Eurographics Conference on Visualization (EuroVis) 2014
H. Carr, P. Rheingans, and H. Schumann
(Guest Editors)

*Volume 33 (2014), Number 3*

# Networks of Names: Visual Exploration and Semi-Automatic Tagging of Social Networks from Newspaper Articles

A. Kochtchi[1], T. von Landesberger[2], and C. Biemann[1]

[1]Language Technology Group, TU Darmstadt, Germany & [2]Interactive Graphics Systems Group, TU Darmstadt, Germany

**Abstract**

*Understanding relationships between people and organizations by reading newspaper articles is difficult to manage for humans due to the large amount of data. To address this problem, we present and evaluate a new visual analytics system, which offers interactive exploration and tagging of social networks extracted from newspapers. For the visual exploration of the network, we extract "interesting" neighbourhoods of nodes, using a new degree of interest (DOI) measure based on edges instead of nodes. It improves the seminal definition of DOI, which we find to produce the same "globally interesting" neighbourhoods in our use case, regardless of the query. Our approach allows answering different user queries appropriately, avoiding uniform search results.*
*We propose a user-driven pattern-based classifier for discovery and tagging of non-taxonomic semantic relations. Our approach does not require any a-priori user knowledge, such as expertise in syntax or pattern creation. An evaluation shows that our classifier is capable of identifying known lexico-syntactic patterns as well as various domain-specific patters. Our classifier yields good results already with a small amount of training, and continuously improves through user feedback.*
*We conduct a user study to evaluate whether our visual interactive system has an impact on how users tag relationships, as compared to traditional text-based interfaces. Study results suggest that users of the visual system tend to tag more concisely, avoiding too abstract or overly specific relationship labels.*

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces (D.2.2, H.1.2, I.3.6)—Interaction styles (e. g. , commands, menus, forms, direct manipulation) I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types

## 1. Introduction

People and organizations interact and influence their environment, the society, and public policy. Information on the interactions of people and organizations is encoded by print media as natural language text. To extract the information on relations from these texts, it is necessary to read and understand them. However, with more than 350 daily newspapers published in Germany alone [Bun11] and rising numbers of online publications, the task of conducting extensive research in newspaper articles of even a single day becomes increasingly difficult.

In the late 20th century, *Mark Lombardi* analysed political and financial scandals by collecting and organizing information from newspapers into a collection of hand-written cards [Smi00]. He showed his results in so-called *narrative structures*, a form that is more accessible and aesthetically pleasing for humans. Conceptually, narrative structures are node-link-diagrams of an underlying social network.

Following Lombardi's approach, we extract and visualize social network information derived from newspapers. Unlike Lombardi, we collect, organize, and visualize the information automatically (for a narrative structure created with Networks of Names, see Figure 1). This approach poses a number of computational problems: First, the understanding of social relationships remains a challenging automation task [KS05]. Second, the underlying social network is potentially large and thus cannot be visualized in a straightforward manner [KMSZ06].

Most research in natural language processing and text mining focuses on development, application, and quantitative evaluation of specific methods in specific domains. In particular, the same is true for the fields of entity recognition

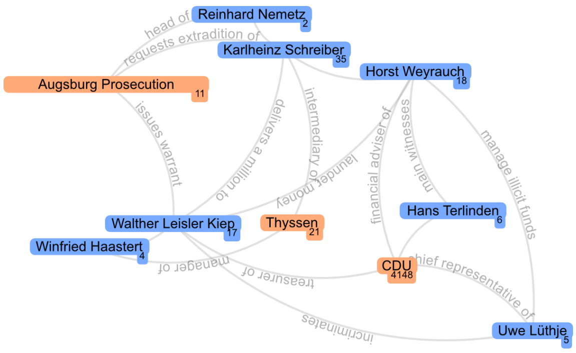



**Figure 1:** *Narrative structure created with Networks of Names: The CDU donations scandal (1999), see* `http://en.wikipedia.org/wiki/CDU_donations_scandal`.

and relation extraction. Results are usually not visualized, or visualized statically for a specific dataset. On the other hand, research in text and network visualization is typically focused on developing and testing visualization concepts that provide the means of exploring specific types of data.

We create a system that enhances state-of-the-art methods from language technology and network visualization for exploration and tagging of relationships from newspapers. Our contributions are:

1. We combine text mining, network visualization, and pattern-based semi-automatic relationship discovery and tagging into one single interactive system.
2. We propose an alternative version of degree of interest that is based on the interestingness of edges instead of nodes, because the seminal method [vHP09] proves unsuitable for our case.
3. We evaluate differences in user behaviour when tagging relationships using a visual versus a text-based system.

This paper is structured as follows: Related work is discussed in Section 2. We present our approach, including network extraction, network exploration techniques, and automatic discovery and tagging of non-taxonomic relationships in Section 3. Details on our user study and evaluation results are given in Section 4, followed by a conclusion in Section 5.

## 2. Related Work

This section covers relevant prior work from the fields of text and network visualization, visual analytics, and non-taxonomic relation extraction.

### 2.1. Text and Network Visualization

*Text visualization* attempts to turn large text corpora into more accessible visual representations. Some approaches create networks from text, for instance by plotting words that appear together in certain contexts as a node-link diagram. **Phrase Nets** [vHWV09] is applied to specific user-selected phrases. The visual exploration of newspaper articles, in particular, was addressed in [GM04] with a tool called **Contexter**. However, only basic query-response interaction is supported, showing most frequent entity collocations as results.

