Browsing Publication Data using Tag Clouds over Concept Lattices Constructed by Key-Phrase Extraction

Gillian J. Greene, Marcel Dunaiski, Bernd Fischer
Computer Science Division
Stellenbosch University, South Africa
ggreene@cs.sun.ac.za, marcel@ml.sun.ac.za, bfischer@cs.sun.ac.za
and
Dmitry Ilvovsky, Sergei O. Kuznetsov
School of Data Analysis and Artificial Intelligence
National Research University Higher School of Economics, Russia
{dilvovsky, skuznetsov}@hse.ru

Abstract. In order to find research on a specific topic or to get an overview of the topics that are published at different academic venues, academics need to browse data from existing academic publications. The title and abstract of publications contains useful key-phrases indicating the topic of the publication, but these need to be directly extracted and presented in a browsable format in order to allow the user to find relevant publications. We extract key-phrases and use these to construct a concept lattice for a dataset of publications. We then present the information in an intuitive interactive tag cloud browser where navigation is supported by the underlying concept lattice.

1 Introduction

In order to find research on a specific topic or to get an overview of the topics that are common at different academic venues, academics need to browse data from existing academic publications. Publications are often associated with specific keywords, but often these are from a restricted vocabulary and thus may not be comprehensive. Relevant information for the publication is however provided as free text in the abstract and title which includes an overview of the work and may contain key-phrases that could be useful in characterizing the research. However, these key-phrases need to be extracted together from the free-text.

We use our ConceptCloud browser [16], which is based on a novel combination of concept lattices and tag clouds, to present key-phrases which we extract from academic publications, and so enable users to browse the publication data by selecting a combination of key-phrases. Our tag clouds allow users to navigate along different paths, and to aggregate the publication data in different ways.

Tag clouds are a simple visualization method for textual data where the importance of each tag (typically its frequency) is reflected in its size. Navigation using tag clouds has previously been explored using a Bayesian approach [29];

however, navigation in our browser is supported by a novel combination of tag clouds and concept lattices [37, 14, 8]. Concept lattices have been shown to be useful for browsing data [9, 25, 7] but large lattices do not provide a suitable data visualization because the relationships between the concepts are difficult to identify in a large Hasse diagram.

Our navigation algorithm maintains a *focus concept* in the underlying lattice. We derive the tag cloud visualization from the current focus concept and update it after each navigation step. Navigation is driven by the user's selection (or de-selection) of tags in the tag cloud.

2 Background

2.1 Key-Phrase Extraction

A key-phrase extraction system typically extracts a list of words or phrases that serve as candidate phrases using some heuristics [20] and then determines which of these candidate phrases are key-phrases using supervised or unsupervised learning approaches. Typical heuristics include using a stop word list to remove commonly occurring stop words [27], and using words with certain part-of-speech tags (e.g., nouns, adjectives, verbs) as the keywords [30]. N-grams [38] or noun phrases [5] that satisfy pre-defined lexico-syntactic patterns [31] can also be used as the candidate phrases. Since most of the approaches are restricted to the boundaries of the sentence key-phrases could be also extracted directly from the paragraph using specific representation called "parse thicket" [10, 13]. In that case extended phrases (including discourse relations) are used instead of regular ones [12].

Supervised learning approaches used to select key-phrases from the pool of candidate key-phrases make use of different types of features such as statistical, syntactic, structural or external resources. Unsupervised learning approaches to key-phrase identification typically make use of a graph-based [30] or clustering approach [18].

2.2 Formal Concept Analysis

Formal Concept Analysis (FCA) [37, 14, 8] uses lattice-theoretic methods to investigate abstract relations between objects and their attributes. Such *contexts* can be imagined as cross tables where the rows are objects and the columns are attributes. Note that we follow the definitions of [14].

