure 15: The sources view, showing the relationship between the European Central Bank and its second president, Jean-Claude Trichet.

rejected by a user. If no suitable selection is found for the second or third representant, the missing number of sentences is added chronologically. The remaining sentences from the cluster are hidden behind a link “and *n* similar sentences”, that can be expanded by the user on demand.

Preparing the Data

In order to be able to cluster sentences, we need to find a suitable representation and measure of similarity. This is a typical task in information retrieval. Usually, information retrieval refers to *documents* and document collection. In our case, a sentence is a document and all sentences linked to a certain relationship are the document collection. For the representation of the sentences, we choose a common and thus straight-forward way in information retrieval and represent each sentence as a vector, where a vector element represents the presence of a word in the sentence and its value the importance of that keyword. We perform the following steps to transform the sentences into vector representation:

1. We create an ordered set of keywords k relevant to the collection of sentences. For that, all sentences from the collection are tokenized, stopwords are removed, and the remaining tokens are stemmed using the Porter Stemmer [47] for German. We remove stopwords, because they are unlikely to express actual semantic similarity of sentences, and because it reduces the number of keywords. We use a stemmer, a program that reduces word forms to their stem (that is not necessarily grammatical) for better comparison.

2. Each sentence is transferred to a vector representation v_i where v_{ij}, i. e. the j-th element of the vector, is tf-idf(k_j), where tf-idf refers to the product of *term frequency* and *inverse document frequency* [39]. The former is the frequency of a keyword in a sentence. The latter is a measure of how common the keyword is across all sentences, given as the logarithm of the fraction of the number of sentences in the collection and the number of sentences that contain the keyword.

Thus, high values of tf-idf are achieved by a high term frequency and a low document frequency. The resulting vectors can be compared using *cosine similarity*.

Markov Clustering Algorithm

For the clustering, we need an algorithm that meets two criteria:

- The number of clusters should be variable and automatically determined by the algorithm.
- The clustering should be flat, because hierarchies of sentences are of no use for our application.

We select the *Markov Clustering Algorithm* (MCL), developed by van Dongen [62], because it meets our requirements and implementations of it are available.

The algorithm is a graph clustering algorithm and operates by simulating *random walks* (or *flows*) on the graph and drawing conclusions on likely clusters, based on the graph clustering paradigm that states that “[a] random walk in [a graph] that visits a dense cluster will likely not leave the cluster until many of its vertices have been visited.” [62].

The algorithm can be parametrized to achieve different granularity of clustering, where the number of clusters does not need to be known in advance. We aim for a small number of clusters and determine suitable parameters by experiments.

Java-ML

As an implementation of MCL we use the implementation from the Java-ML library[33] [1].

Java-ML is a machine learning library that is designed to be extensible and easily usable in software development and research. It contains several algorithms for clustering and classification, databases of well-known example data for research, means of evaluation, as well as mathematical functions for filtering and normalization, feature selection, and calculating distances and similarity.

The library is comparatively well-documented and supports easy inclusion of own data formats (wrapped into a Java-ML representation). This aspect is important, because in our case, after the clustering, we need to be able to trace elements of clusters back to the data objects they represent. Many other libraries, on the other hand, do not offer this possibility or require significant extension to make it possible.

[33] http://java-ml.sourceforge.net/

6 Labelling Relationships between Named Entities

One aspect of Networks of Names is the discovery and labelling of relevant relationships between entities. However, while relationships between two entities in Networks of Names only exist if the entities appear together in at least one sentence (see Section 4), not all such sentences describe actual or significant semantic relationships. And even if semantic relationships exist between two entities, the semantics are unknown to the system.

Users can therefore view the source sentences for possible relationships they encounter in the graph. By assigning tags (consisting of a direction and a label) to relevant sentences, users can describe the semantic relationship between two entities of interest. Once a label is assigned to some source sentence of a relationship, it can be selected to appear as a textual label on the respective edge in the visualization.

By labelling sentences with respect to the relationship between two entities, users produce three sorts of interesting data:

1. They signal that a semantic relationship exists between two entities.
2. They select a sentence that contains or implies the semantic relationship between them.
3. They name the relationship by entering a label (and a direction).

We aim to use this data in order to train a classifier that discovers and automatically labels similar relationships.

6.1 Problem Field

The discovery of semantic relations in text is a fundamental task in natural language processing. To produce meaningful results, especially in the context of extensive user interaction, an automatic system processing natural language text must be able to correctly identify and understand semantic relationships between nominals in general, and between specific entities referred to by those nominals in particular.

This task is crucial for several areas of computer science, such as information extraction, ontology learning, the construction of language models, as well as several higher level tasks, such as machine translation, automatic text summarization, and question answering.

Historically, there are two perspectives onto the problem: On the one hand, the organization of knowledge for the construction of capable artificial intelligence systems, and on the other hand, the processing and understanding of natural language text. While the latter views the (preferably complete) understanding of the structure and content of texts as its goal in itself, the former aims to automate the extraction and formalization of knowledge, where the reason for the interest in text processing is that most knowledge about the world is captured in text [52, 42].

Ontologies

The task of discovery and labelling of relations, both taxonomic and non-taxonomic, has naturally been given much attention in the area of *ontology learning*, a research field that has been on the rise as a result of the availability of large text corpora through the World Wide Web and sufficient computational power to process large portions of it.

An *ontology* is an explicit, and often formal and machine-readable, conceptualization of the world or a specific domain [14]. *Concepts*, i. e. classes of entities, and their *taxonomic relationships*, i. e. their hierarchical *is-a* relations (also called *subclass* or *hyponym relations*), form the core of an ontology. To improve expressiveness, ontology taxonomies are sometimes extended to include a set of other general, pre-defined and pre-named relations such as *has-a* (also referred to as *part-of*, *meronym*, or *compositional relations*).

In Networks of Names, consider two concepts: `Person` and `Organization`. Names of people and organizations that appear in text refer to real-world entities that are *instances* of their respective classes, i. e. are related by an *is-a* relationship.

Non-taxonomic Relationships

We are interested in *non-taxonomic relation extraction* and *labelling* of relations between instances, i. e. the discovery and naming of (arbitrary) semantic relationships between specific people and organizations. Among problems related to ontology learning, non-taxonomic relation extraction is considered the most difficult [34].

Non-taxonomic relation extraction is complementary to *taxonomic relation extraction*. In contrast to taxonomic relations, the discovery of non-taxonomic relations is an *open information extraction* problem [4], in that the amount, types, and names of relations are not predefined. The difficulty of this task stems from some facts about relation extraction and labelling of non-taxonomic relations [34]:

1. Various relations and respective labels between two entities are possible. Thus, a “correct” solution is hard to determine (or does not exist), to the extent that human engineers or analysts can not always easily make decisions and agree with other humans on the exact nature and name of relations.

   A common example is the relation between companies and products. Although a straight-forward relation is for the company to *produce* a product, other relations are easily conceivable, e. g. to *sell*, *advertise* or *consume* a product. Even if the correct relation is identified, it remains unclear in the presence of synonyms what label offers the best or preferred description.

2. There is some consensus in the literature that relations between entities are typically conveyed by verbs. As a result, research usually focuses on the search of suitable verbs or verb phrases that appear in the context of entities. Relations between two entities are then described by *concept-concept-verb triples*.

   However, in general, relations in natural language text are not limited to verbs. They can also be implied or conveyed by grammatical structures other than verbs and verb phrases. For instance, the sentence “Bill and Hillary Clinton travelled to Martha’s Vineyard frequently during his presidential getaways.” implies that Bill and Hillary Clinton are a married couple, which has little relation to the verb “travelled”. Similarly, noun phrases such as in “Former US secretary of state Madeleine Albright joins Twitter.” convey relationships between organizations and people without relying on verbs: While “joins” is in fact a verb that relates Albright to Twitter, the relationship between the United States and Madeleine Albright is that she is a former secretary of state, which is not captured by a verb.

3. The evaluation of systems for automatic discovery and labelling of such relations proves difficult.

   Tasks related to processing of natural language usually relies on the comparison to a *gold standard*. In the case of non-taxonomic relation extraction, however, gold standards are problematic. Apart from no gold standards being widely available, they are difficult to create in general for the reasons discussed above. The ambiguity and range of possible relations and labels limits the expressiveness of a comparison to a static gold standard. Should an automatic system find “surprising”, but correct relations or labels that are missing in the gold standard, they would not be recognized as correct. In addition, the form of the gold standard pre-determines the form of possible relations. For example, if the gold standard expects verb-relations, other forms of relations cannot be evaluated.

   As a result, evaluation is often done with human interaction. This approach, however, poses other problems: First, evaluation by human interaction is susceptible to some bias, since the human can heavily influence the results. Second, depending on the domain of texts, some expertise of that domain is required to understand and appropriately judge results.

   Evaluation without and with user interaction is differentiated and coined *prior* and *posterior* evaluation, respectively. The former is based on precision and recall of the system with respect to the reference ontology, while the latter takes into account what classification results a user has accepted or rejected.

As a result of the complexity and challenges of the field, research has taken several paths addressing specific variations of the problem [51]. Some work focuses on the extraction of a fixed set of non-taxonomic, but pre-defined and pre-named relationships or limits itself to finding domain-specific relations between fixed entities, mostly in the field of biology and medicine. Other work does systematically seek unknown non-taxonomic relations, but relies on a posterior manual labelling by the ontology designer. An alternative approach aims to extract instances of relationships, but assumes specific types of relationships, e. g. that companies always *produce* products. Lastly, there is research devoted to finding labels for arbitrary relations. However, in this case, authors tend to make the assumption that relations are to be labelled by verbs or verb phrases, possibly including the limitation to specific grammatical forms.

An overview of methods and challenges in the discovery of non-taxonomic relations can be found in [34], in more detail in [14], and more recently [42]. Aussenac-Gilles and Jacques demonstrate practical problems by conducting a number of experiments [2].

Tagging and Folksonomies

Tagging can be defined as "linking terms to resources" and the user's role in this activity, while a *folksonomy* is the "collective assemblage of tags assigned by many users" [60]. In particular, tagging can thus be understood as the process by which users select the terminology used within a system.

Folksonomies carry certain advantages and disadvantages: While they suffer from synonymy, misspellings and general inaccuracy, they provide a bottom-up way to reflect user's needs in vocabulary and are very flexible.

Research on tagging and folksonomies is still ongoing and is often focused around the evaluation of tagging communities such as Flickr and del.icio.us, including user behaviour [26] the idea to formalize folksonomies into ontologies [28].

Tagging in Networks of Names slightly differs from the typical focus of existing research on documents, such as websites or photos. While within the scope of this work, we do not gather enough data to compare to longtime and public tagging communities, we do find our approach to exhibit both, the advantages and disadvantages inherent in tagging and folksonomies. Our user experiment also shows that the kind of user interface may influence vocabulary employed by users (see Section 7.2).

6.2 Generating and Applying Patterns

We aim for the classifier in Networks of Names to be trained and applied during operation of the system. Since no training set is present and the types of relationships are not predefined, this needs to happen following user input (used as training data) and with little supervision (since there is no or sparse positive and negative feedback on the classifier's decisions).

By the classification of relation extraction problems from [52], the problem is similar to the case "where we are given one or more relationship types, and our goal is to find all occurrences of those relationships in a corpus". Two approaches to this problem are distinguished for querying the corpus for eligible relationships: *pattern-based*, where sentence patterns are used to describe a relationship, and *keyword-based*, where documents are judged by the presence of certain keywords to describe a relationship between entities contained therein. The latter appears to be better suited for documents larger than sentences and possibly a fixed set of specific relationships. Thus, we select to employ a pattern-based approach. In [52], the problem is discussed with a focus on mining the Web as a corpus. In our case, however, the corpus has already been preprocessed, in that entity instances have already been found and assigned types.

6.2.1 Pattern Generation

Finding lexico-syntactic patterns is a problem that has originally been performed manually, typically for hyponym relations [31, 56], but also semi-automatically for other kinds of relations, such as causation [23] or meronymy [24]. This includes the fabrication of common and generalized grammatical constructions like

<NP$_1$> such as <NP$_2$> and <NP$_3$>

and other similar phrases, where NP stands for “noun phrase” and the sentence pattern implies a hyponym relationship between <NP$_1$> and <NP$_2$>, and between <NP$_1$> and <NP$_3$>, or, from a different perspective, a (non-hyponym) relationship between <NP$_2$> and <NP$_3$> of sharing a common umbrella term or belonging to the same class of entities <NP$_1$>. Examples of instances of this phrase are “countries such as America and Canada” and “famous scientists such as Charles Darwin and Albert Einstein”.

The discovery of such patterns for a given relationship can be done by simple observation and human experience, or by systematically crawling corpora for the co-occurrence of several pairs of entities that are known to share the relationship, identifying commonalities of the result sets, and hypothesizing about what patterns would yield good results on unseen text.

Since manual pattern engineering is tedious, possibly biased, and limited in feasible scope, automatic methods have also be suggested for pattern discovery [52, 61].

From the user interaction in Networks of Names, we not only get an entity pair and a relationship label, but also the accompanying source sentence. Exploiting that, we opt for a simple high-precision low-recall attempt to extract a pattern directly from that sentence.

In our approach, the classifier extracts a phrase from the sentence that contains both entities and possibly a relevant keyword used by the user in his label, generalizes both, the phrase and the label, and saves the result for subsequent application.

For instance, given the sentence

> *Der Leiter der Augsburger Staatsanwaltschaft, Reinhard Nemetz, ist aber optimistisch, dass Schreiber ausgeliefert wird.*
>
> *(But the head of the Augsburg prosecution, Reinhard Nemetz, is optimistic that Schreiber will be extradited.)*

and the user-entered label “ist Leiter der” (“is head of”) for the relationship between *Reinhard Nemetz* and *Augsburger Staatsanwaltschaft*, the system will engage in the following steps to derive a pattern:

1. Tokenize the label and remove stopwords to decide whether a keyword is present. In case there is exactly one, this keyword will be used to generalize the label.

   In the example, the only non-stopword is “Leiter”. Hence, it is a keyword in the aforementioned sense.

2. Extract a subsentence that contains both entity names and the keyword (if present).

   In the example sentence, this corresponds to the phrase “Leiter der Augsburger Staatsanwaltschaft, Reinhard Nemetz”.

3. Generalize the extracted phrase and user-entered label. The phrase is generalized by substituting entity names by placeholders that denote their type and the keyword (if present) by a placeholder for any word. In analogy, the label is generalized by replacing the keyword by a placeholder.

   In our example, the resulting generalized pattern is:

   <WORD> der <ORGANIZATION>, <PERSON>

   The generalized label is:

   ist <WORD> der

The resulting pattern and generalized label are saved into the database and the classifier proceeds by applying the newly learned pattern to the corpus.

In the example, we abstracted from the implementation detail that the direction of relationships is also encoded in the pattern, by marking entity placeholders with a code for subject and object (for a direction from subject to object), or both as subjects (for a undirected relationship), in addition to their type. Thus, assuming the tag "ist Leider der" is directed from *Reinhard Nemetz* to *Augsburger Staatsanwaltschaft*, the derived pattern would be

<WORD> der <ORGANIZATION/O>, <PERSON/S>

to denote the direction from the person to the organization.

6.2.2 Pattern Application

The presence of a semantic relationship in Networks of Names is expressed by a tag. Tags are tied to a sentence, which acts as the source for the semantic relationship, and a relationship between two entities, because sentences might contain mentions of more than two entities (and we consider only binary relations). Consequently, pattern application has to result in tags for sentence-relationship-pairs.

This is implemented in two steps. First, the database is queried for sentences that may conform to the pattern, with wildcards replaced by SQL-wildcards. After sentence and relationship candidates are retrieved, a check for matches is performed by substituting entity names from the relationship into the pattern (if their types match) and applying the pattern to the sentence. This is done using regular expression syntax and the respective programming language features.

For every match, if the label associated with the pattern is generalized, i. e. contains a wildcard, the matching string from the sentence is substituted for the wildcard.