Contemporary systems for *network visualization* must deal with large graphs, for which a straightforward visualization of the complete graph is unreasonable. Adapting the visual analytics mantra [KMSZ06] to the exploration of large graphs, van Ham and Perer devise their approach “Search, Show Context, Expand on Demand” [vHP09], where the user is asked to specify a *search* first, to a-priori reduce the amount of relevant information, and is given search results with a *context* in the form of relevant neighbourhood. The user can explore the data further by *expanding* the graph into regions of interest. For selecting the context, the authors generalize *degree of interest* (DOI) [Fur86] from trees to general graphs and apply the method to a legal citation network. Other systems that implement similar approaches are **Apolo** [CKHF11] and **SaNDVis** [PGU*11]. However, both focus on network exploration, with user interaction not incorporated beyond decisions regarding search and expansion. Recently, a modular DOI specification has been presented, which allows the user to adjust the DOI function [AHSS13]. It uses only node characteristics and requires high level of user expertise.

### 2.2. Relation Extraction

Facts about the world, including the relationships between people and organizations, are captured in *ontologies*. If not compiled manually, data for ontologies is typically obtained automatically by mining and classifying named entities and their relationships from natural language text [Bie05, Sar08, NNSS13]. *Named entity recognition* (NER) is a classification problem that has been studied for a wide variety of languages, domains, and types of entities. Most interestingly for our use case, the German language, the journalistic domain, and the entity types “Person” and “Organization” are among the topics studied best in their respective areas of research. *Relation extraction*, especially the discovery and tagging of *non-taxonomic relationships*, is considered as one of the most difficult problems in ontology learning [KS05], because it is an *open information extraction* problem [BCS*07], in that the amount, types, and names of relations are not predefined. As a result, research has taken several paths, addressing specific variations of the problem [SM08], e. g. by fixing a set of non-taxonomic, but predefined relationships, or limiting itself to finding domain-specific relations between fixed entities. Other work systematically seeks unknown non-taxonomic relations, but relies on a posterior manual tagging by an analyst. This requires expertise that is not available in our scenario, as we assume users without specific knowledge in language technology.

Our work is also related to finding *lexico-syntactic patterns*. This has originally been performed manually, typically for hyponym relations [Hea92, SJN05]. Later, semi-automatic approaches for other kinds of relations, such as causation [GM02], were developed. This includes the fabrication of common and generalized grammatical constructions like "<NP$_1$> such as <NP$_2$> and <NP$_3$>" (where NP stands for "noun phrase"). The pattern implies a hyponym relationship between <NP$_1$> and <NP$_2$>, and between <NP$_1$> and <NP$_3$>. Since manual creation of such patterns is tedious, automatic methods have also been suggested [Sar08]. However, these approaches rely on predefined and possibly also pre-named relationships and thus cannot be applied to our use case. While manual pattern creation is more flexible, it is unsuitable for non-expert users, since it requires knowledge about syntax and lexico-syntactic patterns as well as reasoning about pattern quality.

## 3. Approach

Our system Networks of Names can be divided into two main parts: the *preprocessor* and the *visual interactive system*. The visual interactive system consists of three components: the *interactive visualization* (the system's frontend) as well as the *server* and *classifier* (the system's backend).

The system workflow is depicted in Figure 2: It extracts a social network from a natural language text corpus prior to the operation of the visual interactive system (1) (see Section 3.1 for details).

The visual interactive systems allows for exploration of the extracted network and tagging of the relationships based on the sentences from the underlying corpus (see Figure 2). Using the query dialogue of the visualization, users define one or two focal entities (2a). A subgraph containing the users' search terms is determined by the server (3). Users can then explore the graph visually by expanding or removing nodes (2b). Clicking on a link brings up the *sources view* (2c), which shows the sentences the relationship was extracted from. In this view, users can create tags for the relationship. Tag labels appear on the respective link in the visualization. At the same time, user-created tags are used as input for training a classifier, which subsequently generates tags for other similar relationships automatically (4). Details on the components of the visualization, server, and classifier are found in Sections 3.2, 3.3, and 3.4, respectively.

### 3.1. Preprocessor: Network Extraction

The preprocessor mines a social network from natural language corpora. We focus on German newspaper articles and use corpora from the "Leipzig Corpora Collection" (LCC) [QRB06]. The LCC compiles roughly 70 millions

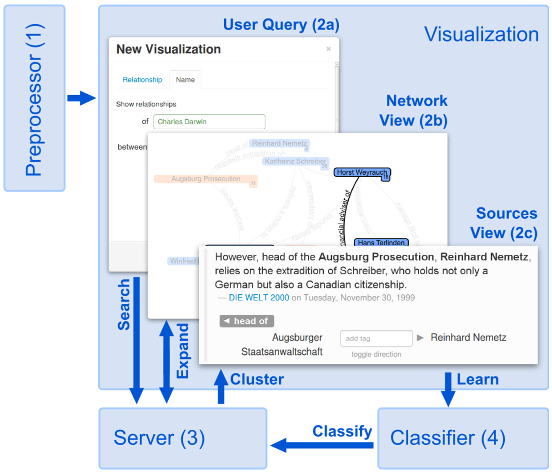



**Figure 2:** *Networks of Names: The preprocessor extracts a network from the corpus (1). Users search (2a) for an interesting subgraph that contains the query and expand on demand (2b). Search and expansion is handled by the server (3). Users tag source sentences (2c). User-created tags are used for classifier training and application (4).*

sentences from German online newspaper publications covering the timeframe from 1995 to 2010. Although we use only one language and only a single data source in the scope of this work, this is no design limitation and the data could be extended in the future.