Definition 1 *A formal* context *is a triple* (G, M, I) *where* G *and* M *are sets of objects and attributes, respectively, and* $I \subseteq G \times M$ *is an arbitrary incidence relation.*

Definition 2 *The common attributes of the objects in* A *are given as* $A' := \{m \in M \mid (g, m) \in I \text{ for all } g \in A\}$ *for* $A \subseteq G$. *The common objects of the attributes in* B *are given as* $B' := \{g \in G \mid (g, m) \in I \text{ for all } m \in B\}$ *for* $B \subseteq M$.

Definition 3 *A formal concept of the context* (G, M, I) *is a pair* (A, B) *with* $A \subseteq G$ *and* $B \subseteq M$ *such that* $A' = B$ *and* $B' = A$.

Under the ordering $(A_1, B_1) \leq (A_2, B_2) :\Leftrightarrow A_1 \subseteq A_2$ *the concepts from any formal context form a complete lattice which is called the* concept lattice.

Efficient algorithms exist for the computation of the concept lattices and the meet and join of concepts in the lattice (for example [26]).

2.3 Tag Clouds

Tag clouds are a common visualization of textual data. In Web 2.0 applications tag clouds are often built from user-generated tags for particular content. However tag clouds can also be generated directly from common words in text. Figure 1 shows a tag cloud generated from the content of Obama and Bush's inaugural speeches. The most commonly used words are presented in the largest font size. The tag cloud provides a simple overview of the speech content and enables comparison between the two speeches.

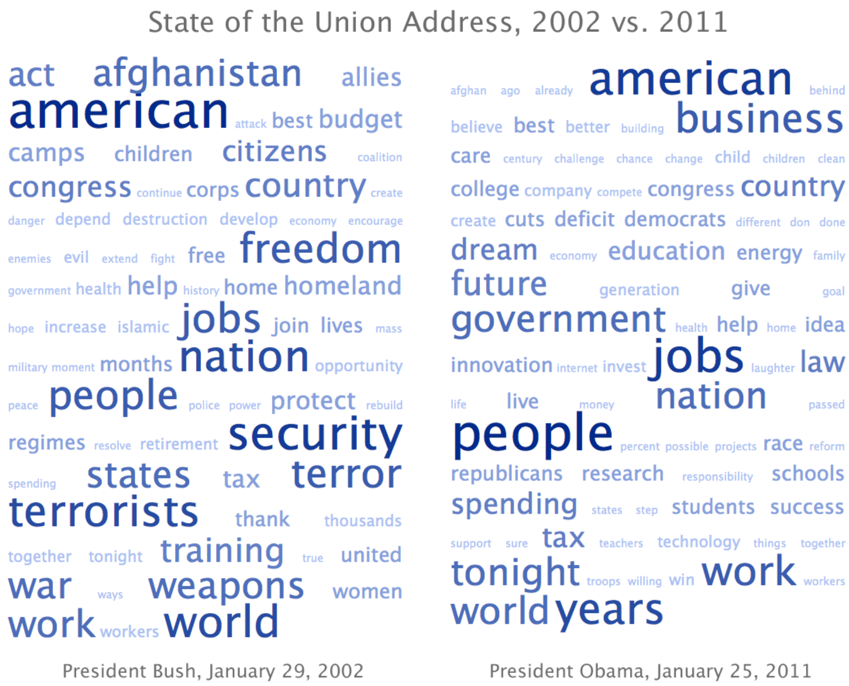

Fig. 1: Tag Cloud Generated from Obama and Bush's Inaugural Speeches [3]

3 Navigation Framework

3.1 Contexts from Publications

In order to generate a context from publication data we use the paper itself represented by its paper-id as the object in the context table and assign attributes from the paper's authors, year of publication and extracted key-phrases from the title and abstract.

Building the context in this way allows us to see papers that share keywords and also papers that share authors. Selecting an author tag will display in the tag cloud all attributes of their publications (including co-authors) and their tags will be sized according to how often the attributes occur.

3.2 Tag Clouds from Concepts

We generate tag clouds directly from concepts in a concept lattice instead of free text or user-generated tag information. The tag cloud provides a more intuitive interface to the information contained in our concept lattice.