From all resulting matches, tags are created and written into the database. Every automatically generated tag is linked to the pattern that is was created from (so that tag validations can be related back to the affected patterns). If the pattern generates a tag that was already user-created, the tag is instantly accepted to be correct.

6.2.3 Pattern Evaluation

Users can accept and reject automatically generated tags in the course of their interaction with the system (as described in Section 5.2). This feedback can be used to evaluate patterns by calculating a precision.

We assume applications to be correct if some user has accepted the corresponding tag (or, more specifically, the label of the tag) as correct and it was not rejected. We assume applications to be wrong if the corresponding tag was removed, and thus rejected, by a user (regardless of whether it was also accepted).

Using the number of accepted and rejected applications, it is possible to calculate a precision of some pattern i by [39]:

$$p(i) = \frac{\text{number of accepted applications}}{\text{number of accepted applications} + \text{number of rejected applications}}$$

The resulting value denotes on a $[0, 1]$ scale the fraction of correct applications. For the interpretation of the metric's meaning, two factors should be considered:

- Since the calculation is based only on the number of validated (accepted or rejected) applications, it is important that the sample is large (and representative) enough to be expressive.
- As we do not have a gold standard for reference, this precision is a result of posterior evaluation and thus susceptible to user bias. Users are potentially biased in two ways: They influence what

parts of the graph they explore, what tags they validate, and they have the freedom to decide what is “correct” and “wrong” when they validate tags. In addition to user bias, the system extrapolates validations to other tags for the same relationships that bear the same label (as noted in Section 5.2).

7 Evaluation

We evaluate several aspects of Networks of Names. First, we argue that the system is suitable for the exploration of relationships between people and organizations as depicted by newspapers in Section 7.1 by means of a case study. We conduct an exploratory user experiment and present our findings in Section 7.2. Last, in Section 7.3, we evaluate the performance of our classifier based upon data from the user experiment.

7.1 Case Study

To show how Networks of Names can be used to explore and present relations of people and organizations in the context of some event under investigation, we give an example walkthrough of the interaction involved in such a task.

As a scenario, we select the *CDU donations scandal*[34] of 1999. Since we need to conduct the search for a specific name at the beginning, we assume that it is already known to us that Walther Leisler Kiep, then treasurer of the CDU[35], was a central figure in the events.

We start our exploration without any tags (manual or automatic) available in the system in order to demonstrate that the classifier works and provides a useful addition to the system.

1. We bring up the *New* dialogue, enter "Walther Leisler Kiep" into the search form for single names and submit the query. The result of the initial query can be seen in Figure 16. The nodes *Weyrauch* and *Casimir Prinz* are duplicates of other nodes already seen on screen (with only parts of their names). We remove them.

2. We view the relationships of the only organization in the graph, *Augsburger Staatsanwaltschaft* (Augsburg prosecution) and find that an arrest warrant has unexpectedly been issued against Kiep, the prosecution is hearing *Horst Weyrauch* in the case, and *Reinhard Nemetz* is the head of prosecution. We create the respective tags: "erlässt Haftbefehl" (issues arrest warrant), "vernimmt" (hears), and "Chef der" (head of). After reordering the nodes, the graphs looks as shown in Figure 17.

3. From viewing the relationships of the prosecution, we also learn that Weyrauch, a tax counsellor of the CDU, is a close friend of Kiep. We view the relationship between Kiep and Weyrauch. One sentence states that the investigation uncovers an institution that Kiep and Weyrauch operated to launder money (although without further context it remains unclear what institution that is). We tag the sentence with "betreiben Geldwäsche" (launder money).

4. We investigate the remaining connections of Kiep and Weyrauch. We find that *Uwe Lüthje* is the chief representative of Kiep and together with Weyrauch was responsible for managing illegal funds, *Hans Terlinden* is an employee of former chancellor Helmut Kohl and an important witness alongside Weyrauch, *Casimir Prinz Wittgenstein*, the state treasurer in Hesse, is also investigated in a related matter. Finally, we learn that a court has accepted charges against Kiep and *Winfried Haastert*, a manager of Thyssen. The graph at this state is shown in Figure 18.

5. We expand *Thyssen* as a neighbour of Haastert, and request additional neighbours of Kiep. We notice that one of the newly expanded nodes is extremely well connected with the existing ones, including the prosecution, Kiep and Thyssen: *Karlheinz Schreiber*. We remove the other newly expanded nodes for clarity. Investigating relationships, we learn that Schreiber acted as an intermediary for Thyssen, an arms manufacturer involved in tank exports to Saudi Arabia at that time, and that Schreiber donated one million Deutsche Mark from a Thyssen account directly to Kiep.

[34] `http://en.wikipedia.org/wiki/CDU_donations_scandal_(1999)`

[35] The *Christian Democratic Union of Germany*, a major catch-all centre-right party in Germany.

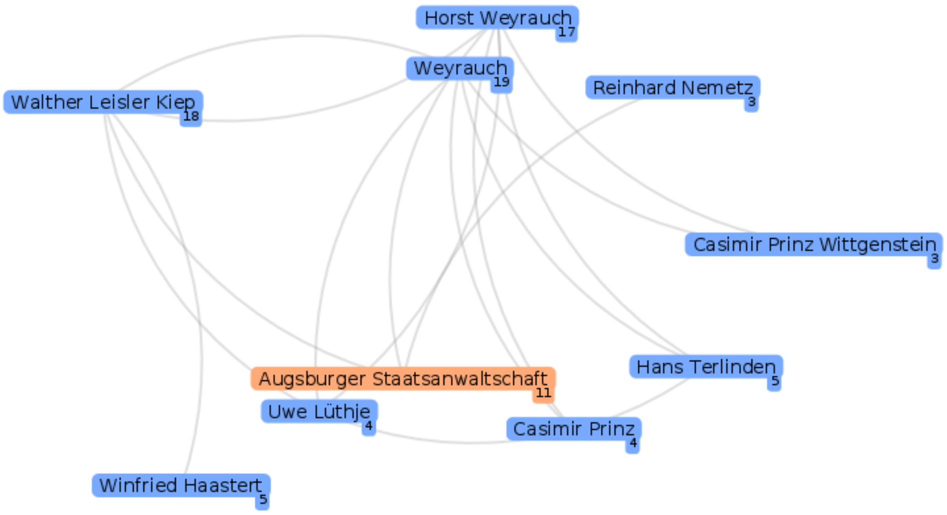


Figure 16: Case Study: CDU donations scandal (1999). The initial search result for *Walther Leisler Kiep*.

We tag the respective relationships with "Mittler für" (intermediary of) and "übergibt Million an" (transfers a million to). Figure 19 shows the visualization after inclusion of the two new nodes.

6. At this state, the visualization gives a good overview over the state of affairs during the scandal. It could be expanded further for the remaining personalities in the graph, possibly leading the exploration towards details on the system of illegal bank accounts that was uncovered in the process of the criminal investigation. For this case study, we stop at this point, and perform one last expansion: Bringing up the CDU as a neighbour of Kiep shows that several links from the party to its officials have already been automatically labelled. We remove several edges that seem unimportant in order to reduce visual clutter. Kiep and Weyrauch have been correctly labelled as "Schatzmeister von" (treasurer of) and "Finanzberater von" (financial advisor of) the party, respectively. The label selection for the edges also contains other valid alternatives. Uwe Lüthje has mistakenly been labelled as "Bundesschatzmeisterei der" (federal treasury of), but is, as a closer look reveals, instead actually "Generalbevollmächtigter der Bundesschatzmeisterei der" (chief representative of the federal treasury of). We delete the wrong label. The final state of the visualization can be seen in Figure 20.

The result demonstrates that Networks of Names can be used to purposefully investigate this kind of affairs. In the course of the user study, users investigated and created visualizations of different types of constellations, after receiving an introduction on how to operate the tool, but without detailed instructions on how to conduct explorations. Examples of the results of user explorations and improvement suggestions given by users are presented in Section 7.2.2. Useful future additions to the system are discussed in Section 8.

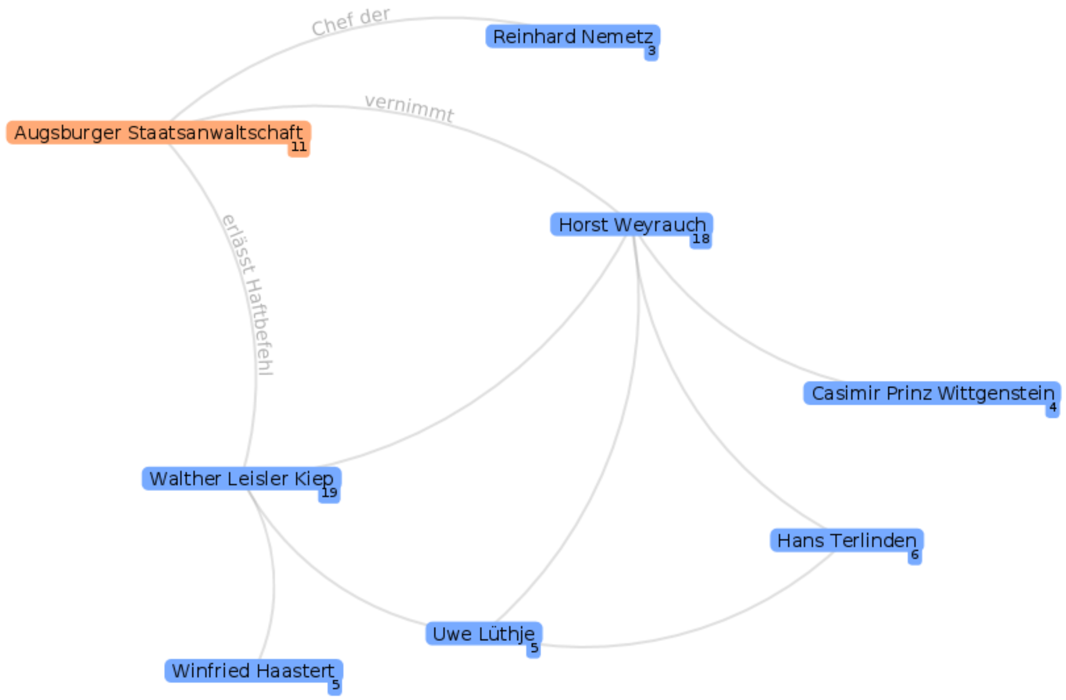


Figure 17: Case Study: CDU donations scandal (1999). The initial graph with artefacts removed, nodes reordered, and relationships of the prosecution labelled.

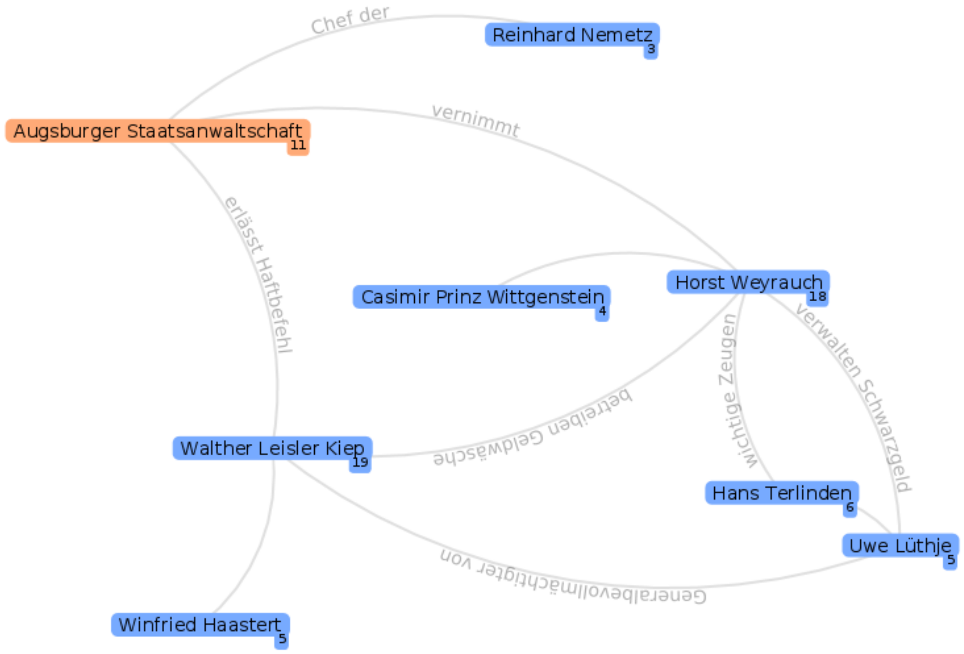


Figure 18: Case Study: CDU donations scandal (1999). The initial graph with all relationships inspected and labelled.

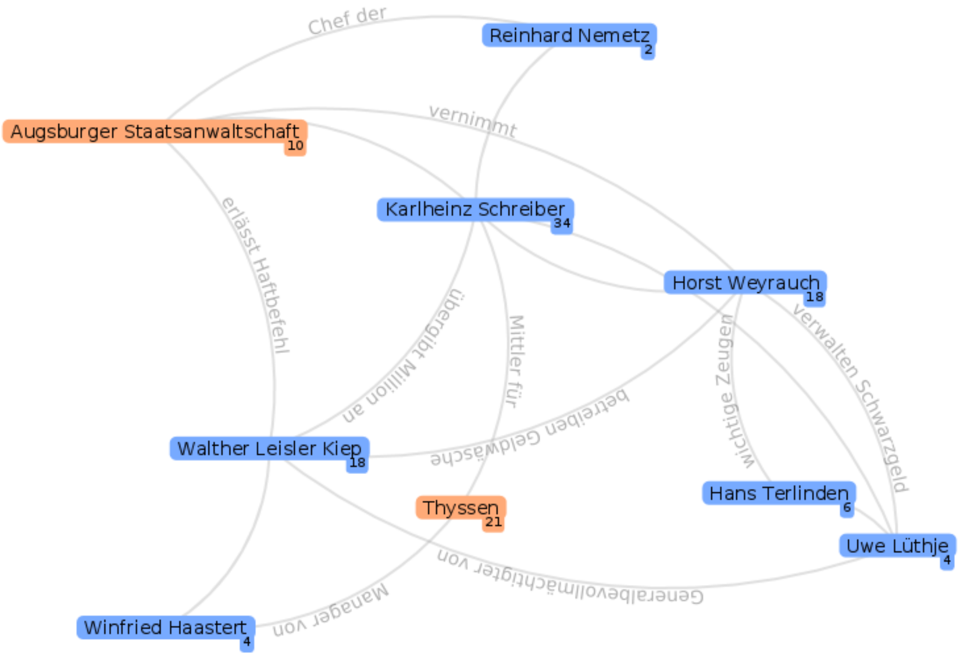


Figure 19: Case Study: CDU donations scandal (1999). Expanding *Thyssen* and *Karlheinz Schreiber*.

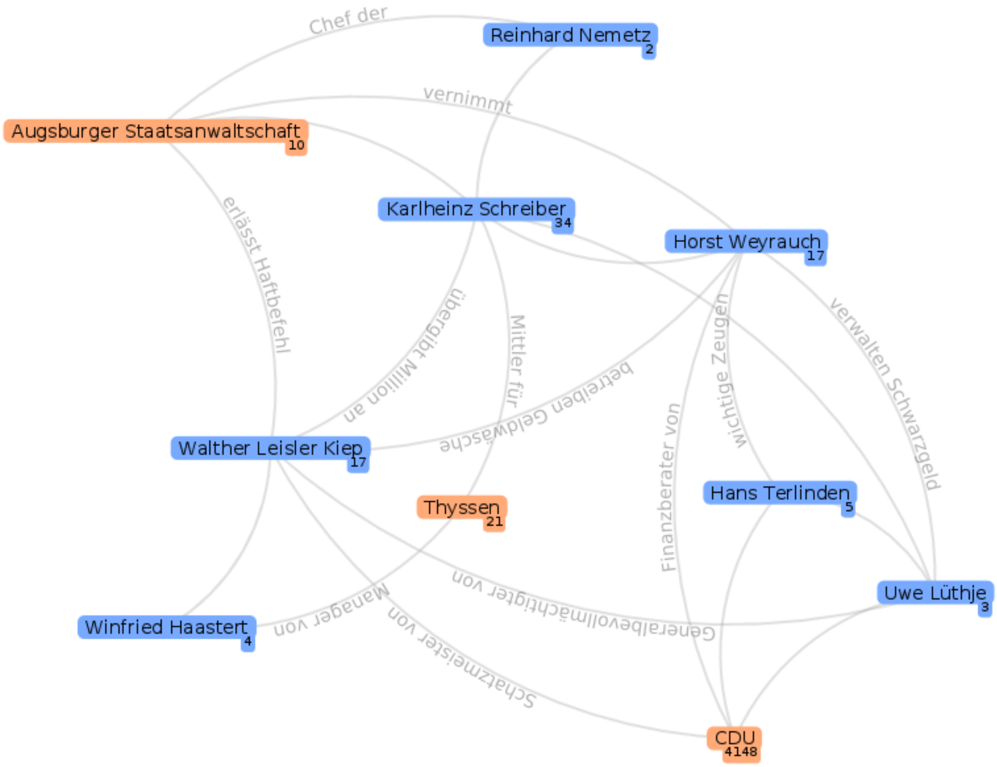


Figure 20: Case Study: CDU donations scandal (1999). Final state after expanding *CDU*.