Vertices of the network represent either people or organizations. Edges indicate binary relationships. To recognize the names of people and organizations and their type (person or organization) we employ the *Stanford Names Entity Recognizer* (Stanford NER) [FGM05]. Specifically, we use the *German NER* [FP10]. We aggregate and count the number of occurrences of each entity.

For relationship extraction, we regard two entities to share a connection if they appear together in a sentence, i. e. *co-occur*. Like with entities, we aggregate and count their occurrences. We follow this approach for two reasons:

1. We take a user-driven approach where classification follows user decisions. Therefore, we do not focus on relationship classification during network extraction.
2. Most methods for relationship extraction classify relationships into a set of predefined relations. We overcome this limitation by allowing for arbitrary types of relationships, defined by a user-created *folksonomy*.

To ameliorate the quality of output, we perform data preprocessing and cleaning: Prior to the extraction, we search/replace "corrupt" symbols and remove sentences that contain non-recognisable characters. This randomly samples the dataset without any substantial influence on the size or expressiveness of the original sample. After extraction, we remove entities that occur less than twice, have very short or generic names, or contain special characters. We remove news agencies, because their occurrence in sentences usually denotes the origin of the news rather than a relationship. Lastly, we whitelist and blacklist a small number of entities that appear frequently, but are commonly misclassified.

The resulting network has 47,939 vertices and 184,053 edges, an average vertex degree of 7.68 and a network clustering coefficient of 0.46. The diameter is 20 and the average shortest path length is 4.56. The power-law exponent is 2.51 (according to [CSN09]). The network shows properties of a *scale-free* and *small-world network*. Scale-free networks contain *hubs*, high-degree nodes that heavily contribute to graph connectivity. In our network, hubs are usually celebrities and politicians, multi-national corporations, famous sports clubs, or political parties. In small-world networks, paths between any two vertices are very short and the diameter is small regardless overall network size. It follows that there usually exists a path between any two entities of the network and the shortest path between them is short.

Since we use a German corpus, the system displays sentences in German and users create German tags during the user study. For better understandability by international readers, all text examples and figures in this paper have been manually translated into English.

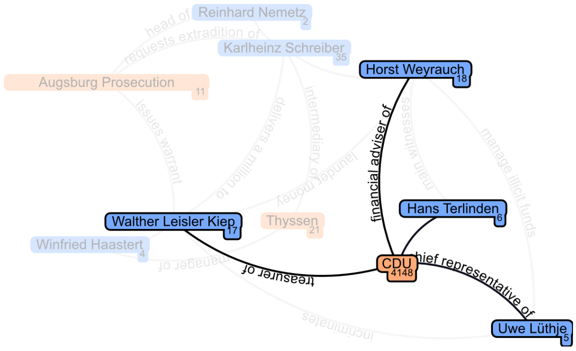



**Figure 3:** *Visual aid for graph exploration: Hovering a node highlights incident edges and adjacent nodes.*

### 3.2. Visualization: Exploring and Tagging Relationships

The interface of Networks of Names is designed for users with no special background knowledge. Thus, it does not contain exhaustive possibilities of parametrisation, but focuses on decisions related to the exploration and analysis process itself.

The exploration process starts by a search for one or two entity names. The network view then displays the respective nodes and their context within the network. The user can explore the network further by expanding or removing nodes (for details on search and expansion, see Section 3.2.1). Clicking on a link in the graph displays the sources view (see Figure 6), which allows the user to view and tag the underlying news sentences, which were used for extracting the relationship represented by the link (see Section 3.2.2).

#### 3.2.1. Network View: Exploring the Network

We visualize the network as a *node-link diagram*, since it is believed to be a particularly intuitive representation of graphs [GFC04] and allows the users to follow paths. Example visualizations can be seen in Figures 1 and 3. A node of the graph represents a person (blue) or an organization (orange). Entity names are depicted as a label within the node. Since not all nodes are shown (see Section 3.3.1), the number of hidden neighbours is indicated in the right bottom corner of each node. A link represents a relationship between two nodes. If the relationship is tagged, the label appears as text on the link.

We use a force-directed layout algorithm for graph drawing. We improve the user's ability to quickly perceive the information drawn onto the screen by initially positioning the same nodes at the same place. For that, we hash entity names onto $(x,y)$ coordinates. This results in similar searches arriving at similar layouts.

Once the search result is displayed, the user has several possibilities to adjust the layout and the view: panning and zooming, dragging and dropping nodes, and hovering nodes and edges to highlight incident edges and/or adjacent nodes (see Figure 3).

The user may also remove nodes and links or display additional nodes. The latter is achieved by expanding more neighbours of on-screen nodes using our extended DOI function (see Section 3.3).

#### 3.2.2. Sources View: Viewing and Tagging Sources

The user can open the sources view by clicking a link in the network view (see Figure 6). The sources view shows the sentences used by the preprocessor to extract the respective relationship. While the existence of a link indicates the existence of a relationship, viewing the sources allows the user to gain an understanding of the relationship's semantics.

Users can add tags to source sentences in order to characterise the relationship's semantics. The user enters the tag into the tagging widget that accompanies every sentence (see Figure 6). For the user, this serves the purpose of making the tag appear as a label of the link in the displayed graph.

In order to ensure diversity of sentences, we aggregate the sentences in clusters and only show a few representants per cluster (see Section 3.3.2 for details).

### 3.3. Server: Computations

The server answers user queries that require calculations on the (complete) network structure or are computationally expensive. This is used for searching entities and returning an "interesting" subgraph to the network view (see Section 3.3.1). If the user requests to see sources for a relationship, the respective sentences are clustered by similarity to reduce the textual load on users (see Section 3.3.2).