Since a concept comprises a set of objects and a set of attributes, it is tempting to use the attributes (i.e., the intent) as the tag cloud. However, this produces degraded clouds because (*i*) the intent only contains the attributes common to all objects, and (*ii*) each attribute only occurs once so that all tags would have the same size. Instead, we use the intents of the extents; more precisely, we collect all attributes of the defining concept of each object in the extent of the focus concept; we also add the objects themselves, to allow their direct selection in the tag cloud.

Definition 4 *The* tag cloud *from a concept* $c = (A, B) \in B(\mathcal{C})$ *is defined as* $\tau(c) = A \uplus \biguplus_{a \in A} \pi_M \sigma(a)$ *where* $\pi_M(c)$ *refers to the concept c's intent and* $\sigma(a)$ *refers to the* defining concept *of object a.*

Here $\uplus$ denotes multiset union. By construction, the objects in the tag cloud induce subconcepts of the concept from which the tag cloud was derived; moreover, all tags have a non-bottom meet with that concept.

3.3 Navigating Concept Lattices with Tag Clouds

The browser maintains a *focus concept*, from which it renders the tag cloud as described above; when the user selects (or deselects) a tag, the browser updates the focus and re-renders the tag cloud. The focus, or more precisely, its extent contains the subset of objects (i.e, academic papers) that share all currently selected tags. The initial focus (corresponding to an empty selection set) is therefore the lattice's top element, whose extent contains the entire data-set.

Navigation is refinement-based: when the user selects another tag, the browser updates the focus by computing the meet of that tag's defining concept and the old focus, rather than recomputing it from the full selection set.

For example in Figure 2, we see the initial tag cloud generated from the top concept in the lattice and after selection of tags "model" and "checking" (Figure 4) the tag cloud is regenerated from an updated focus concept with "model" and "checking" contained in the focus's intent.

Intuitively, deselection should be the inverse of selection: deselecting the last selected tag should move the focus back to its previous position. Therefore for deselection we recompute the focus as the meet of the defining concepts of the remaining selected tags.

4 Key-Phrase Extraction

Our key-phrase extraction technique consists of two steps; we first extract candidate key-phrases and then we remove stop words from the collected phrases.

In order to reduce the key-phrases to only those including more technical information, we extract only noun phrases which do not include pronouns. For each single-word phrase we apply lemmatizing [2] and extract only one phrase for each group of words having the same lemmas.

We do the syntax parsing of the abstract based on Stanford Natural Language Processing (NLP) tool [2] which includes tokenization, sentence splitting, part-of-speech tagging, lemmatizing and parsing itself. We take all noun phrases with the length less than 5 as key-phrases. We therefore compute all subtrees with noun part-of-speech tags in the root that have less than 5 leaves for each syntax tree corresponding to the paper's abstract.

We then remove stop-words from the single-word phrases that we have extracted. The stop list includes common words that are used in research papers but are not domain-specific such as *paper* and *research.* According to our task we consider all long phrases as meaningful and do not remove them.

For example in the sentence "Software development is the process of computer programming, documenting, testing and bug fixing" our system will extract the following key-phrases: "software development", "process", "computer programming", "documenting", "testing", "bug fixing".

5 ConceptCloud Tool

We use our ConceptCloud browser [16] which is a web application available at `www.conceptcloud.org` in order to make the academic publication set browsable. ConceptCloud comprises two main components; a concept constructor tool to construct a context table in the desired format, and a tag cloud display to display the interactive tag cloud (see Section 5.2) of the resulting lattice.

5.1 Concept Constructor Tool

ConceptCloud's ConceptConstructor automates the process of creating a tag cloud visualization from an XML or JSON file and provides a wizard to allow users to construct the table with their desired combination of pre-processing steps. The browser is generic and can show tag clouds of different context types. It is also completely automatic: there are no manual pre-processing steps, and the user only needs upload the dataset, choose which of the pre-processing steps to apply and export the table with the desired objects and attributes. ConceptCloud's ConceptConstructor also allows users to export tables in a ".cxt" format so that they can also be used to generate a lattice diagram. A more detailed description of the tool architecture is available in [17].