Stage	Visual Study	Text Study	Degrees of Freedom
1	Instructions	Instructions	–
2	Conduct Search	–	Search terms, number of searches
3	Explore Graph	–	Time spent, direction of expansion, relationships viewed
4	Tag Relationships	Tag Relationships	Sentences tagged, number of tags, wording

Table 3: User study stages and users' degrees of freedom

7.2 User Experiment

In order to evaluate the possibilities and weaknesses of our visual interactive system, we conducted an exploratory user experiment. Our aims for the experiment were threefold:

1. We wanted to test the system with users that have no special background knowledge on networks, network visualization, or language technology, and no knowledge about the internal working of the system. We wanted to see how users choose to interact with the system and collect feedback on how to develop it further in the future.

2. We wanted to obtain data needed to train, apply and evaluate our classifier. This data is created by user interaction, which would be one-sided and biased if we were to generate it ourselves.

3. We wanted to explore whether working with the visual interactive system impacts how users create tags, as opposed to a comparable text-based tool.

7.2.1 Setup

The user experiment was conducted in two parts: In the first part users were asked to use the visual interactive system (the visual study), while in the second part users were confronted with a text-based system that allows tagging of relationships (the text study). The first part served the purpose of testing our system with users, while the second part was designed to explore the differences between tags created in the visual study, compared to tags that users of a text-based system would create.

Thirteen people participated in the visual and text studies, respectively. The two groups did not overlap. All participants were between the ages of 23 and 34. For the most part, participants did not have special knowledge about language technology, networks, or network visualization. Both groups had few participants with basic knowledge of computer science and the aforementioned fields.

The core stages of each study and users' degrees of freedom are summarized in Table 3.

The Visual Study

First, participants of the visual study were given an introduction into the user interface of the visual interactive system. Specifically:

1. How to conduct searches for single names or relationships between two named entities, how to limit the timeframe of the search, and how to execute the predefined example searches.

2. What visual elements represent: blue nodes for people and orange nodes for organizations, edges between them for potential relationships, edge labels for the name of a semantic relationship.

3. How to interact with visual elements: panning and zooming, dragging nodes, summoning the sources view by clicking on an edge or an edge label, calling a context menu on nodes and edges and what the options in context menus mean.

4. For the sources view, how sources are ordered and clustered.

5. In the sources view, what the visual elements in a tag mean (direction, label and checkmark for validated tags), how to distinguish between manual (blue) and automatic (grey) tags, how to use the tagging widget to create own tags, and how to accept or reject automatic tags.

6. How to navigate the graph by automatic expansion, explicit expansion, and the removal of nodes.

7. That incomplete sentences, name misspellings, and partial names exist as artefacts from automatic processing and can be ignored.

Users were then asked explore one of the example searches and subsequently conduct at least one search of their own. In order to allow users to get a feeling on what kind of results the system can produce, they were encouraged to execute several searches, before committing to the exploration of a scenario.

Apart from being asked to conduct at least one example and one custom search, users were given several degrees of freedom. For searches, users could choose how many searches they conduct and what names they search for. In the following graph exploration, they could navigate the graph to their liking and select on their own what they explore and present using the software. Likewise, it was in the users' control what relationships they view. In order to create edge labels, users needed to view the sources and create tags. It was left to them to decide what relationships they deem relevant or interesting, what sentences they tag, how many tags they produce, and how they word tag labels. Users could decide how much time exactly they spend using the system.

Lastly, users were instructed to accept or reject automatic labels, should they encounter any.

The Text Study

In the text study, users interacted with a text-based system that corresponds to just the sources view of the visual interactive system. Differently from the visual study, there was no graph visualization and users did not conduct their own searches, did not explore a graph, and did not select relationships. Instead, they were successively presented the sources view (identical to that from the visual interactive system) for a number of relationships. The dataset for every user directly corresponded to relationships viewed by a participant of the visual study (with duplicates removed), including the order of relationships. The order of datasets also corresponded to the visual study, i. e. the first participant of the text study was assigned the dataset of the first participant of the visual study, and so on.

Like in the visual study, participants of the text study were given an introduction on the core elements of the user interface and the possibilities to interact. The instructions were the same as instructions 4., 5., and 7. given for participants of the visual study, i. e. only those relevant to the sources view.

Lacking the context of graph exploration and visualization, users were asked to characterize relationships displayed to them as a whole by selecting and tagging relevant sentences. Contained to the sources view, their degrees of freedoms were similar. Users were free to decide whether they create new tags or decide that existing tags – created by other users or automatically by the system – are sufficient. Analogously to participants of the visual study, users were free to decide what aspects of a relationship they deem relevant or interesting, what sentences they tag, how many tags they produce, and how they word tag labels.

Participants were not required to meet any time constraints. However, since they had no control over which and how many relationships they view, the time needed to complete the study was heavily influenced by the size and complexity of the dataset assigned to them.

Like participants of the visual study, users were instructed to accept or reject automatic labels, should they encounter any.

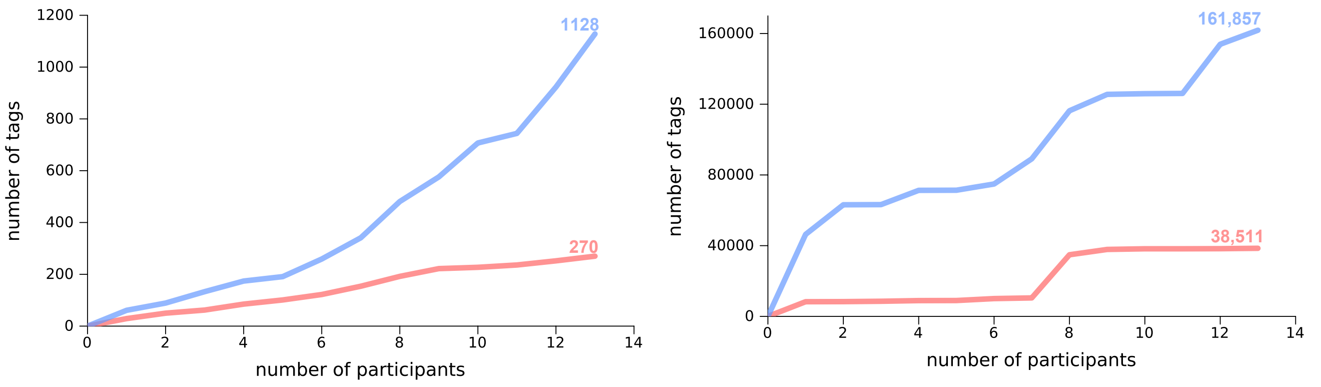


Figure 21: Cumulative number of manual tags (left) and automatic tags (right) created in the course of the user study after each participant. Results from the visual study are plotted in red, results from the text study in blue.

7.2.2 Results and Discussion

By conducting the user experiment, we obtain data that allows us to evaluate how the system performs. Specifically, we discuss how the number of tags evolves through the course of the study, how the interaction with the system compares between groups, what influence visualization has on tagging behaviour, what users remark most frequently about the visual interactive system, and what kind of scenarios they choose to investigate.

Number of Tags

While users interact with the system, they create tags that describe the semantics of a relationship. This triggers classifier learning and subsequent application. We record the number of manual and automatic tags after each participant to see how the numbers evolve.

Figure 21 plots the numbers for both groups. Two aspects can be observed from the data:

1. The amount of automatically generated tags greatly outnumbers manual tags. While there are more tags created overall in the text study, in both cases, the number of auto-tags is larger than the number of user-tags by a nearly identical factor of roughly 143.

2. The plot reveals that the number of automatically generated tags does not grow smoothly, but is instead subject to occasional steep jumps.

The second observation can be explained by the classifier learning patterns that have a high number of applications. This is the case when the user tags a sentence that happens to be well-suited for pattern generation and the resulting pattern is common enough in the corpus to facilitate frequent application.

We assume that users are unlikely to select sentences for their suitability in pattern generation, because it is not important for their task of network exploration (or manual relationship classification) and they have no detailed knowledge about how the classifier works. We assume that instead users select sentences for their content (or arbitrarily). However, as noted in Section 5.3.3, when displaying representants of clusters, we select one sentence by its superficial suitability for pattern generation (namely, when the entities appear close to each other with one, two, or three words between them). The results show that users “hit” good patterns frequently enough to allow the classifier to regularly learn patterns that generate many tags.

Quantitative Differences between Groups

Table 4 shows per-user metrics for both studies. We measure the time users spend using the software, the number of tags they create, and the number of automatic tags they accept and reject.

	Visualization	Text-Only
Average time [min]	55	85
Average number of tags	20.77	86.77
Average number accepted tags	10.15	59.30
Average number rejected tags	4.92	15.23

Table 4: Metrics on user interaction per user for the visual versus the text study.

Users of the visual study were able to choose how long exactly they spend working with the software. From the average time of 55 minutes (which is longer than we anticipated) we conclude that interaction with Networks of Names is capable of keeping users engaged beyond the strict requirements of the study. Users in the text study spent 85 minutes on average, which is considerably longer than in the visual study. Given the fact that the datasets were identical to the visual study and the activities of text study participants were a subset of activities in the user study, this result is surprising. In part, this could be attributed to a bias in the instructions. Apart from that, we assume that it is possible that using a text-based system users were compelled to read more sentences per relationship, and read them more thoroughly, significantly raising the time required to process all relationships.

Looking at the number of tags created by users on average, we see that in the text study, users did create more than four times as many tags than participants of the visual study. In relation to the average time, this also means that the number of tags per minute was also higher in the text study. We assume that this difference can largely be attributed to the following aspects of the circumstances:

- In the text study, users were involved with the sources view only, thus spending their time exclusively in the part of the system where tags can be created.

- Users in the text study, as opposed to the visual study, had the explicit analytical task of characterizing relationships by tagging, without the overarching goals to visually explore a social graph. As the creation of tags is more central to the text study, it seems natural that more tags were created.

- In the visualization, it seems to be a common behaviour to open a relationship, decide that it is not interesting enough or does not properly relate to the rest of the graph, and delete the relationship from the graph without tagging any sentences. Given the same relationship, users in the text study are likely to tag it, however, since they were explicitly asked to do so and for the lack of the respective context from the visualization.

The numbers for automatic tags accepted and rejected by users[36] show that users did encounter sentences that were pre-tagged by the classifier and that more tags were correct than false. At the same time, the overall number of validated tags is considerably less than 1%, with several patterns having no feedback at all. There are several possible explanations for the low number of validations:

- While 13 participants in each group are sufficient to derive insightful information on how users interact with the system and what data is produced, this number is not high enough to validate comparatively large amounts of automatic tags.

- At the beginning of each study, the system does not contain any automatic tags, since the classifier is untrained. Thus, automatic tags emerge successively. Consequently, earlier participants have less chances to encounter automatic tags.

[36] The numbers include validations extrapolated by the system to other tags in the same relationship that bear the same label as the accepted/rejected tag, as discussed in Section 5.2.

	Visualization	Text-Only
Number of manual labels	171	877
Average number of words	1.53	2.67

Table 5: Metrics on tag labels created by users in the visual versus the text study.

- Users conduct searches from different domains, but they explore only small parts of the graph overall. However, tags created by the classifier can appear in all parts of the graph. It is thus natural that some relationships, especially less interesting ones, are rarely found by users.
- Users in the visual study tend to overlook or forget to validate automatic tags, because they are focused on other aspects of their interaction with the system, such as the exploration of the graph. For similar reasons, users may not read all sentences for a relationship (and thereby miss tagged sentences), if they feel that the sentences they have already read (and possibly tagged), are sufficient for their understanding.
- If, using the visual interactive system, users find edges that are already labelled, they might be satisfied with the existing label and not investigate the relationship more closely by bringing up the sources view. In this case, a label would be perceived as correct, but no feedback to the system would be given.

We conclude that more user interaction is needed to obtain more expressive results for the evaluation of the classifier. We do evaluate the precision of our classifier in a more controlled environment and give the results in Section 7.3.

Differences in Tag Labels

For a qualitative comparison of differences in tags between the visual and text study, we compare the two respective sets of labels entered by users. The exact wording of labels was a degree of freedom in both groups. Table 5 shows the number of labels and the average number of words per label in the visual and text study, respectively. With five times as many unique labels in the text study, the difference between the two groups with respect to the number of labels is even larger than the difference with respect to the number of tags.

By direct comparison, the two sets of labels share only 30 labels. To get a better quantitative idea of the similarity of the two sets, we relax equality criteria to account for similar labels, and consider labels similar if they are phrases of each other, are the respective masculine/feminine/neuter or plural forms, are modified by temporal markers, or have suffixes that signal membership or presidency. Thus, we allow a label “Vorstand” (executive) to be matched by variants like “ist Vorstand” (is executive), “ist Vorstand bei” (is executive at), “ist ein langjähriges Vorstandsmitglied von” (is a longtime executive board member of), and similar such variations with different phrasing, articles, prepositions, and combinations thereof. Results of this similarity analysis, along with a list of labels that are unique to the respective parts of the study, can be found in Appendix A. Using this notion of similarity, 73 and 112 labels have counterparts it the other set, respectively. This corresponds to 42.69% of labels from the visual study and 12.77% of labels from the text study. The larger amount of labels unique to the text study can partly be attributed to the larger amount of tags in general and to tags created for relationships that were viewed, but not labelled in the visual study (and thus potentially belonging to different domains).

However, this discrepancy leads us to investigate labels of the text study in detail. We find several peculiarities. One aspect is that some labels are considerable longer than labels from the visual study.

ist gegenwärtiges Aufsichtsratsmitglied genau wie

(is current member of the supervisory board just like)

muß sich nicht wegen Annahme einer Millionenspende verantworten

(is not held accountable for accepting a donation of one million)

Labels like these (entered by different users) convey a reasonable relationship, but are very verbose compared to the wording usually selected by users using the visual system.

Several other labels describe very high-level similarities.

Ikonen

(icons)

Frauen

(women)

für Ideale gekämpft

(fought for ideals)

Such labels (also entered by different users) may, but do not necessarily describe meaningful semantic relationships.

Another category of labels is extremely specific in both, wording and content.

Autoren von [T]exten über das kompl[i]zierte Ökosystem Wald

(authors of text about the complicated ecosystem called forest)

sind die schlimmsten Bluthunde

(are the worst bloodhounds)

These labels (entered by different users as well) use a very specific wording. Both can be found to be rephrases of direct quotes from the respective source sentences.

The examples show that there is a reason to assume that the sets of labels are in fact significantly different from each other. We assume the visualization to have an impact on this observation for two reasons:

1. Extremely long or comparatively inexpressive relationship labels such as the examples above are not false, but are difficult to imagine as a label on an edge and in the context of a larger graph visualization.

2. Although users of the text study were explicitly asked to characterize relationships as a whole (and to proceed if they decide that a relationship is sufficiently characterized), the examples suggest that they may have gotten involved with the details of source sentences more than users of the visual interactive system. This could have lead to the under- and over-specific labels mentioned above, which directly correspond to newspaper contents that would not have been chosen for tagging by visual study participants due to their under- or over-specificity.