#### 3.3.1. Expansion: Extracting Interesting Contexts

The user initiates exploration by entering an entity name. Our aim is to return the corresponding vertex (called the *focal vertex*) and a context – its "interesting" neighbourhood. More formally, given a focal vertex $y$ from a large graph $G$, we extract a connected subgraph $F$ of $G$ that contains $y$, has some predefined size $n$, and maximal total "interestingness".

We quantify the interestingness of a subgraph by using a degree of interest (DOI) function. The seminal node-based definition of DOI [vHP09] defines vertex interestingness by a combination of an a-priori vertex interestingness (in our case the entity's frequency in the corpus), the distance to the focal node, and possibly additional user criteria. Based on DOI, an algorithm can select an interesting neighbourhood of the focal vertex. However, we find that this node-based definition of DOI uniformly produces the same "globally interesting" results, regardless of the query term (see Figure 4). This is most likely caused by the scale-free and small-world properties described in Section 3.1, as high-frequency nodes drive the expansion away from the original search.

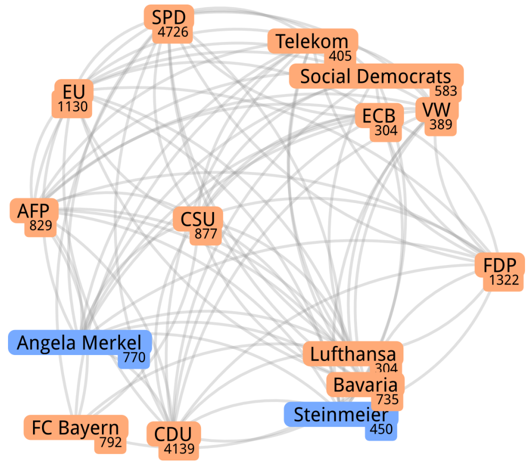
To counteract this problem, we propose an alternative DOI function that operates on edges instead of nodes. We define $\mathrm{DOI}_{\mathrm{edge}}$ in Equation 1, where $\{u,v\}$ is the edge of which the DOI is to be evaluated and $y$ is the focal vertex; API defines an *a-priori interest* of the edge and D the *distance* of the edge from the focal vertex; the factors $\alpha$ and $\gamma$ can be used to weight the components. Using our measure, we obtain the result seen in Figure 5.

$$\begin{aligned}\mathrm{DOI}_{\mathrm{edge}}(\{u,v\}|y,z) = \alpha \cdot \mathrm{API}_{\mathrm{edge}}(\{u,v\}) \\ + \gamma \cdot \mathrm{D}_{\mathrm{edge}}(\{u,v\},y)\end{aligned} \tag{1}$$

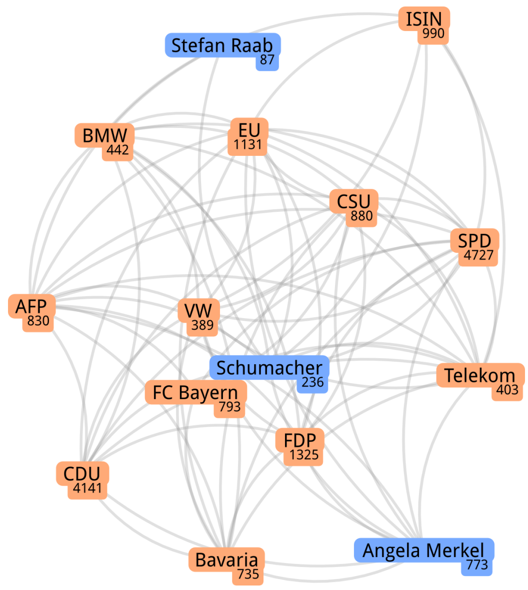
This variant allows us to employ a larger variety of measures to quantify a-priori interestingness. Inspired by information theory, we express $\mathrm{API}_{\mathrm{edge}}$ by *normalized pointwise mutual information* (NPMI) [Bou09], a significance measure for co-occurrence. We define *positive NPMI* by scaling its interval from $[-1,1]$ to $[0,1]$. Hence, applying positive NPMI to an edge $\{u,v\}$ yields 1 if and only if $u$ and $v$ always appear together and 0 if $u$ and $v$ never co-occur.

We define the elements of $\mathrm{DOI}_{\mathrm{edge}}$ as follows:

$$\mathrm{API}_{\mathrm{edge}}(\{u,v\}) = \mathrm{npmi}^{+}(\{u,v\})$$
$$\mathrm{D}_{\mathrm{edge}}(\{u,v\},y) = -(0.5^{\mathrm{d}(\{u,v\},y)} \cdot \mathrm{npmi}^{+}(\{u,v\}))$$




(a) Search for Angela Merkel with seminal node-based DOI




(b) Search for Stefan Raab with seminal node-based DOI

**Figure 4:** *Search and expand using a DOI measure based on node frequency. Two very different searches, Angela Merkel (the chancellor of Germany) and Stefan Raab (an entertainer and TV host) have very similar results due to a strong bias towards high-frequency nodes.*

where $\mathrm{npmi}^{+}(\{u,v\})$ is our positive NPMI measure and $\mathrm{d}(\{u,v\},y)$ denotes the shortest path distance of the edge $\{u,v\}$ to the focal vertex $y$ (meaning $\min(\mathrm{d}(u,y),\mathrm{d}(v,y))$). The definition of $\mathrm{D}_{\mathrm{edge}}$ is defined to discount the value of $\mathrm{API}_{\mathrm{edge}}$ based on distance to the focal node (discounting by absolute values is not possible for NPMI). We weight the components equally by setting $\alpha = \gamma = 1$.