For the lattice construction, we use a method based on the Colibri/Java library [15] which constructs concepts on the fly. We thus never need to compute the full lattice and are able to render an initial tag cloud relatively quickly.

5.2 Tag Cloud Visualization

We make use of a tag cloud visualization that can be customized to show different views on the publications. Multiple different visualizations for different metrics were found to confuse users [4]. We therefore propose one uniform visualization that can be used to explore various different aspects of a data archive.

The simplest and most popular tag cloud layout [28] is as an alphabetically sorted list of tags in a roughly rectangular shape which was found by Schrammel et al. to perform better than random or semantic layouts [34]; we use this layout because it simplifies textual search within the tag cloud. We scale each tag i between the given minimum and maximum font sizes f_{min} and f_{max}, according to its weight t_i in relation to the minimum and maximum weights in the context table, t_{min} and t_{max}; hence,

$$\text{size}(i) = \left\lceil \frac{(f_{max} - f_{min}) \cdot (t_i - t_{min})}{t_{max} - t_{min}} \right\rceil + f_{min} - 1$$

for $t_i > t_{min}$ and $\text{size}(i) = f_{min}$ otherwise.

A variety of alternative tag layout methods have been proposed, such as tag flakes by Caro et al. [6]. Tag flakes are used in order to provide context for tags as basic tag clouds fail to show how the tags are related. However, instead of using such complex visualization that depicts the relationships between the tags, we use incremental refinement in the tag cloud to provide context and structure to the tag clouds. By selecting a tag in the tag cloud the resulting cloud will provide background for the selected tag.

The initial tag cloud shown in ConceptCloud includes tags from all attributes and objects in the context table (using the top concept in the lattice as the focus). This allows the user to select any tag from the extracted publication data. Tags in the initial tag cloud will be at their largest size because we scale all tags according the maximum and minimum tags in this cloud. Making selections in the initial tag cloud will result in clouds with smaller tags, indicating that the cloud is only showing attribute tags from a subset of the total objects in the context table.

A tag is *implied* if it has not been selected explicitly, but corresponds to an attribute in the focus' intent. Implied tags thus reveal the dataset's internal structure, similar to the way association rules reveal the implicit structure of shopping baskets [39] but without any additional cost.

6 Illustrative Case Study

We build a tag cloud from data extracted from the proceedings of the Automated Software Engineering Conference [1]. This dataset comprises 1400 papers and contains their titles, abstracts, author information and some optional IEEE/ACM keywords for the papers.

Figure 2 shows the 200 most common words extracted from the abstracts and titles of 1400 publications appearing in the ASE [1] conference. We see keywords such as "automated", "development" and "results" but the overall themes of the conference are difficult to identify from the individual key-phrases.

In Figure 3, showing the 200 most common key-phrases extracted from the abstracts and titles of the same set of publications, we see the introduction of the key-phrases which better characterize the topic of the research. We see key-phrases such as "Model Checking", "Design Patterns" and "Formal Specifications" which are difficult to identify in the single words of Figure 2. Key-phrases thus better highlight the content of the conference proceedings. The key-phrase extraction has also removed verbs from the keywords and from Figure 2 we see that the verbs contain little information when compared to the nouns.

Selecting the tag for "Model Checking" (indicated in red) in Figure 5 and still showing the 200 most common tags, shows which authors commonly work on "Model Checking" at this conference and also what other keywords, such as "State Space" are associated with "Model Checking", sized according to how often they appear together. Using only single words from the abstract in the tag cloud would mean that phrases are not automatically visible in the tag cloud and have to be selected by selecting two individual tags. When the tags for "Model" and "Checking" are selected separately (see Figure 4) and not as a key-phrase they may also not appear together in the abstract and so may show papers of which "Model Checking" is not a topic. From Figure 4 where words in the title and abstract have only been split we can also see that identifying key-words related to "Model Checking" through the tag cloud is difficult because it is unclear which of the other keywords in the cloud are related to each other. For example, from this cloud we would not be able to see that both keywords "state" and "space" often occur together along with "model" and "checking" unless we were to select these as additional tags.