It is thus possible that the presence of a visualization has a regulating effect on the emerging folksonomy.

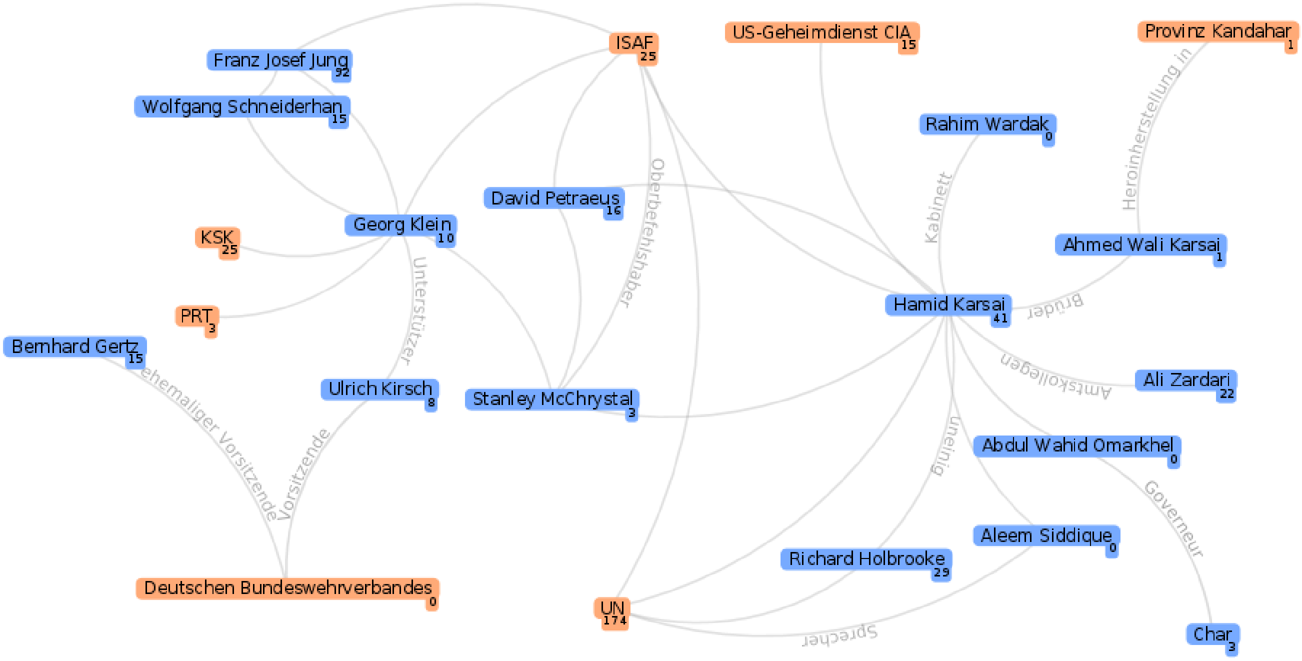


Figure 22: User-created narrative structure: Georg Klein and Hamid Karsai.

User Feedback

After the study, users were asked to anonymously fill out a survey form. In addition to their age and information on their background knowledge about language technology, networks, and network visualization for completeness, we asked participants of the visual study whether Networks of Names could be used as an alternative to traditional newspaper research (i. e. the exploration of newspaper contents by reading newspaper articles) and gave the opportunity to express suggestions for improvement.

11 of 13 people answered the question: 5 answered “yes” without further comment, 3 answered “yes” if additional analytic features were implemented, and 3 found that it could be used complementary to other methods of research.

Users stated that in certain domains background knowledge is needed to understand the situation (we assume this refers to scenarios such as political and financial scandals). Users that voiced concrete improvement suggestions stated that, while it is possible to derive relevant information from single sentences, in many cases too much context is missing, making the information seem incomplete or be incomprehensible altogether. Another suggestion was the inclusion of a possibility to view details not only on edges, but also on nodes (containing information such as the lead paragraph from Wikipedia).

Examples of Narrative Structures Created by Users

Section 7.1 details a possible workflow using Networks of Names by walking through an example scenario. In the course of the user study, users visualized other scenarios. Although they were instructed on what possibilities of interaction are available to them, we gave no explanations on our ideas of the intended workflow. Furthermore, all study participants were first-time users of the system. Thus, user results are interesting, because they give additional examples of narrative structures obtained from Networks of Names, and because they are unaffected by our own usage bias.

Figure 22 shows a visualization in which the initial search was an example search for the *Kunduz airstike*[37] of 2009. Starting from this original search, the user investigated the scenario, removed unrelated nodes and tagged a few selected relationships of the people involved, such as that *Ulrich Kirsch*, the head (“Vorsitzender”) of the *Deutscher Bundeswehrverband* (German Armed Forces Association), was

[37] `http://en.wikipedia.org/wiki/Kunduz_airstrike`

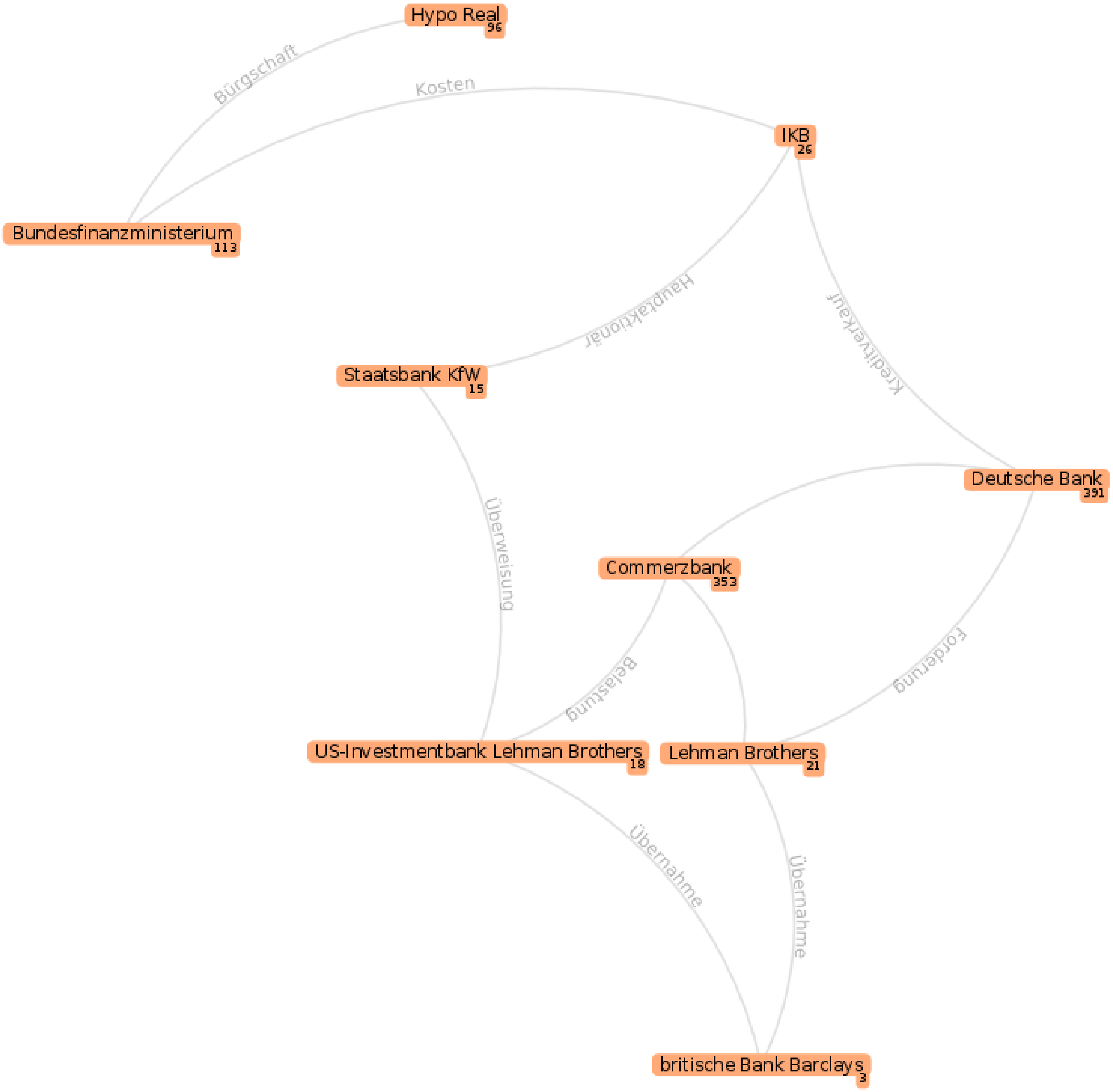


Figure 23: User-created narrative structure: The 2008 financial crisis.

a supporter (“Unterstützer”) of Georg Klein, the commanding officer responsible for the attack. The user then apparently changed focus by driving the exploration towards *Hamid Karsei*, the president of Afghanistan, and proceeded expanding and tagging his relationships.

In Figure 23, a user created a visualization of Germany-centred relationships with regard to the *2008 financial crisis*[38], although the graph contains several artefacts from preprocessing (“Hypo Real” instead of “Hypo Real Estate”, the prefixes “Staatsbank” (federal bank) for “KfW” and “US-Investmentbank” (US investment bank) for “Lehman Brothers”, and “britische Bank” (British bank) for “Barclays”). The labels chosen here seem too short to convey enough information for a third party to understand the situation without additional knowledge or own investigation, but it can be roughly seen how the collapse of *Lehman Brothers* influenced financial difficulties of German banks, which eventually had to be bailed out by the government, represented here as “Bundesfinanzministerium” (Federal Ministry of Finance).

7.3 Classifier Performance

The results of the user study provided authentic classifier data. However, this data is too sparse to be used for the evaluation of classifier performance. In addition, feedback obtained from user interaction does not properly sample pattern applications, because it is biased towards certain regions of the graph (such as those provided as example searches), and is thus distributed neither evenly, nor randomly.

As a result, we evaluate the classifier manually and in a more controlled environment. Deviating from how the classifier actually encodes patterns, we omit information of tag direction in the examples given in this section for brevity.

7.3.1 Setup

From the study, two datasets are available to us as a possible basis for our evaluation. In the visual study, the classifier learned 225 patterns that were applied to 38,511 relationships. The text study yielded 850 patterns and 161,857 applications.

By reviewing some of the patterns manually, we see that the data from the text study is much more exhaustive. Thus, we select the dataset obtained from the text study. We discussed in Section 7.2.2 that tagging behaviour of users showed significant differences between groups. However, this does not directly affect classifier performance for two reasons: First, classifier learning and application is performed algorithmically, triggered by user input, but without further user interaction (or knowledge about how the process works in detail). Second, tagging of peculiar sentences will typically not yield meaningful patterns. Instead, such patterns will often apply only to the sentence they are derived from. To exclude such low-quality patterns, we filter our all patterns that were applied less than five times. This leaves 133 patterns for validation.

While in the visual interactive system users have the choice between accepting and rejecting labels (or not validating them at all), we introduce an additional judgement of being “close” to the correct solution. The reasoning for this relates to the way the classifier is implemented: Labels entered by users are generalized by replacing a keyword by a wildcard, while the rest of the label remains unchanged. As a result, some automatic tags may result from proper pattern application, but be oddly worded. For instance, it would be natural to label a relationship between Sigmar Gabriel and his political party, the SPD, conveyed by the phrase “SPD-Vorsitzender Sigmar Gabriel” (SPD-chairman Sigmar Gabriel), as “Vorsitzender der” (chairman of (f.)). This pattern is generalized to “<ORGANIZATION>-<WORD> <PERSON>” and its label to “<WORD> der”. However, in German, this phrasing of the label demands an organization that is grammatically feminine. Applying the pattern to the phrase “ZDF-Intendant Thomas Bellut” (ZDF-director Thomas Bellut) would generate the label “Intendant der” (director of (f.)), with the correct form being “Intendant des” (director of (n.)), because ZDF is grammatically neuter. Similar situations arise if users specify temporal markers such as “former” or “longtime” in the label, which do not apply to other instances. Another common problem stems from adjectival declension: Given the phrase “der Chef der

[38] http://en.wikipedia.org/wiki/Financial_crisis_of_2007-2008

SPD, Sigmar Gabriel" (the head of the SPD, Sigmar Gabriel) and the label "Chef der" (head of), the system would generate the pattern "<WORD> der <ORGANIZATION>, <PERSON>". However, other candidates for keyword substitution, such as "Vorsitzender" (chairman), appear in an analogous phrase as "der Vorsitzende der SPD, Sigmar Gabriel" (the chairman of the SPD, Sigmar Gabriel). As a result, the generalized label would be instantiated to "Vorsitzende der" (adjectival nominativ of "chairman of"), which is grammatically incorrect as a stand-alone expression without a leading definite article. Adjectival declension is the most frequent phenomenon, but other declension-related errors do also occur. The case study demonstrated another problem with substitution of keywords into labels: In the phrase "Generalbevollmächtigter der Bundesschatzmeisterei der CDU, Uwe Lüthje" (chief representative of the federal treasury of the CDU, Uwe Lüthje), the above pattern can be applied, but yields the keyword "Bundesschatzmeisterei" (federal treasury) instead of "Generalbevollmächtigter der Bundesschatzmeisterei" (chief representative of the federal treasury). To recognize such cases correctly, the system would need to detect the complete noun phrase, instead of a single word, and substitute it for the wildcard as a whole.

We randomly sample 1000 automatic tags created by the 133 patterns and annotate the tags manually into three categories: accepted, rejected, close, and undecided (the latter being a possibility to skip a tag, if we cannot decide if it is correct or not). From the annotations, we can derive a precision for the classifier. The value is a posterior precision, since we validate the data manually (and not against a reference ontology). We attempt to validate the data objectively, but note that as discussed in Section 6, posterior values can be susceptible to user bias.

7.3.2 Results and Discussion

For the performance of the classifier (as defined in Section 6.2.3), we arrive at 0.53 for accepted tags only, and 0.61 including tags that were "close". In the context of precision values given for other approaches employing lexico-syntactic patterns [56], this precision seems to be a reasonable result, given our rather simple approach of pattern extraction and application.

All patterns that we sampled, ordered by precision, are listed in Appendix B.

High-precision Patterns

High-precision patterns that emerge from the validation show close similarities to lexico-syntactic patterns usually stated and used in literature, such as in [31, 56, 4, 43]. For instance, the most frequent patterns are

<WORD> der <ORGANIZATION>, <PERSON>
(<WORD> of <ORGANIZATION>, <PERSON>)

<ORGANIZATION>-<WORD> <PERSON>

and other similar patterns that describe positions of people in organizations, but vary in word order and sentence construction. Keywords substituted for <WORD> include "Vorsitzender" (chairman), "Leiter" (head), Chef (head), "Experte" (expert), "Sprecher" (spokesman), and similar. Such patterns reach (especially "close") precisions of nearly 1, but suffer from frequent grammatical or phrase-related errors discussed above.

A group of patterns that are often given in research literature and appear with similarly high precision in our results, are patterns that describe hyponymy, such as:

<WORD> wie <PERSON> und <PERSON>
(<WORD> such as <PERSON> and <PERSON>)

<WORD>, darunter <PERSON> und <PERSON>
(<WORD>, including <PERSON> and <PERSON>)

<PERSON> und <PERSON>, den beiden <WORD>

(<PERSON> and <PERSON>, both (accusative) <WORD>)

Typical substitutions for <WORD> are “Stars” (stars), “Sänger” (singers), “Autoren” (authors), and other collective terms. But hyponymy is also correctly recognized by using more specific patterns that find enumerations with known representatives of some class of entities, such as for investment banks:

<ORGANIZATION>, Goldman Sachs und <ORGANIZATION>

(<ORGANIZATION>, Goldman Sachs and <ORGANIZATION>)

Since Networks of Names was created for the exploration of social networks, it is to be expected that some patterns for the relationships between people emerge. More general such patterns are:

<PERSON> und seinem <WORD> <PERSON>

(<PERSON> and his (dative) <WORD> <PERSON>)

<PERSON> langjähriger <WORD> <PERSON>

(<PERSON>('s) longtime <WORD> <PERSON>)

These patterns appear in contexts where <WORD> is substituted by terms such as “Freund” (friend), “Vertrauter” (intimate/close friend), “Kollege” (colleague), and so on.