We extract the subgraph as follows: The set of selected

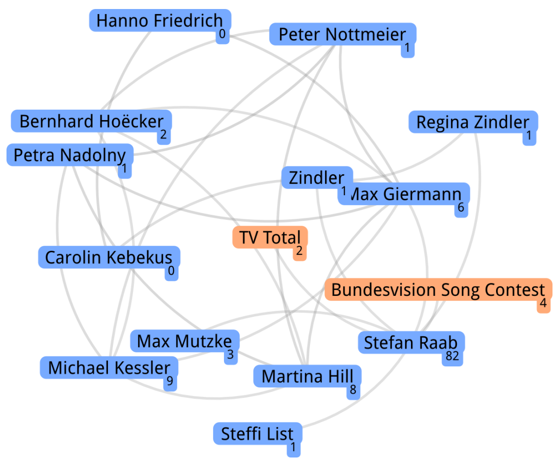



**Figure 5:** *Search for Stefan Raab using our edge-based DOI measure. The result shows a neighbourhood containing vertices that are closely related to the focal node (instead of expanding globally interesting nodes such as political parties and politicians, as seen in Figure 4 (b)).*

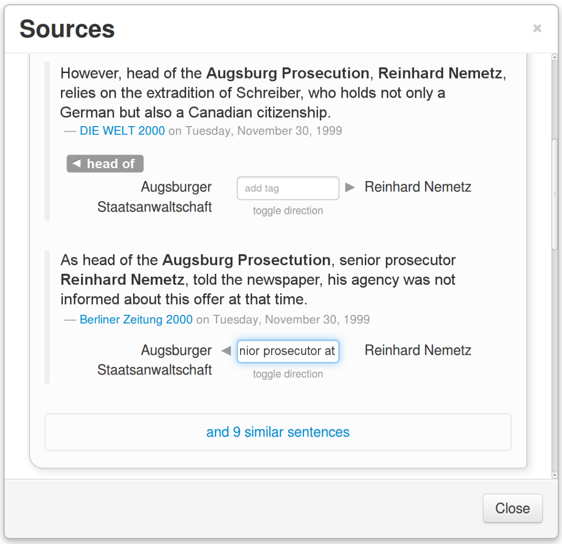



**Figure 6:** *The sources view shows source sentences for a relationship. Sentences appear grouped by similarity and can be tagged by users. Tag labels appear on the corresponding graph edge. User-created tags are utilized to train an automatic relationship classifier.*

vertices $S$ initially contains the focal vertex $y$. We maintain a priority queue of candidate edges $Q$, ordered by $\mathrm{DOI}_{\mathrm{edge}}$. Initially, $Q$ is filled with all edges incident to $y$. In every step, the algorithm pops the edge with highest DOI from $Q$ and adds the endpoint $v$ that is not yet in $S$ to $S$. All incident edges of $v$ are added to $Q$. This step is repeated until $S$ reaches the desired size $n$ or until no candidates are left in $Q$. The result is the subgraph induced by all vertices in $S$.

In addition to the search for one entity, we allow the user to enter two focal entities. In this case, we start by searching a path that connects the two entities. Since shortest paths have limited expressiveness in small-world graphs [HMLH10], we construct paths of maximum interestingness instead. We obtain such paths by calculating a (preferably short) *maximum-capacity path* [Pun91, MC02], i. e. a path where the capacity of the minimum-capacity edge is maximal. For edge capacity, we use the a-priory interest $\mathrm{API}_{\mathrm{edge}}$ rounded to two decimal digits (to prevent small differences in capacity having an impact). An interesting subgraph is extracted with the algorithm for one focal node, but with $S$ initialized to all vertices on the max-capacity path.

Regardless the definition of DOI, low-quality expansion can be caused by the presence of hubs (i. e. high-frequency nodes such as political parties). Hubs are frequent in our network due to the scale-free property (see Section 3.1). Expanding a hub introduces a large number of candidates for further expansion (e. g. the members of the party), which are likely to include relationships of high DOI. This favours the expansion of vertices that are only marginally related to the original search. Preferring direct connections to connections via hubs instead, we do not add the neighbours of hubs as candidates for expansion (they may still be expanded as neighbours of other nodes or manually by user interaction).

#### 3.3.2. Clustering Sentences

In the sources view, sentences are displayed clustered by similarity. We determine clusters using the *Markov Clustering Algorithm* [vD00], comparing sentences by *cosine similarity* of their vector representation based on tf-idf [MRS08], a measure that gives the importance of a term in a document collection by relating the number of times a word appears in a single document to the frequency of the word in the corpus.

We display three representative sentences from each cluster: First, the earliest source for the cluster. Second, a sentence in which the two entities appear close to each other. Third, a sentence with an automatic tag that was not yet accepted or rejected by a user (see Section 3.4). All other sentences from the cluster are hidden and can be shown on demand (see Figure 6).

### 3.4. Classifier: User-driven Automatic Tagging

During network exploration, users produce tags that describe a semantic relationship between two entities. Thereby, users produce three sorts of interesting data:

1. They signal that a semantic relationship exists between two entities.
2. They identify a sentence that contains or implies the semantic relationship between them.

3. They name the relationship by entering a label.

We implement a classifier that exploits this data. The classifier is trained and applied during the system's operation. Initially, training data is scarce. Thus, the classifier must work with little supervision and almost no feedback on its decisions.