The addition of the key-phrase extraction allows users to refine the publication set to only papers referring to a particular subset of the domain. In addition when one key-phrase is selected the tag cloud shows which other phrases are commonly used together with the selected phrases in the same publication. This allows the user to investigate related key-phrases to a particular research topic.

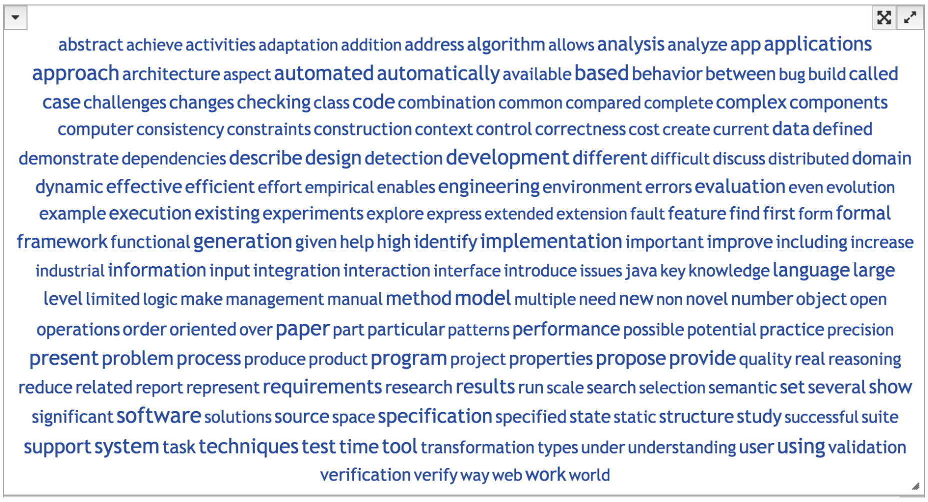

Fig. 2: Tag cloud constructed using only basic pre-processing from 200 most common words in the titles and abstracted of papers appearing at ASE [1].

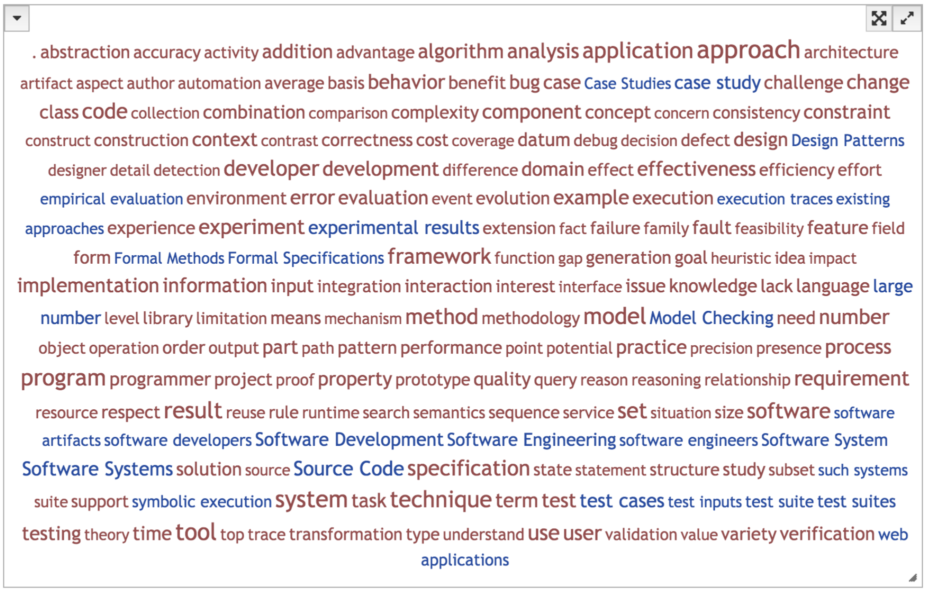

Fig. 3: Tag cloud constructed using key-phrase extraction from the 200 most common keywords and key-phrases of appearing at ASE [1]. Key-phrases are indicated in blue