More specific patterns that have significantly fewer applications, but appear to have high-precision, can also be found.

<PERSON> und Gitarrist <PERSON>

(<PERSON> and guitarist <PERSON>)

This pattern can be used to identify band members. Such patterns can usually not be found in relevant literature, but are useful in practice, because they can reliably identify and label an expressive semantic relationship between two people.

Low-precision Patterns

Several patterns have a high number of application, but comparatively low (or only moderate) precision. For instance:

<PERSON> und <WORD> <PERSON>

(<PERSON> and <WORD> <PERSON>)

This pattern can in several cases be used to identify relationships between people, such as in “Angela Merkel und Ehemann Joachim Sauer” (Angela Merkel and husband Joachim Sauer) or “Bundeskanzlerin Merkel und Innenminister Friedrich” (chancellor Merkel and secretary of state Friedrich). It this case, Jochim Sauer is the husband of Angela Merkel and Friedrich is secretary of state of (the cabinet of) Angela Merkel. However, many sentences do not imply this kind of relationship, such as in

Ein Dreivierteljahr ist es her, dass sich US-Präsident George W. Bush und Bundeskanzler Gerhard Schröder dort begegneten.

(Three quarters of a year have passed since US president George W. Bush and chancellor Gerhard Schröder have met there.)

where the keyword “Bundeskanzler” (chancellor) does not relate Schröder to Bush. It is also possible that the sentence shows a different structure altogether.

Similarly dependent on the situation is the pattern:

<PERSON> <WORD> <PERSON>

This works well if <WORD> is a verb, but yields incorrect tags otherwise.

General Observations

Some problems with patterns of lower precision stem from the simple approach of pattern generation and application, which does not take into account part of speech, sentence structure, and structural dependencies. This could be improved in the future by investigating the matter more closely, revising the classifier and expanding it by more elaborate methods from natural language processing and information extraction. On the other hand, this approach allows us to extract diverse lexico-syntactic patterns, which include, but are not limited to, patterns from related research literature.

We rely on user interaction to obtain information used for pattern generation, but derive the patterns without explicit manual construction or evaluation of the pattern by the user. Especially, users are not required to have any morpho-syntactic background knowledge and do not have to reason about the feasibility and potential quality of patterns. Combined with pattern extraction and application methods that are language-independent, this approach can be transferred to different languages in a straightforward manner, given users of the system have the respective language skills.

The precision of the classifier is given as an average of precisions for individual patterns. Since our classifier is designed to be trained, applied, and evaluated during operation of the system, it can be imagined to disable specific patterns and revoke their application once the system has enough evidence that their precision is low. By removing low- and keeping high-performance patterns, the classifier could self-improve over time, given enough user feedback.

8 Future Work

Several aspects of the system can be addressed to improve results and user experience of existing features. Future development and research could utilize the current stage Networks of Names and contribute to its expansion as well. Based on the current development state and our evaluation results, we identify the following possibilities for future work.

Preprocessing

We discussed the problem of **named entity disambiguation** in Section 4.2. Working with the system reveals that performing disambiguation and normalizing names could contribute to the quality of the dataset and usability of the system. For instance, in some situations during the user study, users felt inclined to use more than one node for one entity, because different spellings of the same entity were involved in different relationships (see "Lehman Brothers" in Figure 23). Additionally, unnormalized names appear in places where normalized names would feel more natural, e. g. as part of the tagging widget in the sources view.

Within the scope of this work, we focused on the extraction of people and organizations as entities. Another useful type of entity in the context of relationships between people and organizations are **events**. Event extraction, however, is rather different from the recognition of named entities in text, especially in the context of abstract events (such as a financial crisis) as opposed to more concrete events (like a conference). Thus, the inclusion of such a change would require significant work, in both research and implementation.

Networks of Names relies on newspaper text as sources. As a corpus, we used the Leipzig Corpora collection and focused on German text. The extension of the system to **include other sources**, especially recent ones, could allow users to not only explore relationships that are depicted in newspapers, but do so in the context of current events. One possibility for that are newspaper APIs, which were discussed in Section 4.1. Additional sources could also include newspapers in **different languages** to improve diversity and counteract the current bias towards German public life. This change would further increase heterogeneity and require a suitable concept for the human-computer interface.

Visual Interactive System

For the selection of interesting subgraphs, we applied *pointwise-mutual information* (PMI) as a metric for edge interestingness. However, several other metrics exist that could be employed instead of PMI [45]. It is possible that an evaluation of **alternative metrics** could improve search and expansion results in Networks of Names.

Our motivation to deviate from the original notion of DOI stems from the problem that it proved to be impractical for our use case. For future research, it could be investigated whether quantifiable metrics of networks and use cases exist that indicate the possibility to **reuse certain versions and parametrizations of DOI**.

Participants of the user study suggested that the possibility to view **details on nodes** (and not only on edges) could significantly improve the exploration process. This could be realized by retrieving information from a knowledge base and displaying it on screen. Doing so would require addressing the named entity disambiguation problem mentioned above.

Using **knowledge bases** for details on nodes could be extended to a more complete integration. Knowledge bases are designed to be machine readable and contain extensive information on the relationships of people and organizations based on some predefined ontology with relationships such as *married-to*, *CEO-of*, and similar. If high-quality knowledge on certain relationships between people and organization are already present, making it available in Networks of Names would constitute a useful baseline. Instead of generating this baseline relation information by their own interaction, users could spend their time assigning labels to other, unknown relationships. Likewise, information on non-taxonomic relationships obtained from the operation of Networks of Names could be used to enrich existing knowledge bases.

In some situations, users know that a relationship between two entities exist, but no source for that relationship is available in the system. To remedy this problem, users could be allowed to specify **tags with no evidence**, or to **add new sources manually**. The latter approach could be used to improve performance of automatic crawling of sources.

The user study showed that user feedback on automatic tags can be sparse. Therefore, it could be viable to investigate, implement, and evaluate additional possibilities and **incentives for users to validate** more tags. More feedback would also enable improvements of the classifier.

While we have argued the general suitability of Networks of Names to explore relationships and conducted a user experiment to generate data and explore possible impacts of a visualization on tagging behaviour of users, a **usability study** could reveal more information on how our system performs in aiding the user and supporting human cognition.

Classifier

We evaluated our classifier on the basis of data obtained from the user study. The results show a precision that is reasonable, but not particularly high. There exist several possibilities to improve on this: First, **more sophisticated methods for relationship extraction** could be employed. Since it is generally considered to play an important role in relationship extraction, part of speech tags, syntactic parse trees, or dependency trees could be considered to improve pattern generation or application [52]. Similarly, instead of working with lexico-syntactic patterns as we have done in the scope of this work, **other methods**, such as the ones referenced in Section 6, could be implemented instead. Both, the improvement of existing and the implementation of new methods could incorporate and benefit from the inherent user interaction.

As discussed with regard to the evaluation of the classifier in Section 7.3.2, the classifier could be extended to **dynamically disable patterns** if user feedback suggests a low performance.

9 Conclusion

Our aim was to employ automatic methods to make social network information contained in newspapers available in an accessible form.

We developed Networks of Names, a web-based interactive system that is capable of extracting relationships between people and organizations from large text corpora and making them available for visual exploration. The system incorporates current research from visual analytics and integrates methods from language technology and other fields of computer science.

For the exploration of large graphs, we built upon the notion of degree of interest (DOI) from [63] and modified it to operate on edges instead of nodes. We utilized pointwise-mutual information (PMI) to express the interestingness of edges. We found the combination of DOI for edges and PMI to work substantially better on our network than the original DOI metric, producing interesting subgraphs for most searches. We extend the original expansion algorithm to optionally work with more than one focal node and used this capability to implement a search for two nodes and a connection between them by calculating maximum-capacity paths.

Exploiting user interaction, we implemented a pattern-based classifier for the discovery and labelling of non-taxonomic relationships. The classifier is trained and applied during the operation of the system, without the necessity for users to deal directly with the design and evaluation of lexico-syntactic patterns. Feedback of users on the correctness of individual tags created by the classifier can be used to derive performance values for patterns.

We conducted a user experiment with the aim of testing the system, generating data for the classifier and exploring the impact of a visualization on the tagging behaviour of users, as opposed to a comparable text-based system. The results showed that our visual interactive system is suitable for the task of exploring the underlying network. We also found that visualization can have an impact on how user word relationship labels, in that they labelled more concisely and refrained from tagging extremely abstract or over-specific relationships.

Evaluating the performance of our classifier, we found that for our simple approach it has reasonably high precision. Without being predefined or manually crafted, the patterns with the highest precision turned out to correspond to patterns cited in research that deals with lexico-syntactical patterns.

Companion Website

A companion website for this thesis, where the source code of Networks of Names and a demo installation is available, can be found at `http://maggie.langtech.informatik.tu-darmstadt.de/thesis/master/NetworksOfNames/`.

A Labels in the User Experiment

Labels as entered by users in the course of the user experiment, sorted into three categories: Common Labels (where two labels are equal if they are exactly the same, phrases of each other, or frequent modifications). Table 6 shows common labels, while labels unique to the text and visual study, respectively, are listed in the subsequent sections.

Common Labels

Aufsichtsratsvorsitzender	Aufsichtsrat
Aufsichtsratsmitglied	Aufsichtsrat
kauft Anteile	kauft Anteile der
Autorin	Autor von
Amtskollegen	sind Amtskollegen
Bürgschaft	bestätigt Bürgschaft
Chef, Ex-Chef, Chef von	Chef, Chef von, Chef der, ist Chef der, ist Chef des, ist Chef von
CEO	CEO, CEO der
Doppelgänger	ist "Doppelgänger" von
Ermittlung, ermittelt gegen	in Ermittlungen, ermittelt gegen, führt Ermittlungsverfahren gegen
Einstieg	möglicher Einstieg bei
Forderung	fordert auf, reicht Forderungen ein
Freund, Freunde	Freund, Freunde, Freund von, befreundet, Freundschaft, Freundin von, beste Freunde, ist Freundin von, ist befreundet mit
Hauptfigur in	Hauptfigur in
Hauptrolle, Hauptrolle in	ist die Hauptrolle von
Kollegen	Kollege von, Kollegin von
Kabinett, Kabinettsmitglied	Kabinettskollege von
Kommandeur, neuer Kommandeur, alter Kommandeur	ist Kommandeur, ist Kommandeur von
Konkurrenten	Konkurrenten
Moderator	ist Moderator von
Mitschüler	Mitschüler, Mitschülerin
Mitglied	Mitglied
Mentor	ist Mentor von
Millionenspende	offenbar Millionenspende
Nachfolger, Nachfolgerin	Nachfolger, Nachfolgerin, Nachfolger von, Nachfolgerin von
Oberkommandeur	ist Oberkommandeur, ist Oberkommandeur der
Präsident	Präsident, Präsident des
Regisseur, Regisseur von	Regisseur von
Risikovorstand	Risikovorstand, Risikovorstand von
Rivalen	Rivalen
Spende	mögliche Spende
Schauspieler, Schauspielerin	Schauspieler, Schauspieler in, Schauspielerkollegen
ehemaliger Schatzmeister, ehemalige Schatzmeisterin	Schatzmeister, langjähriger Schatzmeister
Stratege	Stratege
Sohn, Sohn von	Sohn von, ältester Sohn von
Schauspielkollegen	Schauspiel
Vorsitz, Vorsitzende, Vorsitzender, ehemaliger Vorsitzende	Vorsitzende, Vorsitzender, Vorsitzender des, Vorsitzender von, ist Vorsitzender des, ist ehemaliger Vorsitzender des

Vorgesetzter	Vorgesetzter von
Vorbild	hat zum Vorbild, ist Vorbild von
Vorstand, Vorstandsmitglied, Vorstandsvorsitzender	Vorstand, ist Vorstand von, Vorstandsmitglied, Vorstandsvorsitzender, ist Vorstandsmitglied von, Vorstandsvorsitzender von, ist Vorstandsvorsitzender der, ist neuer Vorstandsvorsitzender von
alias	alias, heißt, steht für, Bezeichnung für
arbeiteten zusammen	arbeitet mit, arbeitete mit, arbeiten zusammen, arbeiteten gemeinsam, arbeiteten am selben Institut
bekannt	bekannt durch
getroffen	traf, trifft, treffen, trafen sich, treffen sich, treffen aufeinander, Teilnahme am Treffen
hat belastet	belastet
kontrolliert	kontrolliert
kauft	kauft
startet von	startet von
Rolle, spielt, spielt in, Darsteller von, schauspielert in, spielt die Rolle von	spielt, stellt dar, verkörpert, spielt den, gespielt von, besetzt Rolle, wird gespielt von, Schauspieler der Figur
unterstützt	unterstützt

Table 6: Matching between similar labels as entered by participants of the visual (left) versus the text study (right).

Unique Labels (Visualization)

Absolventen der gleichen Universitaet, An der Entscheidung beteiligt, Arbeitgeber, Aktionär, Büroleiter, Bandkollege bei Blur, Bewertung, Brüder, Belastung, Bundesvorsitzende, Briefwechsel, Crew, Datenkauf, Direktor des Instituts für Finanzwissenschaft der, Durchsuchung, Drehbuchautor von, Ex-Mitarbeiter, Ex-Lebensgefährten, Eigentümer, gleicher Entdeckungsdrang, ehemaliger Finanzberater, spielte in seinem Film, ehemaliger Generalbevollmächtigte der Bundesschatzmeisterei, Gegner, selbe Grabstätte, Generalsekretär, Gründer, Generalinspekteur, Governeur, Hauptaktionär, Heroinherstellung in, hat Haftbefehl erlassen gegen, ISAF-Offiziere, erwarten ein Kind, Krise, Kreditverkauf, Kosten, Komplizen, Liebespaar, hatte Lehrstuhl an, Mutterkonzern, Maschendrahtzaun, Mehrzahl, Machendrahtzaun, Mitarbeiter, Management, Nachname, beide Nobelpreistraeger, Nationale Sicherheitsberater, Oberst, Oberbefehlshaber, Produkt von, Paar, Rollen von Stallone, Rolle in Rocky Balboa, Roman von, Schmiergelder, deutsche Synchronstimme, Sitzung, Sprecher, Schauspielkollegen in Rocky IV, Schauspielkollegen in The Expandables, Synchronstimme, teilten Spendengeld, Tänzer bei, letzter Teil der Serie, Unterstützer, Vater von, Verkauf, Verteidigungsminister beteiligter Streitkräfte, Veranstalter, enge Vertraute, Verleger von, Zusicherung von Straffreiheit, bezahlt, collaborators, drehen zusammen, erstellt, gleichauf, glaubt nicht, glaubt an, hat bestochen, hat aufgekauft, kritisierte, klagt an, nicht getroffen, repariert, stellte bildlich dar, start von, transportiert, uneinig, verkauft, verhasst, verteidigt, Ökonom, Übernahmeangebot, Überweisung, Übernahme