According to Sarawagi's taxonomy [Sar08], our problem is similar to the case where "given one or more relationship types, [...] our goal is to find all occurrences of those relationships". Our case is slightly different in that we obtain not only an entity pair and a relationship label, but also the accompanying source sentence. Using this, we build our classifier to extract lexico-syntactic patterns directly from that sentences. We propose a high-precision low-recall approach: The pattern is the shortest phrase from the sentence that contains both entity names and possibly a non-stopword keyword from the label. Both phrase and label are generalized by substituting names and keyword by placeholders. For instance, given the sentence

> *However, head of the Augsburg Prosecution, Reinhard Nemetz, relies on the extradition of Schreiber, who holds not only a German but also a Canadian citizenship.*

and the user-entered label "head of" for the relationship between *Reinhard Nemetz* and the *Augsburg Prosecution*, the system derives the pattern "<W> of the <O>, <P>", where <W>, <P>, and <O> are respective placeholders for the keyword, person, and organization (the system also encodes tag direction, but we omit this detail for brevity). Analogously to the pattern, the label is generalized to "<W> of", i. e. replacing the only non-stopword keyword "head" by a placeholder for any expression.

If a pattern is extracted, the classifier proceeds by applying the newly learned pattern to the whole corpus. For that, it instantiates the generalized label to new tags for every match by replacing the placeholder <W> by the expression that appears in the matched sentence. For instance, the classifier would tag a sentence that contains the phrase "*treasurer of the CDU, Walter Leisler Kiep*" by instantiating the above label to "treasurer of" for the relationship between the *CDU* and *Kiep*.

Automatic tags that were already user-created are instantly accepted as correct. Other automatic tags can be manually accepted or rejected by users. In addition, the system assumes a user to accept a tag if he selects its label to appear on the corresponding link or if creates another tag with the same label for the same relationship. Using the number of accepted and rejected applications of pattern $i$, it is possible to calculate the pattern's *precision* [MRS08] $p(i)$, which denotes the fraction of correct applications on a $[0,1]$ scale:

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

With regard to the metric's meaning, two factors should be considered: First, since the calculation is based only on the number of applications that were accepted or rejected, it is important that the sample be sufficiently large and representative. Second, due to lack of a gold standard, the precision is a result of posterior evaluation (i. e. evaluation that includes user decisions) and is thus susceptible to user bias.

## 4. Evaluation and Results

In order to evaluate the capabilities of our visual interactive system, we conducted a user study (Section 4.1). Based on data from the study, we also evaluated our classifier (Section 4.2).

### 4.1. User Study

The study aim was threefold:

1. We wanted to test the system with users that have no special knowledge on networks, network visualization, or language technology, and no knowledge about the 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 authentic data needed to train, apply and evaluate our classifier.
3. We wanted to analyse whether our visual interactive system impacts how users create tags, as opposed to a comparable text-based tool.

#### 4.1.1. User Study Setup

We conducted the user study with 26 participants using a between-group design: The users were split into two groups of 13 participants each. Users from the first group were asked to use the visual interactive system (the visual study), users from the second group were confronted with a text-based system (the text study). The text-based system corresponds to the sources view of our visual interactive system, but lacks the graph visualization and all possibilities of interaction related to graph exploration. Instead, it successively opens sources views for a number of relationships.

Participants of the visual study were asked explore one of the example searches, conduct at least one search of their own choice, and to accept or reject automatic labels, should they encounter any. 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 graph exploration, they could navigate the graph freely and choose what they explore and present. Users could freely open the sources view, tag sentences, as well as decide how many tags they produce and how they word tag labels. Users could determine how much time they spend with the system.

In the text study, users were presented the sources view for relationships that directly corresponded to relationships

| | 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 1:** *Metrics on user interaction per user for the visual versus the text study.*

| | Visual | Textual |
|---|---|---|
| Number of manual labels | 171 | 877 |
| Average number of words | 1.53 | 2.67 |

**Table 2:** *Metrics on tag labels created by users in the visual versus the textual study. Users of a textual system create considerably more and longer labels.*

viewed by a participant of the visual study (with duplicates removed). Using the sources view only, participants were asked to characterize relationships displayed to them as a whole by tagging relevant sentences, and to accept or reject automatic labels, should they encounter any. Apart from the lack of visual graph exploration, their degrees of freedoms were similar: Users could 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 could choose which sentences they tag, how many tags they produce, and how they word tag labels.

Although a study of larger scale was not within the scope of this work, we attempted to mitigate group difference effects: All participants were between the ages of 23 and 34. Most participants did not have specific knowledge about language technology, networks, or network visualization. Both groups had few participants with basic knowledge of computer science and the aforementioned fields. Both groups were given an introduction on how to operate the system.

#### 4.1.2. User Study Results

We recorded the number of manual and automatic tags after each test. As expected, the amount of automatically generated tags greatly outnumbers manual tags (by a factor of 143 in both studies). While the number of manual tags grows continuously between participants, the number of auto-tags is subject to regular steep jumps. Such jumps occur when users tag sentences that happen to be well-suited for pattern generation and the resulting pattern is sufficiently common in the corpus to facilitate frequent application.

Table 1 shows per-user metrics for both studies. In the text study, users created 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 higher in the text study. We assume that this difference can conclusively be attributed to the circumstance that in the text study, users spent their time exclusively in the part of the system where tags can be created and were given the explicit analytical task of creating tags.