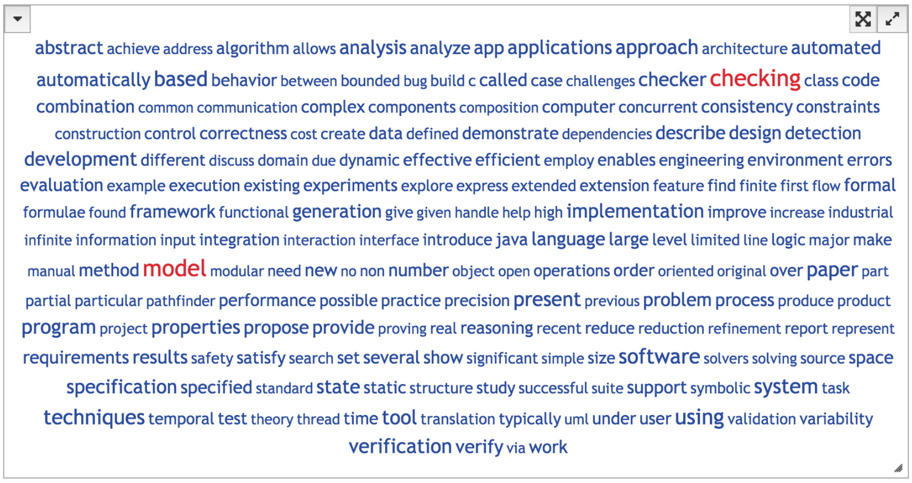

Fig. 4: Tag cloud constructed using only basic pre-processing on selection of "Model" and "Checking" from Figure 2. Selected tags are shown in red

Fig. 5: Tag Cloud including key-phrases on selection of "Model Checking" (Selection of "Model Checking" from Figure 3). Key-phrases are shown in blue, selected tags are shown in red and authors are shown in green.

7 Related Work

7.1 Tag Clouds and Navigation

Mesnage and Carmen use a Bayesian approach for navigation in tag clouds that allows tags related to one or more selected tags to be shown in the cloud, where previously clouds could only be created for one selected tag [29]. Gwizdka and Bakelaar look at displaying a tag cloud history, which allows users to keep track of their previous navigation steps, when clouds are used for pivot navigation [19]. This approach is not directly applicable to our tag clouds since we use refinement navigation where multiple tags can be selected. Hernandez et al. use multiple linked tag clouds to browse semi-structured clinical trial data [21]. These tag clouds are generated from the results of an initial search query and each represent one facet (e.g. medical condition), of the data. A multi-faceted view can also be created in ConceptCloud by moving tag categories into separate tag clouds.

7.2 Key-Phrase Extraction from Scientific Articles

Key-phrase extraction from the scientific texts is an application of common extraction techniques (see Section 2.1) to a dataset of research publications. Our approach focuses on the candidate phrase (in the form of nouns or noun phrases) selection step of the key-phrase extraction process. Given a document, candidate identification is the task of detecting all key-phrases. Candidate phrase selection methods are largely based on n-grams [22, 36, 33] or parts-of-speech (POS) tag sequences [5, 32, 24]. A comprehensive analysis of the accuracy and coverage of candidate extraction methods was carried out by Hulth [23]. She compared three methods: n-grams (excluding those that begin or end with a stop word), POS sequences (pre-defined) and (Noun Phrase) NP-chunks, excluding initial determiners ("a", "an" and "the"). In our approach we make use of a modification of the standard approach based on the extraction of NP-chunks.