Unique Labels (Text-Only)

ist gegenwärtiges Aufschtsratsmitglied genau wie, Arbeitgeber und Freund, übernahm das Amt von, muß sich nicht wegen Annahme einer Millionenspende verantworten, ist Antipode von, haben ökonomische Analysen gemacht, Austritt, in Affäre verstrickt, ist gegenwärtiges Aufsichtsratsmitglied genau wie, hat viele Abenteuer erlebt mit, korrigiert irreführende Aussage über, ist Aufsichtsratsmitglied genau wie, erhielt Airbus-Provisionen, Aufklärer, kommentiert die Arbeit von, Autoren von texten über das komplzierte Ökosystem Wald, bestätigt Aktien, Autokonzernchef, produzierten beide perfekte Abenteuerfilme, haben Abflug gemacht, ist die Abkürzung des, Agent, gespielt von, haben Affäre, Anwalt von, ist der Autor des Buches: The Trial of, Anklage, Autoren, kämpft gegen Auslieferungsersuchen, ist Achtplatzierter des, nimmt Aktie auf, Abkürzung für, Aufstieg, beantragte Auslieferung, verdrängte im Amt, weltbekannte Autoren, arbeiten bei der Deutschen Bank, Autorin soll abgekupfert haben, hatten Angststörungen, sind optimistischer Auffassung, größter Anteilseigner an, platziert Aktien, Automobilkonzernchefs, rufen die Arabischen Christen an mit, prominente Atheisten, verstorbene Autoren, hatten Auseinandersetzung, einer der wichtigsten Autoren des Verlegers, Arbeitswerttheorien, Aktuelle/ehemalige Mitglieder der Band Blur, für Auszeichnung vorgeschlagen, ist ebenfalls Autor wie, im Bundestag für, Bandkollegen, Brecht, sucht den deutschen Beitrag für das Final des, Brief an, übernimmt Bewertung, Bücher verboten, verbrannt, Bergriesen, Bücher von, Bücher ins Feuer geworfen, beide auf Berlinale, ist jüngster Bruder von, Biographien in englisch, fordert Bericht, ihre Bücher wurden verbrannt, habe den Befehl von, bestreitet Beteiligung, schrieb Brief an, Bartträger, Bücher als "schädlich und unerwünscht" eingestuft, Bürokollegen, unter den 100 Besten, in einer Beziehung, malte Bild von, sind Bandkollegen, übernimmt Bankgeschäft, Bücher nährten das Feuer, spielen in Bestsellerverfilmung, Bekanntmachung von Neuigkeiten zu, ist diese Beziehung leid, Bücher verdammt als "undeutsch", "dekandent", "zersetztend", verwenden nicht den Begriff Kapitalismus, Bestechung, Bücher verbrannt, waren im Bund der Gerechten, Biographien in englischer Sprache, hat Beziehung zu, trafen Blair, sind die schlimmsten Bluthunde, kamen nach Caputh, Chemiker, angeblich Chaos pur, CDU-Scharzgeldverwalter, Chefökonom bei, Chef Autokonzern, verkauft Chrysler an, hat die Castingshow gewonnen von, CSU-Spezi von, politisches Credo, von Daniel Barenboim interpretiert, Direktor, wird im Deutschen gerufen mit, Dokumente, gemainsam auf DNA, Daxunternehmen, erhielten Denkmäler, fährt für Deutschland zum, Diven, DNA auf Parfüm, Denker, Dikatator, gehören zum universitären Diskurs, Denkmal von, ihre Doppelgänger treten auf, sind Denkmalfiguren, haben gemeinsames Denkmal, Direx, bringt Dillinger zur Strecke, gespielt von, veröffentlichten bei den Dietz Verlag, kommen für Depotaufstockungen in Betracht, Dichter und Denker, große Dikatatoren, Deutschland-Chef, bittet für den Dialog mit, Erstausgaben in der Ausstellung, droht mit Enthüllungen, ist liiert mit der Ex-Freundin von, ist Entdecker und Gönner von, soll politischen Einfluss überprüfen, Ehrenträger, ist enttäuscht über Entscheidung von, teilen sich Erfahrungen, Ehepaar, hat Erwartung gegenüber, befürwortet Enteignung von, braucht Einsatz von, erhielten Ehrendoktorwürde, teile einstimmige Entscheidung des Vorstandes mit, wollten Erhebung des Proletariats miterleben, beschrieben Entwicklungsprozess des Kapitalismus, Erfinder von Spielfiguren, hat sich getrennt von Ex-Freundin von, erhoben gemeinsam Einwand, Erklärung unterzeichnet, stehen für ihre Epochen, Engagement, Emotion, optimistische Einschätzung gegenüber, keine Einmischung, teilte einstimmige Entscheidung des Vorstandes mit, ausgemacht als Empfänger der ATG-Gelder, Einschätzung für, kein Einstieg bei, Ex-Finanzberater, hätten E-mails geschrieben, Einschätzung, genehmigt Einzelnachweise von, tadelt Entscheidung von, bestreitet Einflussnahme von, Entwicklungsvorstand bei, negative Einschätzung gegenüber, gemeinsame Führung, lernt aus Fehlern von, beendet Freundschaft mit, Forscher, wirft Falschaussage vor, Filmfiguren, macht Fehler in, Führer, schuf die Figur, Frauen, spielt den Freund von, nationale Frage, waren keine Friedensfürsten, spielt die Freundin (Hermine) von, Führer von, ist Film von, erhöhte Figuren des 20. Jahrhunderts, gemainsam auf Foto, Figuren, Film von, forschten an Friedrich-Wilhelms-Universität, den Flammen übergeben, Fraktionsgeschäftsführer, spielte in seinen Filmen, aus Film bekannt, Fuchs-Panzer-Export, Filmfigur für den Synchronsprecher, von Fiati Musica Aperta gespielt, zwei der drei Figuren, überweist Geld, beschrieben Globalisierung, ist zu Gast bei, Gesellschaftsprägung, ist Gefährtin von, Gastgeber von, Geschäftsführer, Gegner Kapitalismus, in Göttingen, erhielt Geld von, will ihn vor Gericht, sind Gäste der G-8, produzieren gemeinsam die Gorillaz, waren "Große Linkshänder", beschafft sich Geld über, verkörpert Ginny Weasley in, beteiligt an GC, ist Gedicht von, gemeinsames Geschäftsfeld, geimeinsame Geldwäche, erhöht Gewinnschätzung, Gelehrte, internationale Größen, ähnliches Geschäftsfeld, General Manager, hat ein Gehöft in, gemainsam auf Geschäftszahlen, planen Hochzeit, gemainsam auf Hochstufungen, Hochzeit steht bevor, ggf. Hilfe bei Steuerermittlung gegen, Haftbefehl gegen, ist Halbbruder von, Herabstufung, kreatives Herz von, Harmonie, bat um Hilfe, Hauptdarsteller in, vor Hawking, ist Herzensfreundin von, Harry Potter, alias Hermine, befreundet mit, ist der Herr, Hollywoodgrößen, kauft Hypothekenspezialisten, Hauptdarsteller in Harry Potter, wurde fälschlicherweiser als Inspirationsquelle gehalten für Rowlings, einzigen unabhängigen Investmentbanken, vergewaltigten Indiana Jones, Im Labor der Friedrich-Wilhelms-Universität, hat angeblich ein Immunitätsabkommen mit, Ikonen, Ihre Bücher wurden ins Feuer geworfen, Investmentbanker, für Ideale gekämpft, sind Interviewpartner von Christiane Amanpour, gemainsam auf Investmentbankern, Institutsdirektor, Interesse an Kauf von, Ideologen, für Ideen gekämpft, drehten zusammen Indiana Jones, arbeitet an Institut, Ikonen der Gegenwart, Ihre Bücher wurden verbrannt, verkörpert Jesse James im Film, in Japan, fertigen in Japan, hatte von Juli-September 1932 einen Briefwechsel mit, aus einem Jahrhundert, beauftragten Jeff Nathanson, drängen die Justiz, spielt John Dillinger in, Kohls Helfer, Konkurrenz, Künstler und Intellektueller, gerettet durch Breker, einer der aussichtsreichsten Kandidaten für Nachfolge von, abgeschlagene Köpfe, wurde in das Kontrollgremium wiedergewählt genau wie, Künstler, machten Kampfansage, Kandidat Nachfolge von, Klassiker, hatten Kenntnis von Auslands-Millionen, hat Karriere begonnen bei, Kinderdarsteller, storniert Kredit, hatten gemeinsame Kritiker, gemeinsamer Kooperationspartner, ist bisheriger Kreditvorstand der, Kandidaten für Hilfspaket, ist Kreditvorstand von, Kumpane, Kongressteilnehmer, nahmen an Kongress teil, verantwortlich für schwarze Kassen bei, entdeckten Kernspaltung, verkauft riskanter Kredite an, Konzernchef von, gemeinsames Kind, stilisierten sich zum Künstler, führt Korruptionsverfahren gegen, Kunden der Kanadier, haben Kommen zugesagt, droht mit Konsequenzen, erhöht Kursziel von, gibt grünes Licht für Start, sind auf Liste des Guardian, hat eine Liste von Opfern übergeben an, Leben wird in romanesker Form anbgehandelt, ist Leitgröße von, Leiter des Büros von, hat Laudatio gehalten auf, hat ins Leben gerufen, ist Lieblingsfeind von, auf Liste, will erneut auf Leinwand bringen, ist Lebensgefährtin von, in Lateinamerika beliebt, Mitglieder der Leopoldina, Mitglieder der Band Blur, Mitstreiter von, Manager von, 500000 Mark zugeordnet, schrieben Manuskripte zusammen, verfassten Manifest, Milosevic als Wiedergänger von, 100000 Mark-Spende, sind MItglieder, attackierten Menschen, haben Machenschaften, eröffnen Messe, zum zweiten Mal gespielt von, ist Mädchen von, übergab eine Million Mark in bar, Massenschlächter, beleidigten Menschenverstand, ist Minister von, Mitglieder, Mitnominierter von, Mitglieder der Akademie, ist Mitkandidat von, Millionen-Spenden, verkörpern Mätressen, schrieben Manifest, soll eine Million Mark entgegengenommen haben, von, erhielt 3,8 Millionen an Schmiergeldern, schrieben Manifest gemeinsam, Mitbeschuldigte, dichteten das Manifest, Mitglied des Aufsichtsrats von, übergab Millionen-Koffer an, Mitaktionär von, Mitglieder Deutschen Bank, formulierten Manifest, eröffneten Messe, haben Mitkämpfer, schrieben das kommunistische Manifest, veröffentlichten Manifest, deckt Mitwisser des Scharzgeldsystems, ordentliche auswärtige Mitglieder, schenkt Motorrad, berühmte Mitglieder, Manifest herausgegeben, Massenmörder, veröffentlichten gemeinsam das Manifest, setzt neue Maßstäbe mit, haben möglicherweise bald Nachwuchs, Naturwissenschaftler, große Namen, Neoatheisten, sucht Nachfolger von, Naturforscher, Nobelpreisträger, gründeten "Norfolk"-Stiftung, brach die Nase von, schrieben Neujahrsgedichte, ist Oberkommandierender der, ist (nach dem Obersten Gericht in Kuala Lumpur) das allgemein arabische Wort für, Obmann, gemeinsame Opfer, Opposition von, stellten Osteuropa in Schatten, übergibt Posten, nutzt Prozessor, Preisträger, erhoben Protest, posieren auf Pressekonferenz, behebt Probleme, als Professor Snape in, Partei des Vositzenden, hat nichts an seinem Prinzip geändert, zählen zu PfB, Posten als Aufsichtsratsmitglied, Produzentenfreunde, übernimmt Posten von, spielt den Piratenkapitän, strategischer Partner von, Prominenz, Paten, lässt die Pläne überprüfen, historische Persönlichkeiten, teilten sich Polen auf, haben Pinguin-Doubles, nimmt an Pressekonferenz teil zum, Parteimitglied, diskutierten Politik, gibt es als Puppen, ist Pilot von, PDS Vorstand, gleiche Prognose, gemeinsame Partei, lehnten Parteipatriotismus ab, Person wechselt zu, teilten sich Restsaldo der "Nordfolk"-Konten, auf Rückkehr zu, Regierungschef Mitgliedstaats, könnte der Rückzug einen Friedensnobelpreis einbringen wie einst, stehen in einer Reihe, Repräsentanten der Wiener Klassik, liebten der Roadster, führt Regie über den Schauspieler, möchte Reform, Romanfigur von, Rede über Taiwan, spielt seine Rolle besser als, Regisseure, unter Regie von, Referenzkunde, erstes echtes 3d-Rennspiel von, zettelten zusammen Revolution an, wurden von Raab herausgebracht, haben zusätzliche Rückspiegel, Redetalente, sorgt für den Rücktritt vor von, verbunden mit Revolutionen in den Naturwissenschaften, Romantiker, boomende Raumwunder, deutsche Sozialisten, übernimmt Spitzenplatz, neue Stars, Sendung über, ist Sicherheitsberater von, als Staatsfeind Nummer eins, verfolgt durch, Spitze von, treten zusammen in einer Serie auf, deutscher Synchronsprecher von, Schulfreunde, kann wie er auch Sicht verändern, begründeten Sozialismus, steht an Startrampe von, sind auf Scheinen, nahm Spende an von, hat Straffreiheit zugesichert, im Suhrkamp Verlag erschienen, Spielehersteller, Spendenaffäre, sind Straßennamen, an der Spitze von, Synchronsprecher von, sind beide Schuld, kauft Steuerdaten von, gemeinsames Schaffen, spielt an Seite von, gemeinsame Schüler, Schüler in Hogwarts, steht an Seite von, tote Spitzenverdiener, bestreitet Spende an, auf Sie wird berufen, haben Sprachrohr, gemeinsame Sprache, Schriften, bügsierte Spende von, atmen Sauerstoff, ideologische Schlacht, großartige Schauspieler, hatten Sex?, sind Schwellenländer, ist Sprecher der Botschaft der, Schulden, Stammesgenossen, ist bekannt geworden durch eine Show von, steht im Schatten von, Sozialisten, Spitzenprodukt von, Stars, nutzen Serviceleistung von Akamai, sagt Start ab, unterschiedliche Strategien, Sonderberater von, macht sich Sorgen um, nimmt in Schutz, ist Sprecher der Mission in Afghanistan der, Sprecher der Hauptfiguren, auf Seite von, ist Schiff von, kommen im Spiel vor, kauft Steuerdaten der, Star in, Schriften erzielten hohe Preise, Schauspieler in Verfilmung von Autorin, brachte in Sicherheit, ist zu sehen an der Seite von, hatten wichtige Theorien, Tochterfirma von, auf Top 10 Liste, ist Tochterfirma von, hat eine Theorie aufgestellt in Anlehung an die Evolutionstheorie von, spielen zusammen in TV-Serie, will Tariferhöhungen korrigieren von, gleich auf bei Transaktionen, ist Teil von, Täger des Nobelpreises für Literatur, Treuhandverhältnis, umfangreiche Teile entnommen, Terroristen, treffen Therese Huber, ist Teil des, Teil des Topgremiums, ehemalige Tochterfirma von, sind Täter, Teil des Vorstandes, wichtige Theorien, Traumpaar, Teil der Bundeswehr, Teil der gerechtigkeitstheoretischen Diskussion, erhielt Teil einer Spende, übernimmt Teile, spielt Titelhelden, Teil des Spitzengremiums, haben Urenkel, keine Unterstützung für, sind Ungeheuer, fusioniert mit Unternehmen von, Uhrzeit, weist Vorwürfe zurück, hegt Verdacht, Vergewaltigt, Vorbilder, lieferte die Vorlage für einen Song von, verkehrten in Villa, enger Vertrauter von, Vertriebschef, Vertreter von, Vositzender des Kontrollgremiums, neuer Vorstandschef, Volkswirt, Vositzender, hat Vermutungen über, Villa in Südfrankreich gemietet, Verteidigungsminsiter Mitgliedstaats, hielt sich an Vorgaben von, bezogen nichtmarktmäßiges Verhalten ein, Verteidiger von, hat seinen Vertrag nicht verlängert, Vorreiter, Vorwurf der Fremdinspiration, VW, Vergichen mit, Vereinbarungen, Vorstandsvorsitzender der Tochterfirma von, gemeinsame Vereinbarung, Vertrag geschlossen, Vertrauter von, führten Völker ins Unglück, sind in Vergangenheit, Vorstandchef, Vorstandsmitglieder, Vorsitz Direktorenrat, ist Verteidigungsminister unter Präsident, schlimmste Verbrecher, Vorstand der Bank, ist das Wort für, Werke verbrannt, von Warhol gemalt, warten im Wald, Werke Opfer des Feuers, Wissenschaftler, Wiederaufnahme von Beobachtung, Werke wurden vorgestellt, hat die Wahl zum Generalinspekteur der Bundeswehr begrüßt, Weltreisende, "Wüstenbewohner", haben die Welt mit ihren Werken verändert, langjährige Weggefährten, nutzen Wasserzeichen, unbeeirrbar ihren Weg gegangen, möchte zur Wahrheit zwingen, ihre Werke werden verbrannt, nicht gewillt am Wettrüsten, gemeinsames Werk, Wertpapieridentifikation von, Werke ins Feuer, führende Wissenschaftler, im Wettbewerb, berühmte Wissenschaftler, Wissenschaft, treten auf Wunsch auf, gemainsam auf Werke, hatten zusammen Zugriff, setzt in Zugzwang, Zusammenbruch belastet, hat Zuschauer aufgefordert zu verprügeln, Ziel, ringen um Zuhörergunst, anders als, adelt, arbeitet bei, appeliert für, bedeutet, beneidet, begegnen sich, begegnet, beschenkt, begrüßt, beraten von, behauptet, nicht gespendet zu haben, beschuldigt, bürgt für, berät, beide durchsucht worden, beurteilt, beeindruckt von, bereit, sich vernehmen zu lassen, blond, denkt nicht mehr positiv über, darf unterstützen, distanziert sich von, empfing, ersetzt, empfahl, extremistisch, erzählt von, empfängt, entdeckte, ebenwürdig, empfiehlt ihm etwas, erfinderisch, erzählt über, fliegt zu, folgte, folg nach, feuert, feierten gemeinsam, fühlt sich berufen von, gewann mehr, gemeinsam geschrieben, gemainsam auf verkörpert, glauben was sie sagen, gibt frei, höher platziert, hob ab, haben bemerkt, haben sich getrennt, hebt ab von, hat berühmt gemacht, haben am besten abgeschnitten, in, ist, interessiert an, inspiriert von, inspizierte, ist nicht gestartet von, ist in, ist dankbar, konkurriert mit, korrespondiert mit, können zulegen, könnte gefährlich werden für, kritisiert, kommunistisch, kündigt, kann man miteinander vergleichen, lernten ihn kennen, lebte vor, landete, leitet, löst ab, mit ihnen fing es an, musizieren, musizieren gemeinsam, mimte, müssen weichen, mahn, nimmt teil am, präsentierte, präsentiert, plaziert, produzierte, polemisch, reiste zu, revolutionär, sind sich einig, spielte mit bei, sagt etwas über, steht, schrieb, spielten zuvor schon zusammen, startete von, sitzt vor, schwer belastet, sind verrückt, steht in, schrieben gemeinsam, stuft herab, schreiben sich, stammt von, spielt auch in, spielt mit in, spricht sich aus für, schüchtern nicht ein, stertete von, soll spielen, spricht für, sozialistisch, spricht bei, treten verliebt und glücklich und harmonisch auf, treten auf, trennt sich von, teilt mit über, treten an, tanzt bei, trimmt, taten es, traf nicht, totalitär, treten gemeinsam auf, traten gemeinsam auf, unbekannter als, unterscheiden sich, veröffentlichte, verlegt, vertreten in, verkörperte, verdammenswert, verfolgt, verloren mehr, veröffentlicht, verheiratet, waren sich einig, werden erwartet, wird geplaudert, würde gerne dabei haben, weist an, waren weg, wurden verhört, war, wurden nicht verschrottet, wird sprechen mit, wird ersetzt, wurde gespielt von, wendet sich an, werden gehasst, war beeindruckt von, wird geschrieben wie, war in, wurden geschlagen, will ihn wieder treffen, wird eng für beide, wird geleiten von, wurde unterstützt, zusammenkommen, zu sehen in, zusammen beschuldigt, zusammen verdächtigt, zufrieden mit, zeigt, beschäftigen sich mit Ökonomie, distanziert sich vom Übernahmeangebot der, übersetzt, überzeugt, übernimmt, übertrumpft