The exact wording of labels was free user's choice in both groups. Table 2 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 with respect to the number of labels is even larger than the difference with respect to the number of tags. Not only the number of different labels, but also the number of words per label is higher in the text study. By direct comparison, the two sets of labels share only 30 labels. To explore the similarity of the sets, we consider labels similar if they are phrases of each other (e. g. “head of” and “is head of”), differ in grammatical gender, number, or case, are modified by temporal markers (e. g. “former”, “current”, or “ex-”), or have suffixes that signal membership or presidency. Using this notion, 73 and 112 labels have similar counterparts it the other set, respectively. This corresponds to 42.69% of labels from the visual study, but only 12.77% of labels from the text study.

This discrepancy initiated detailed analysis of the labels in the text study. We found that several labels are considerably longer than labels from the visual study. They express a reasonable relationship, but are very verbose (“is current member of the supervisory board just like”). Other labels describe very high-level relationships (“women”) or rephrase quotes from the respective sentences (“authors of text about the complicated ecosystem called forest”).

We assume that the visualization has an impact on this result for two reasons: First, with the context of a larger graph visualization in mind, concise labels are easier to imagine as edge labels. Second, the examples suggest that users of the text-based system may have been concerned with less relevant details of sentences, resulting in more under-specific and over-specific labels.

These observations suggest that the presence of a visualization has a regulating effect on the emerging folksonomy.

### 4.2. Classifier Evaluation

The user study provided authentic classifier data (i. e. data by users not involved in the design and development of Networks of Names). However, this data is too sparse to be used for the evaluation of classifier performance. In addition, the data is biased towards certain regions of the graph (e. g. those provided as example searches), and thus is distributed neither evenly, nor randomly. As a consequence, we evaluate the classifier manually and in a more controlled environment.

#### 4.2.1. Classifier Evaluation Setup

We used the dataset obtained from the text study for its larger size. We note that differences in tagging behaviour between visual and text study do not directly affect classifier performance for two reasons: First, training and application is performed triggered by user input, but without user interaction or knowledge about how the process works in detail. Second, tagging of peculiar sentences typically results in patterns that can be applied only to the sentence they are extracted from. Excluding all patterns that were applied less than five times, 133 patterns were left for evaluation.

In addition to accepting or rejecting tags, we annotated tags to be "close" if they resulted from proper application, but were oddly worded or contained grammatical mistakes. Randomly sampling 1000 tags created by the 133 patterns, we assigned them manually into the categories "accepted", "rejected", "close", or "undecided". From the annotations, we derived a precision for the classifier. The value is a posterior precision, since it is based on user decisions (instead of a reference ontology). We attempted to validate the data objectively, but note that as discussed previously in Section 3.4, posterior values can be subject to user bias.

#### 4.2.2. Classifier Performance

The classifier's accuracy is 53% for accepted tags only, and 61% including "close" matches. This precision is reasonable compared to other approaches that use lexico-syntactic patterns [SJN05], especially given our straightforward method of pattern extraction.

Several high-precision patterns emerge from the validation. Such patterns cover many lexico-syntactic patterns usually stated and used in literature [Hea92, SJN05, BCS*07, OT10]. For instance, the most frequent patterns are patterns such as "<W> of <O>, <P>" or "<O>-<W> <P>" that describe positions of people in organizations (with <W> substituted by terms like "chairman" or "expert").

Problems with patterns of moderate or low precision stem from our approach of pattern generation and application, which does not take into account part of speech, sentence structure, and grammatical dependencies. Furthermore, the precision of the classifier is given as an average of individual pattern precisions. Since our classifier is trained, applied, and evaluated during operation of the system, individual patterns could be identified and their application revoked once the system has enough evidence to conclude that their precision is low, leading to eventual self-improvement.

The advantage of our approach is that pattern are extracted without explicit construction or evaluation of the pattern by the user. Especially, no knowledge about morpho-syntactic details or reasoning about the potential feasibility and quality of patterns is required. Since our approach does not rely on any language-specific methods, it is directly applicable to other languages.

## 5. Conclusion and Future Work

We designed and developed Networks of Names, a system that is capable of extracting relationship information from large text corpora and making it available for visual exploration and interactive tagging. The system incorporates and enhances current research from visual analytics and integrates methods from language technology and other fields of computer science.

Within the scope of this work, we focused on the extraction of people and organizations as entities. Working with the system reveals that performing disambiguation and normalizing names would look more natural and contribute to the quality of the dataset as well as the usability of the system. Another useful type of entity in the context of relationships between people and organizations are events. Especially for abstract events (such as a financial crisis), the addition would require significant work in both research and implementation. Furthermore, visualizing the time dimension could enable users to not only explore relationships between entities, but also their change over time.

We presented a new edge-based *DOI* function, utilizing *normalized pointwise mutual information* to express the interestingness of edges. We found it to work substantially better for our scenario than the seminal version of DOI [vHP09]. Additionally, we extended the original expansion algorithm to work with not only one, but two focal vertices (using *maximum-capacity paths* for calculating interesting connections between vertices).

We conducted a user study to test the system, generate data for the classifier, and explore the impact of a visualization on the tagging behaviour of users. We found that users of the visual interactive system tagged more concisely and refrained from tagging extremely abstract or over-specific relationships. This allows the assumption that the presence of a visualization may have a regulating effect on the emerging folksonomy.

Evaluating the performance of our classifier, we found that our semi-automatic approach has reasonably high precision. Without being predefined or manually crafted, our classifier produced high-precision patterns that are known from research literature. The classifier could be improved in the future by implementing more sophisticated methods for relationship extraction. For this, part of speech, syntactic parse trees, or dependency trees could be considered [Sar08].

#### Acknowledgements and Companion Website

This work has been partially supported by DFG within a project in SPP Visual Analytics (SPP 1335), and by LOEWE as part of the research center *Digital Humanities*.