8 Conclusions and Future Work

We have combined key-phrase extraction with tag clouds and concept lattices in order to provide an interface through which users can browse academic publications using key-phrases. Our approach allows formal contexts to be built automatically using their desired combination of pre-processing steps and key-phrase extraction. Browsing of the dataset is then supported by our ConceptCloud tool. The addition of key-phrases as opposed to only the keywords in the tag clouds allow users to investigate research topics more accurately and also to identify related topics.

We see many avenues for future work. The key-phrase extraction process typically includes an extraction and selection step. Our current model is based on a simple stop-word selection technique for the extracted single words. Currently, the stop-list that is used is a manually defined from common words. This does not scale in size and over different academic domains since different disciplines use varying common phrases. To overcome this drawback, a solution would be to cluster the papers into topics, compute the frequencies of words within each cluster, and build an adaptable and more comprehensive stop-word list from the intersection of frequently used words from the clusters. In future we could also improve our key-phrase extraction by using a ranking or learning approach based on computing tf/idf-like scores and features for the extracted phrases.

We could use structural syntactic and discourse representation (so called "parse thicket" [11, 12, 35]) of the whole abstract as an attribute in the context table to provide more navigation structure for the dataset. It would then also be possible to use soft matching between the abstracts in the context table to link related papers. We could also extract keywords from the publication's full text in order to enrich the tag cloud.

Our tag cloud for academic paper browsing could also be improved by adding additional data to the context table, such as citation counts for the papers and author's university affiliations.

Acknowledgements

This research is funded in part by a STIAS Doctoral Scholarship, NRF Grant 93582, RFBR Grant 14-01-93960 and the MIH Media Lab. We thank Jean Breytenbach for building the ConceptConstructor component of ConceptCloud.

References

1. Ase conferences. `http://ase-conferences.org/`.
2. Stanford nlp. `http://nlp.stanford.edu/software/`.
3. Tag cloud of obama and bush's inaugural speechs. `https://en.wikipedia.org/wiki/Tag_cloud#/media/File:State_of_the_union_word_clouds.png`.
4. C. Anslow, S. Marshall, J. Noble, and R. Biddle. Sourcevis: Collaborative software visualization for co-located environments. In *Software Visualization (VISSOFT), 2013 First IEEE Working Conference on*, pages 1–10, Sept 2013.
5. K. Barker and N. Cornacchia. Using noun phrase heads to extract document keyphrases. In *Proceedings of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence*, AI '00, pages 40–52, London, UK, UK, 2000. Springer-Verlag.
6. L. D. Caro, K. S. Candan, and M. L. Sapino. Navigating within news collections using tag-flakes. *Journal of Visual Languages and Computing*, 22(2):120 – 139, 2011.
7. C. Carpineto and G. Romano. A lattice conceptual clustering system and its application to browsing retrieval. *Machine learning*, 24(2):95–122, 1996.
8. B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order (2. ed.)*. Cambridge University Press, 2002.
9. B. Fischer. Specification-based browsing of software component libraries. *Autom. Softw. Eng.*, 7(2):179–200, 2000.