B Patterns in the Classifier Evaluation

Figure 7 shows patterns from the classifier evaluation, sorted by their performance (counting “close” results as accepted). Overall, the patterns were applied 162993 times, the number of sampled applications is 1000 (986 that were not “undecided”), which is a coverage of roughly 1%. The average precision is 0.53, 0.61 counting “close” results as accepted.

The column *applied* shows how often the pattern was applied, *sampled* how many instances we sampled, *accepted (close)* how many instances were accepted (including “close” instances in braces), *rejected* how many instances were rejected. The columns *p*, *p’* and *c* denote the performance, performance including “close” instances, and coverage for that pattern, respectively.

Pattern	applied	sampled	accepted (close)	rejected	p	p’	c
<PERSON/S>, <WORD> von <ORGANIZATION/O>	1594	8	8 (8)	0	1.00	1.00	0.0050
<PERSON/O>, <WORD> von <ORGANIZATION/S>	1594	7	7 (7)	0	1.00	1.00	0.0044
<PERSON/O>, <WORD> bei <ORGANIZATION/S>	1041	5	5 (5)	0	1.00	1.00	0.0048
<PERSON/S> <WORD> "<PERSON/O>	344	2	2 (2)	0	1.00	1.00	0.0058
<PERSON/O> <WORD> "<PERSON/S>	344	1	1 (1)	0	1.00	1.00	0.0029
<ORGANIZATION/S> hat das Kursziel für <ORGANIZATION/O>	77	1	1 (1)	0	1.00	1.00	0.0130
<PERSON/S> an der Seite von <PERSON/O>	75	2	2 (2)	0	1.00	1.00	0.0267
<PERSON/S> wirft <PERSON/O>	58	2	2 (2)	0	1.00	1.00	0.0345
<PERSON/O> langjähriger <WORD> <PERSON/S>	53	1	1 (1)	0	1.00	1.00	0.0189
<WORD> von Präsident <PERSON/O>, <PERSON/S>	53	1	1 (1)	0	1.00	1.00	0.0189
<PERSON/S> (<WORD> von <PERSON/O>	47	2	2 (2)	0	1.00	1.00	0.0426
<PERSON/S> und Justizminister <PERSON/O>	43	1	1 (1)	0	1.00	1.00	0.0233
<PERSON/S> trifft <PERSON/S>	33	1	1 (1)	0	1.00	1.00	0.0303
<ORGANIZATION/O>, Oberst <PERSON/S>	32	3	3 (3)	0	1.00	1.00	0.0938
<PERSON/O>, im Vorstand der <ORGANIZATION/S>	32	2	2 (2)	0	1.00	1.00	0.0625
<PERSON/O>, der Chef von <ORGANIZATION/S>	29	3	3 (3)	0	1.00	1.00	0.1034
<ORGANIZATION/O> mit <PERSON/S> an der <WORD>	27	2	2 (2)	0	1.00	1.00	0.0741
<ORGANIZATION/O>, US-General <PERSON/S>	22	3	3 (3)	0	1.00	1.00	0.1364
<PERSON/S> beim <WORD> des <ORGANIZATION/S>	22	2	2 (2)	0	1.00	1.00	0.0909
<PERSON/S> und Gitarrist <PERSON/S>	19	3	3 (3)	0	1.00	1.00	0.1579
<ORGANIZATION/S>, Goldman Sachs und <ORGANIZATION/S>	14	3	3 (3)	0	1.00	1.00	0.2143
<PERSON/S>, Jet Li, <PERSON/S>	10	2	2 (2)	0	1.00	1.00	0.2000
<PERSON/S>, Stalin und <PERSON/S>	9	3	3 (3)	0	1.00	1.00	0.3333
<PERSON/S>, ehemals <WORD> des <ORGANIZATION/O>	9	1	1 (1)	0	1.00	1.00	0.1111
<ORGANIZATION/O> an den Finanzinvestor <ORGANIZATION/S>	8	2	2 (2)	0	1.00	1.00	0.2500
<PERSON/S> geht davon aus, dass die <ORGANIZATION/O>	8	2	2 (2)	0	1.00	1.00	0.2500
<WORD> der Afghanistan-Schutztruppe <ORGANIZATION/O>, US-General <PERSON/S>	7	2	2 (2)	0	1.00	1.00	0.2857
<ORGANIZATION/S> hat eine Bürgschaft für den Immobilienfinanzierer <ORGANIZATION/O>	7	1	1 (1)	0	1.00	1.00	0.1429
<ORGANIZATION/S>) den <WORD> der angeschlagenen <ORGANIZATION/O>	6	2	2 (2)	0	1.00	1.00	0.3333
<PERSON/S> und Lord <PERSON/S>	6	2	2 (2)	0	1.00	1.00	0.3333
<ORGANIZATION/S> <WORD> Teile von <ORGANIZATION/O>	6	2	2 (2)	0	1.00	1.00	0.3333
<WORD> aus Afghanistan und Pakistan, <PERSON/S> und Asif <PERSON/S>	6	2	2 (2)	0	1.00	1.00	0.3333
<ORGANIZATION/S> hilfesuchend an die <ORGANIZATION/O>	6	2	2 (2)	0	1.00	1.00	0.3333
<PERSON/S>, dessen <WORD> die <ORGANIZATION/O>	6	1	1 (1)	0	1.00	1.00	0.1667
<PERSON/S>, Fiat-Chef Sergio Marchionne, GM-Europachef <PERSON/S>	5	4	4 (4)	0	1.00	1.00	0.8000
<PERSON/S> hat die Wahl von General <PERSON/O>	5	3	3 (3)	0	1.00	1.00	0.6000
<PERSON/S> und dem japanischen Ministerpräsidenten <PERSON/S>	5	2	2 (2)	0	1.00	1.00	0.4000
<ORGANIZATION/S> <WORD> mit 35 Milliarden Euro für die Rettung von <ORGANIZATION/O>	5	2	2 (2)	0	1.00	1.00	0.4000
<PERSON/S> und <PERSON/S> nur noch zwei unabhängige <WORD>	5	2	2 (2)	0	1.00	1.00	0.4000
<PERSON/S>, Mickey Rourke, <PERSON/S>	5	1	1 (1)	0	1.00	1.00	0.2000
<PERSON/S>, Risikomanager <PERSON/S>	5	1	1 (1)	0	1.00	1.00	0.2000

<ORGANIZATION/S> hat eigene irreführende Aussagen über die Zukunft der Münchner <ORGANIZATION/O>	5	1	1 (1)	0	1.00	1.00	0.2000
<PERSON/S>, <WORD> der <ORGANIZATION/O>	7367	41	40 (40)	1	0.98	0.98	0.0056
<ORGANIZATION/O>-<WORD> <PERSON/S>	9223	52	45 (50)	2	0.87	0.96	0.0056
<WORD> von <ORGANIZATION/O>, <PERSON/S>	2134	15	12 (15)	0	0.80	1.00	0.0070
<WORD> des <ORGANIZATION/O>, <PERSON/S>	8043	40	30 (39)	1	0.75	0.98	0.0050
<ORGANIZATION/S>- <WORD> <PERSON/O>	6020	32	24 (32)	0	0.75	1.00	0.0053
<PERSON/S>, der <WORD> der <ORGANIZATION/O>	1071	4	3 (4)	0	0.75	1.00	0.0037
<WORD> der <ORGANIZATION/O>, <PERSON/S>	14073	77	55 (76)	1	0.71	0.99	0.0055
<WORD> wie <PERSON/S> und <PERSON/S>	2208	12	8 (11)	1	0.67	0.92	0.0054
<PERSON/O> und seinem <WORD> <PERSON/S>	389	3	2 (3)	0	0.67	1.00	0.0077
<PERSON/S>" von <PERSON/O>	302	3	2 (2)	1	0.67	0.67	0.0099
<PERSON/S> und seiner <WORD> <PERSON/S>	132	3	2 (2)	1	0.67	0.67	0.0227
<ORGANIZATION/S> hat die Aktien der <ORGANIZATION/O>	53	3	2 (2)	1	0.67	0.67	0.0566
<WORD>" zwischen <PERSON/S> und <PERSON/S>	21	3	2 (2)	1	0.67	0.67	0.1429
<WORD> wie <PERSON/S>, <PERSON/S>	3616	22	13 (17)	5	0.59	0.77	0.0061
<PERSON/S> <WORD> <PERSON/O>	17649	80	45 (45)	35	0.56	0.56	0.0045
<PERSON/S>, dem <WORD> der <ORGANIZATION/O>	277	2	1 (2)	0	0.50	1.00	0.0072
<PERSON/S> (l.) und <PERSON/S>	248	2	1 (1)	1	0.50	0.50	0.0081
<WORD> von <PERSON/S> und <ORGANIZATION/S>	191	2	1 (1)	1	0.50	0.50	0.0105
<PERSON/S> spielt <PERSON/O>	68	2	1 (1)	1	0.50	0.50	0.0294
<WORD> der <ORGANIZATION/S> bei der <ORGANIZATION/O>	61	2	1 (2)	0	0.50	1.00	0.0328
<PERSON/S> und <PERSON/S>, den beiden <WORD>	31	2	1 (2)	0	0.50	1.00	0.0645
<ORGANIZATION/O>"-<WORD>s <PERSON/S>	25	2	1 (2)	0	0.50	1.00	0.0800
<PERSON/S> aus dem <WORD> der <ORGANIZATION/O>	21	2	1 (1)	1	0.50	0.50	0.0952
<PERSON/S>, Sophia Loren, <PERSON/S>	9	2	1 (1)	1	0.50	0.50	0.2222
<ORGANIZATION/O>» mit <PERSON/S>	9	2	1 (1)	1	0.50	0.50	0.2222
<PERSON/O> den Einstieg bei <ORGANIZATION/S>	6	2	1 (1)	1	0.50	0.50	0.3333
<PERSON/S>, der sich von <PERSON/O>	5	2	1 (1)	1	0.50	0.50	0.4000
<PERSON/O> <WORD> <PERSON/S>	17649	78	32 (36)	42	0.41	0.46	0.0044
<WORD> für <PERSON/S> und <PERSON/S>	454	3	1 (2)	1	0.33	0.67	0.0066
<WORD> der <ORGANIZATION/S> im Bundestag, <PERSON/O>	235	3	1 (3)	0	0.33	1.00	0.0128
<PERSON/S> <WORD>e, dass <PERSON/O>	42	3	1 (1)	2	0.33	0.33	0.0714
<WORD> von <PERSON/S> und von <PERSON/S>	25	3	1 (1)	2	0.33	0.33	0.1200
<PERSON/O>, soll <PERSON/S>	11	3	1 (1)	2	0.33	0.33	0.2727
<PERSON/S> und <PERSON/S> wird es <WORD>	7	3	1 (1)	2	0.33	0.33	0.4286
<PERSON/S> und <PERSON/S> damals <WORD>	7	3	1 (2)	1	0.33	0.67	0.4286
<PERSON/S> und <PERSON/S> - als <WORD>	7	3	1 (1)	2	0.33	0.33	0.4286
<ORGANIZATION/O> hat <WORD> <PERSON/S>	727	4	1 (1)	3	0.25	0.25	0.0055
<PERSON/S>, <WORD> von "<PERSON/O>	13	4	1 (1)	3	0.25	0.25	0.3077
<PERSON/O>, <WORD> <PERSON/S>	7943	34	7 (7)	27	0.21	0.21	0.0043
<ORGANIZATION/O> <WORD> <PERSON/S>	5919	28	5 (8)	20	0.18	0.29	0.0047
<PERSON/O> und <WORD> <PERSON/S>	20297	98	16 (16)	82	0.16	0.16	0.0048
<ORGANIZATION/S>, die <ORGANIZATION/S>	1455	7	1 (1)	6	0.14	0.14	0.0048
<WORD> von <PERSON/S> und <PERSON/S>	5983	32	1 (2)	30	0.03	0.06	0.0053
<ORGANIZATION/O>- <WORD> <PERSON/S>	6020	34	0 (33)	1	0.00	0.97	0.0056
<ORGANIZATION/S> <WORD> <PERSON/O>	5919	28	0 (10)	18	0.00	0.36	0.0047
<ORGANIZATION/S> <WORD> <ORGANIZATION/O>	4645	25	0 (1)	24	0.00	0.04	0.0054
<WORD> von <PERSON/S>, <PERSON/S>	3988	26	0 (2)	24	0.00	0.08	0.0065
<WORD>n wie <PERSON/S> oder <PERSON/S>	639	1	0 (0)	1	0.00	0.00	0.0016
<WORD>s von <PERSON/S> und <PERSON/S>	438	1	0 (0)	1	0.00	0.00	0.0023
<PERSON/O>) und <PERSON/S>	249	2	0 (0)	2	0.00	0.00	0.0080
<PERSON/S>- <PERSON/S>	244	2	0 (0)	2	0.00	0.00	0.0082
<PERSON/S> und Bundeskanzlerin <PERSON/S>	189	3	0 (0)	3	0.00	0.00	0.0159
<PERSON/S> und Präsident <PERSON/O>	185	2	0 (0)	2	0.00	0.00	0.0108
<PERSON/S> oder <PERSON/S> be<WORD>	133	2	0 (0)	2	0.00	0.00	0.0150
<ORGANIZATION/O> will die <ORGANIZATION/S>	98	1	0 (0)	1	0.00	0.00	0.0102
<WORD>, darunter <PERSON/S> und <PERSON/S>	75	1	0 (1)	0	0.00	1.00	0.0133
<PERSON/O>, Wirtschaftsminister <PERSON/S>	70	1	0 (0)	1	0.00	0.00	0.0143
<PERSON/S> <WORD> das <ORGANIZATION/O>	60	1	0 (0)	1	0.00	0.00	0.0167
<WORD>es von <PERSON/S> und <PERSON/S>	59	1	0 (0)	1	0.00	0.00	0.0169
<PERSON/S>, dass <PERSON/O>	43	1	0 (0)	1	0.00	0.00	0.0233
<PERSON/S> und Produzent <PERSON/S>	28	3	0 (0)	3	0.00	0.00	0.1071
<PERSON/S>, Stalin, <PERSON/S>	23	3	0 (0)	3	0.00	0.00	0.1304
<PERSON/S>" und "<ORGANIZATION/S>	21	1	0 (0)	1	0.00	0.00	0.0476
<WORD> des <ORGANIZATION/O> hat <PERSON/S>	21	1	0 (0)	1	0.00	0.00	0.0476
<PERSON/S> <WORD>n Freund <PERSON/S>	16	1	0 (0)	1	0.00	0.00	0.0625
<ORGANIZATION/S>, der von <PERSON/O>	16	1	0 (0)	1	0.00	0.00	0.0625
<PERSON/S> <WORD> den von der <ORGANIZATION/O>	12	3	0 (0)	3	0.00	0.00	0.2500