A companion website with source code and demo installation is located at `http://maggie.lt.informatik.tu-darmstadt.de/thesis/master/NetworksOfNames/`.

## References


[AHSS13] Abello J., Hadlak S., Schumann H., Schulz H.-J.: A Modular Degree-of-Interest Specification for the Visual Analysis of Large Dynamic Networks. *IEEE Transactions on Visualization and Computer Graphics* (2013). 2

[BCS*07] Banko M., Cafarella M. J., Soderland S., Broadhead M., Etzioni O.: Open Information Extraction from the Web. In *IJCAI* (Hyderabad, 2007), pp. 2670–2676. 2, 9

[Bie05] Biemann C.: Ontology Learning from Text: A Survey of Methods. *LDV Forum 20*, 2 (2005), 75–93. 2

[Bou09] Bouma G.: Normalized (Pointwise) Mutual Information in Collocation Extraction. In *GSCL* (Potsdam, 2009), pp. 31–40. 5

[Bun11] Bundesverband Deutscher Zeitungsverleger e.V.: Die deutschen Zeitungen in Zahlen und Daten, 2011. 1

[CKHF11] Chau D. H., Kittur A., Hong J. I., Faloutsos C.: Apolo: Making Sense of Large Network Data by Combining Rich User Interaction and Machine Learning. In *CHI* (Vancouver, 2011), pp. 167–176. 2

[CSN09] Clauset A., Shalizi C. R., Newman M. E.: Power-law Distributions in Empirical Data. *SIAM Review 51*, 4 (2009), 661–703. 4

[FGM05] Finkel J. R., Grenager T., Manning C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *ACL* (Ann Arbor, 2005), pp. 363–370. 3

[FP10] Faruqui M., Padó S.: Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *KONVENS* (Saarbrücken, 2010). 3

[Fur86] Furnas G. W.: Generalized Fisheye Views. In *CHI* (Boston, 1986), pp. 16–23. 2

[GFC04] Ghoniem M., Fekete J.-D., Castagliola P.: A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations. In *InfoVis* (Austin, 2004), pp. 17–24. 4

[GM02] Girju R., Moldovan D. I.: Text Mining for Causal Relations. In *FLAIRS* (Pensacola, 2002), pp. 360–364. 3

[GM04] Grobelnik M., Mladenić D.: Visualization of News Articles. *Informatica 28*, 4 (2004). 2

[Hea92] Hearst M. A.: Automatic Acquisition of Hyponyms from Large Text Corpora. In *COLING* (Nantes, 1992), pp. 539–545. 3, 9

[HMLH10] Hangal S., MacLean D., Lam M. S., Heer J.: All Friends are Not Equal: Using Weights in Social Graphs to Improve Search. In *SNAKDD* (Washington, D.C., 2010). 6

[KMSZ06] Keim D. A., Mansmann F., Schneidewind J., Ziegler H.: Challenges in Visual Data Analysis. In *IV* (London, 2006), pp. 9–16. 1, 2

[KS05] Kavalec M., Svaték V.: A Study on Automated Relation Labelling in Ontology Learning. In *Ontology Learning from Text: Methods, Evaluation and Applications*, Buitelaar P., Cimiano P., Magnini B., (Eds.), Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, 2005, pp. 44–58. 1, 2

[MC02] Malpani N., Chen J.: A Note on Practical Construction of Maximum Bandwidth Paths. *Information Processing Letters 83*, 3 (2002), 175–180. 6

[MRS08] Manning C. D., Raghavan P., Schütze H.: *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, 2008. 6, 7

[NNSS13] Nastase V., Nakov P., Séaghdha D. Ó., Szpakowicz S.: Semantic relations between nominals. *Synthesis Lectures on Human Language Technologies 6*, 1 (2013), 1–119. 2

[OT10] Ohshima H., Tanaka K.: High-speed Detection of Ontological Knowledge and Bi-directional Lexico-Syntactic Patterns from the Web. *JSW 5*, 2 (2010), 195–205. 9

[PGU*11] Perer A., Guy I., Uziel E., Ronen I., Jacovi M.: Visual Social Network Analytics for Relationship Discovery in the Enterprise. In *VAST* (Providence, 2011), pp. 71–79. 2

[Pun91] Punnen A. P.: A Linear Time Algorithm for the Maximum Capacity Path Problem. *European Journal of Operational Research 53*, 3 (1991), 402–404. 6

[QRB06] Quasthoff U., Richter M., Biemann C.: Corpus Portal for Search in Monolingual Corpora. In *LREC* (Genoa, 2006), pp. 1799–1802. 3

[Sar08] Sarawagi S.: Information Extraction. *Foundations and Trends in Databases 1*, 3 (2008), 261–377. 2, 3, 7, 9

[SJN05] Snow R., Jurafsky D., Ng A. Y.: Learning Syntactic Patterns for Automatic Hypernym Discovery. In *Advances in Neural Information Processing Systems 17*, Saul L. K., Weiss Y., Bottou L., (Eds.). MIT Press, Cambridge, 2005, pp. 1297–1304. 3, 9

[SM08] Sánchez D., Moreno A.: Learning Non-taxonomic Relationships from Web Documents for Domain Ontology Construction. *Data & Knowledge Engineering 64*, 3 (2008), 600–623. 2

[Smi00] Smith R.: 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. 1

[vD00] van Dongen S. M.: *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht, 2000. 6

[vHP09] van Ham F., Perer A.: "Search, Show Context, Expand on Demand": Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 953–960. 2, 5, 9

[vHWV09] van Ham F., Wattenberg M., Viégas F. B.: Mapping Text with Phrase Nets. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1169–1176. 2