10. B. Galitsky, G. Dobrocsi, J. De La Rosa, and S. Kuznetsov. From generalization of syntactic parse trees to conceptual graphs. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6208 LNAI:185–190, 2010.
11. B. Galitsky, D. Ilvovsky, S. Kuznetsov, and F. Strok. Matching sets of parse trees for answering multi-sentence questions. pages 285–293, 2013.
12. B. Galitsky, D. Ilvovsky, S. Kuznetsov, and F. Strok. Finding maximal common sub-parse thickets for multi-sentence search. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8323 LNAI:39–57, 2014.
13. B. Galitsky, S. Kuznetsov, and D. Usikov. Parse thicket representation for multi-sentence search. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7735 LNCS:153–172, 2013.
14. B. Ganter and R. Wille. *Formal concept analysis: mathematical foundations*. Springer Science & Business Media, 2012.
15. D. N. Götzmann. Colibri/java. `http://code.google.com/p/colibri-java/`, 2007.
16. G. Greene and B. Fischer. Interactive tag cloud visualization of software version control repositories. In *Software Visualization (VISSOFT), 2015 IEEE 3rd Working Conference on*, pages 56–65, Sept 2015.
17. G. J. Greene and B. Fischer. Conceptcloud: A tagcloud browser for software archives. In *Proceedings of the 22Nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2014, pages 759–762, New York, NY, USA, 2014. ACM.
18. M. Grineva, M. Grinev, and D. Lizorkin. Extracting key terms from noisy and multitheme documents. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 661–670, New York, NY, USA, 2009. ACM.
19. J. Gwizdka and P. Bakelaar. Tag trails: navigation with context and history. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems*, pages 4579–4584. ACM, 2009.
20. K. S. Hasan and V. Ng. Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland, June 2014. Association for Computational Linguistics.
21. M.-E. Hernandez, S. M. Falconer, M.-A. Storey, S. Carini, and I. Sim. Synchronized tag clouds for exploring semi-structured clinical trial data. In *Proceedings of the 2008 Conference of the Center for Advanced Studies on Collaborative Research: Meeting of Minds*, CASCON '08, pages 4:42–4:56. ACM, 2008.
22. A. Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, pages 216–223, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
23. A. Hulth. *Combining machine learning and natural language processing for automatic keyword extraction*. 2004.
24. S. N. Kim, T. Baldwin, and M.-y. Kan. The use of topic representative words in text categorization. In *Australasian document computing symposium (ADCS 2009)*.
25. C. Lindig. Concept-based component retrieval. In *IJCAI*, pages 21–25, 1995.
26. C. Lindig. Fast concept analysis. In *Working with Conceptual Structures*, pages 152–161, 2002.

27. Z. Liu, P. Li, Y. Zheng, and M. Sun. Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 257–266, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
28. S. Lohmann, J. Ziegler, and L. Tetzlaff. Comparison of tag cloud layouts: Task-related performance and visual exploration. In *INTERACT (1)*, pages 392–404, 2009.
29. C. S. Mesnage and M. J. Carman. Tag navigation. In *Proceedings of the 2Nd International Workshop on Social Software Engineering and Applications*, SoSEA '09, pages 29–32. ACM, 2009.
30. R. Mihalcea and P. Tarau. TextRank: Bringing order into texts. In *Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, 2004.
31. C. Q. Nguyen and T. T. Phan. An ontology-based approach for key phrase extraction. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, pages 181–184, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
32. T. D. Nguyen and M.-Y. Kan. Keyphrase extraction in scientific publications. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, pages 317–326. Springer, 2007.
33. M.-S. Paukkeri, I. T. Nieminen, M. Pöllä, and T. Honkela. A language-independent approach to keyphrase extraction and evaluation. In *COLING (Posters)*, pages 83–86, 2008.
34. J. Schrammel, M. Leitner, and M. Tscheligi. Semantically structured tag clouds: An empirical evaluation of clustered presentation approaches. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 2037–2040, New York, NY, USA, 2009. ACM.
35. F. Strok, B. Galitsky, D. Ilvovsky, and S. Kuznetsov. Pattern structure projections for learning discourse structures. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8722:254–260, 2014.
36. T. Tomokiyo and M. Hurst. A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18*, pages 33–40. Association for Computational Linguistics, 2003.
37. R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. In *Ordered sets*, pages 445–470. Reidel, 1982.
38. I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning. Kea: Practical automatic keyphrase extraction. In *Proceedings of the Fourth ACM Conference on Digital Libraries*, DL '99, pages 254–255, New York, NY, USA, 1999. ACM.
39. M. J. Zaki and M. Ogihara. Theoretical foundations of association rules. In *In 3rd ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 1998.