<PERSON/S> hatte die <ORGANIZATION/O> aufge<WORD>	12	2	0 (0)	2	0.00	0.00	0.1667
<PERSON/S> (CDU) hat <PERSON/O>	12	1	0 (0)	1	0.00	0.00	0.0833
<WORD>n gehörten <PERSON/S>, <PERSON/S>	11	3	0 (3)	0	0.00	1.00	0.2727
<ORGANIZATION/S> wurde von <PERSON/O>	10	1	0 (0)	1	0.00	0.00	0.1000
<PERSON/S> kann sich die <ORGANIZATION/O>	9	3	0 (0)	3	0.00	0.00	0.3333
<PERSON/S> und Drehbuchautor <PERSON/S>	7	3	0 (0)	3	0.00	0.00	0.4286
<WORD> des afghanischen Präsidenten <PERSON/O>, <PERSON/S>	6	3	0 (3)	0	0.00	1.00	0.5000
<PERSON/S> und den Briten <PERSON/S>	6	3	0 (0)	3	0.00	0.00	0.5000
<PERSON/S>" <WORD> "<PERSON/O>	6	2	0 (0)	2	0.00	0.00	0.3333
<PERSON/O> und Lord <PERSON/S>	5	1	0 (0)	1	0.00	0.00	0.2000
<PERSON/S>, die "<PERSON/O>	5	1	0 (0)	1	0.00	0.00	0.2000

Table 7: Patterns and their performance.

List of Figures

List of Tables

References

[1] Thomas Abeel, Yves Van de Peer, and Yvan Saeys. Java-ML: A Machine Learning Library. *The Journal of Machine Learning Research*, 10:931–934, 2009.

[2] Nathalie Aussenac-Gilles and Marie-Paule Jacques. Designing and Evaluating Patterns for Relation Acquisition from Texts with Caméléon. *Terminology*, 14(1):45–73, 2008.

[3] Nguyen Bach and Sameer Badaskar. A Review of Relation Extraction. *Literature Review for Language and Statistics II*, 2007.

[4] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open Information Extraction from the Web. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2670–2676, Hyderabad, 2007.

[5] Albert-László Barabási and Réka Albert. Emergence of Scaling in Random Networks. *Science*, 286 (5439):509–512, 1999.

[6] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media*, San Jose, 2009.

[7] Chris Bennett, Jody Ryall, Leo Spalteholz, and Amy Gooch. The Aesthetics of Graph Visualization. In *Proceedings of the 3rd Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, pages 57–64, Banff, 2007.

[8] Chris Biemann and Uwe Quasthoff. Networks Generated from Natural Language Text. In Niloy Ganguly, Andreas Deutsch, and Animesh Mukherjee, editors, *Dynamics On and Of Complex Networks*, Modeling and Simulation in Science, Engineering and Technology, pages 167–185. Birkhäuser, Boston, 2009.

[9] Gerlof Bouma. Normalized (Pointwise) Mutual Information in Collocation Extraction. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, pages 31–40, Potsdam, 2009.

[10] Bundesverband Deutscher Zeitungsverleger e.V. Die deutschen Zeitungen in Zahlen und Daten, 2011. `http://www.bdzv.de/fileadmin/bdzv_hauptseite/markttrends_daten/wirtschaftliche_lage/2011/assets/ZahlenDaten_2011.pdf`, accessed on September 23rd 2013.

[11] Nan Cao, Jimeng Sun, Yu-Ru Lin, David Gotz, Shixia Liu, and Huamin Qu. FacetAtlas: Multifaceted Visualization for Rich Text Corpora. *IEEE Transactions on Visualization and Computer Graphics*, 16 (6):1172–1181, 2010.

[12] M.S.T. Carpendale. Considering Visual Variables as a Basis for Information Visualisation. Technical Report 2001-693-16, University of Calgary, Calgary, 2004.

[13] Duen Horng Chau, Aniket Kittur, Jason I. Hong, and Christos Faloutsos. Apolo: Making Sense of Large Network Data by Combining Rich User Interaction and Machine Learning. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 167–176, Vancouver, 2011.

[14] Philipp Cimiano, Paul Buitelaar, and Johanna Völker. Ontology Construction. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*, Machine Learning & Pattern Recognition, pages 577–604. Chapman & Hall/CRC, Boca Raton, 2010.

[15] Aaron Clauset, Cosma Rohilla Shalizi, and Mark E.J. Newman. Power-law Distributions in Empirical Data. *SIAM Review*, 51(4):661–703, 2009.

[16] Reuven Cohen and Shlomo Havlin. Scale-free Networks are Ultrasmall. *Physical Review Letters*, 90 (5), 2003.

[17] Marian Dörk, Nathalie Henry Riche, Gonzalo Ramos, and Susan Dumais. PivotPaths: Strolling through Faceted Information Spaces. *IEEE Transactions on Visualization and Computer Graphics*, 18 (12):2709–2718, 2012.

[18] Manaal Faruqui and Sebastian Padó. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of the Conference on Natural Language Processing*, Saarbrücken, 2010.

[19] Ivan P. Fellegi and Alan B. Sunter. A Theory for Record Linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.

[20] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370, Ann Arbor, 2005.

[21] George W. Furnas. Generalized Fisheye Views. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 16–23, Boston, 1986.

[22] Mohammad Ghoniem, Jean-Daniel Fekete, and Philippe Castagliola. A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 17–24, Austin, 2004.

[23] Roxana Girju and Dan I. Moldovan. Text Mining for Causal Relations. In *Proceedings of the 15th International Florida Artificial Intelligence Research Society Conference*, pages 360–364, Pensacola, 2002.

[24] Roxana Girju, Adriana Badulescu, and Dan Moldovan. Learning semantic constraints for the automatic discovery of part-whole relations. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 1–8, Edmonton, 2003.

[25] Michelle Girvan and Mark E.J. Newman. Community Structure in Social and Biological Networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[26] Scott Golder and Bernardo A. Huberman. The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, 2006.

[27] Ido Guy, Adam Perer, Tal Daniel, Ohad Greenshpan, and Itai Turbahn. Guess Who?: Enriching the Social Graph Through a Crowdsourcing Game. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 1373–1382, Vancouver, 2011.

[28] Harry Halpin, Valentin Robu, and Hana Shepherd. The Complex Dynamics of Collaborative Tagging. In *Proceedings of the 16th international conference on World Wide Web*, pages 211–220, Banff, 2007.

[29] Xianpei Han and Jun Zhao. Named Entity Disambiguation by Leveraging Wikipedia Semantic Knowledge. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 215–224, Hong Kong, 2009.

[30] Sudheendra Hangal, Diana MacLean, Monica S. Lam, and Jeffrey Heer. All Friends are Not Equal: Using Weights in Social Graphs to Improve Search. In *Proceedings of the 4th International Workshop on Social Network Mining and Analysis*, Washington, D.C., 2010.

[31] Marti A. Hearst. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, pages 539–545, Nantes, 1992.

[32] Mark Huisman and Marijtje A.J. Van Duijn. Software for Social Network Analysis. In Peter J Carrington, John Scott, and Stanley Wasserman, editors, *Models and Methods in Social Network Analysis*, pages 270–316. Cambridge University Press, Cambridge, 2005.

[33] Piers Ingram, Michael Stumpf, and Jaroslav Stark. Network Motifs: Structure Does Not Determine Function. *BMC Genomics*, 7(108), 2006.

[34] Martin Kavalec and Vojtech Svaték. A Study on Automated Relation Labelling in Ontology Learning. In Paul Buitelaar, Philipp Cimiano, and Bernado Magnini, editors, *Ontology Learning from Text: Methods, Evaluation and Applications*, Frontiers in Artificial Intelligence and Applications, pages 44–58. IOS Press, Amsterdam, 2005.

[35] Daniel Keim, Jörn Kohlhammer, Geoffrey Ellis, and Florian Mansmann, editors. *Mastering The Information Age - Solving Problems with Visual Analytics*. Florian Mansmann, 2010. `http://www.vismaster.eu/wp-content/uploads/2010/11/VisMaster-book-lowres.pdf`, accessed on September 23rd 2013.

[36] Daniel A. Keim, Florian Mansmann, Jorn Schneidewind, and Hartmut Ziegler. Challenges in Visual Data Analysis. In *Proceedings of the 10th International Conference on Information Visualisation*, pages 9–16, London, 2006.

[37] John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, Montreal, 2001.

[38] Navneet Malpani and Jianer Chen. A Note on Practical Construction of Maximum Bandwidth Paths. *Information Processing Letters*, 83(3):175–180, 2002.

[39] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, 2008.

[40] Thorsten May, Martin Steiger, James Davey, and Jörn Kohlhammer. Using Signposts for Navigation in Large Graphs. *Computer Graphics Forum*, 31(3pt2):985–994, 2012.

[41] David Nadeau and Satoshi Sekine. A Survey of Named Entity Recognition and Classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[42] Vivi Nastase, Preslav Nakov, Diarmuid Ó Séaghdha, and Stan Szpakowicz. Semantic relations between nominals. *Synthesis Lectures on Human Language Technologies*, 6(1):1–119, 2013.

[43] Hiroaki Ohshima and Katsumi Tanaka. High-speed Detection of Ontological Knowledge and Bidirectional Lexico-Syntactic Patterns from the Web. *JSW*, 5(2):195–205, 2010.

[44] Joshua O'Madadhain, Danyel Fisher, Padhraic Smyth, Scott White, and Yan-Biao Boey. Analysis and Visualization of Network Data using JUNG. *Journal of Statistical Software*, 10:1–35, 2005.

[45] Pavel Pecina. An Extensive Empirical Study of Collocation Extraction Methods. In *Proceedings of the ACL Student Research Workshop*, pages 13–18, Stroudsburg, 2005.

[46] Adam Perer, Ido Guy, Erel Uziel, Inbal Ronen, and Michal Jacovi. Visual Social Network Analytics for Relationship Discovery in the Enterprise. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 71–79, Providence, 2011.

[47] Martin F. Porter. An Algorithm for Suffix Stripping. *Program: Electronic Library and Information Systems*, 14(3):130–137, 1980.

[48] Abraham P. Punnen. A Linear Time Algorithm for the Maximum Capacity Path Problem. *European Journal of Operational Research*, 53(3):402–404, 1991.

[49] Uwe Quasthoff, Matthias Richter, and Chris Biemann. Corpus Portal for Search in Monolingual Corpora. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pages 1799–1802, Genoa, 2006.

[50] Lev Ratinov and Dan Roth. Design Challenges and Misconceptions in Named Entity Recognition. In *Proceedings of the 13th Conference on Computational Natural Language Learning*, pages 147–155, Boulder, 2009.

[51] David Sánchez and Antonio Moreno. Learning Non-taxonomic Relationships from Web Documents for Domain Ontology Construction. *Data & Knowledge Engineering*, 64(3):600–623, 2008.

[52] Sunita Sarawagi. Information Extraction. *Foundations and Trends in Databases*, 1(3):261–377, 2008.

[53] Prithviraj Sen. Collective Context-Aware Topic Models for Entity Disambiguation. In *Proceedings of the 21st International Conference on World Wide Web*, pages 729–738, Lyon, 2012.

[54] Ben Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343, Boulder, 1996.

[55] Roberta Smith. Mark Lombardi, 48, an Artist Who Was Inspired by Scandals. The New York Times, 2000. `http://www.nytimes.com/2000/03/25/arts/mark-lombardi-48-an-artist-who-was-inspired-by-scandals.html`, accessed on September 23rd 2013.

[56] Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. Learning Syntactic Patterns for Automatic Hypernym Discovery. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1297–1304. MIT Press, Cambridge, 2005.

[57] Charles Sutton and Andrew McCallum. An Introduction to Conditional Random Fields. *Machine Learning*, 4(4):267–373, 2011.

[58] James J. Thomas and Kristin A. Cook. A visual analytics agenda. *IEEE Computer Graphics and Applications*, 26(1):10–13, 2006.

[59] Ioannis Tollis, Peter Eades, Giuseppe Di Battista, and Loannis Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, Upper Saddle River, 1998.

[60] J. Trant. Studying Social Tagging and Folksonomy: A Review and Framework. *Journal of Digital Information*, 10(1), 2009.

[61] Peter D. Turney. Expressing Implicit Semantic Relations without Supervision. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 313–320, Sydney, 2006.

[62] Stijn Marinus van Dongen. *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht, 2000.

[63] Frank van Ham and Adam Perer. "Search, Show Context, Expand on Demand": Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):953–960, 2009.

[64] Marcos R. Vieira, Humberto Luiz Razente, Maria Camila Nardini Barioni, Marios Hadjieleftheriou, Divesh Srivastava, Caetano Traina, and Vassilis J. Tsotras. On Query Result Diversification. In *Proceeding of the 27th IEEE International Conference on Data Engineering*, pages 1163–1174, Hannover, 2011.

[65] Tatiana von Landesberger, Melanie Görner, Robert Rehner, and Tobias Schreck. A System for Interactive Visual Analysis of Large Graphs Using Motifs in Graph Editing and Aggregation. In *Proceedings of the Vision, Modeling, and Visualization Workshop 2009*, pages 331–340, Braunschweig, 2009.

[66] Hanna M. Wallach. Conditional Random Fields: An Introduction. Technical Report MS-CIS-04-21, University of Pennsylvania, Philadelphia, 2004.

[67] Jeremy M. Wolfe. Guided Search 2.0 A revised model of visual search. *Psychonomic Bulletin & Review*, 1(2):202–238, 1994.

[68] Jing Yang, Yujie Liu, Xin Zhang, Xiaru Yuan, Ye Zhao, Scott Barlowe, and Shixia Liu. PIWI: Visually Exploring Graphs Based on Their Community Structure. *IEEE Transactions on Visualization and Computer Graphics*, 19(6):1034–1047, 2013.