Jan Kralj, Anita Valmarska, Marko Robnik-Šikonja, Nada Lavrač: Mining text enriched heterogeneous citation networks. In T. Cao (Ed.). Proceedings of 19th Pacific Asia Conference on Knowledge Discovery in Databases, PAKDD 2015, Ho Chi Minh City, Vietnam, pp. 672-683, 2015

Mining text enriched heterogeneous citation networks

Jan Kralj[1,2], Anita Valmarska[1,2], Marko Robnik-Šikonja[3] and Nada Lavrač[1,2,4]

[1] Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia
[2] Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia
[3] Faculty of Computer and Information Science, Večna pot 113, 1000 Ljubljana, Slovenia
[4] University of Nova Gorica, Vipavska 13, 5000 Nova Gorica, Slovenia

Abstract. The paper presents an approach to mining text enriched heterogeneous information networks, applied to a task of categorizing papers from a large citation network of scientific publications in the field of psychology. The methodology performs network propositionalization by calculating structural context vectors from homogeneous networks, extracted from the original network. The classifier is constructed from a table of structural context vectors, enriched with the bag-of-words vectors calculated from individual paper abstracts. A series of experiments was performed to examine the impact of increasing the number of publications in the network, and adding different types of structural context vectors. The results indicate that increasing the network size and combining both types of information is beneficial for improving the accuracy of paper categorization.

Keywords: · Network analysis · Heterogeneous information networks · Text mining · Document categorization · Centroid classifier · PageRank

1 Introduction

The field of *network analysis* is a well established field which has existed as an independent research discipline since the late seventies (Zachary, 1977) and early eighties (Burt and Minor, 1983). In recent years, analysis of *heterogeneous information networks* (Sun and Han, 2012) has gained popularity. In contrast to standard (homogeneous) information networks, heterogeneous networks describe heterogeneous types of entities and different types of relations. To encode even more information into the network, analysis of *enriched heterogeneous information networks*, where nodes of one type carry additional information in the form of experimental results or text documents, has arisen in recent years (Dutkowski and Ideker, 2011; Hofree et al., 2013).

This paper addresses the task of mining *text enriched heterogeneous information networks* (Grčar et al., 2013). Compared to the original methodology,

our implementation allows for the analysis of much larger heterogeneous networks given significantly decreased computation time. This was achieved by a modified PageRank computation, which takes into account only the parts of the network reachable from the given node, as explained in the methodology section. We showcase the utility of the improved approach on a large citation network in the field of psychology, where nodes—representing publications—are enriched with the publication abstracts. We analyze how the size of the network and the amount of structural information affect the accuracy of paper categorization.

The paper is structured as follows. Section 2 presents the related work. Section 3 presents the upgraded methodology used to analyze text enriched heterogeneous information networks. Section 4 presents the application of the methodology on a large data set of publications from the field of psychology. Section 5 presents the evaluation and analysis of information different components contribute to the quality of classifiers. Section 6 concludes the paper and presents the plans for further work.

2 Related work

Network mining algorithms perform data analysis in a network setting, where each data instance is connected to other instances in a network of connections.

In ranking methods like Hubs and Authorities (HITS) (Kleinberg, 1999), PageRank (Page et al., 1999), SimRank (Jeh and Widom, 2002) and diffusion kernels (Kondor and Lafferty, 2002), authority is propagated via network edges to discover high ranking nodes in the network. Sun and Han (2012) introduced the concept of *authority* ranking for heterogeneous networks with two node types (bipartite networks) to simultaneously rank nodes of both types. Sun et al. (2009) address authority ranking of all nodes types in heterogeneous networks with a star network schema, while Grčar et al. (2013) apply the PageRank algorithm to only find PageRank values of nodes of a particular node type.

Classification is another popular network analysis task. Typically, the task is to find class labels for some of the nodes in the network using known class labels for a part of the network. A typical approach to solving this problem involves propagating the labels in the network, a concept used in (Zhou et al., 2004) and (Vanunu et al., 2010). The concept of label propagation was expanded to heterogeneous networks by Hwang and Kuang (2010), performing label propagation to different node types with different diffusion parameters, similarly to the GNETMINE algorithm proposed by Ji et al. (2010). Classification in heterogeneous networks can also be assisted by ranking, as shown by the ranking based classification approach described by Sun and Han (2012).

Another important concept, related to our work, is the concept of mining *enriched* information networks. While Dutkowski and Ideker (2011) and Hofree et al. (2013) explore biological experimental data using heterogeneous biological networks, Grčar et al. (2013) perform videolectures categorization in a heterogeneous information network of nodes enriched with text information.

Following the work of Grčar et al. (2013), our work is related also to text mining. The task addressed is text categorization in which one has to predict the category of a given document, based on a set of prelabeled documents. Most text mining approaches use the bag-of-words vector representation for each processed document. The resulting high dimensional vectors can be used by any machine learning algorithm capable of handling such vectors, such as a SVM classifier (Manevitz and Yousef, 2002; Kwok, 1998; D'Orazio et al., 2014), kNN classifier (Tan, 2006), Naive Bayes classifier (Wong, 2014), or a centroid classifier (Han and Karypis, 2000).

3 Methodology

This section presents the basics of the methodology of mining text enriched information networks, first introduced by Grčar et al. (2013). The methodology combines text mining and network analysis on a text enriched heterogeneous information network (such as the citation network of scientific papers) to construct feature vectors which describe both the node content and its position in the network.

The information network is represented as a graph, a structure composed of a set of vertices V and a set of edges E. The edges may be either directed or undirected. Each edge may also have a weight assigned to it. The vertices (or nodes) of the graph in the information network are data instances. A *heterogeneous information network*, as introduced by Sun and Han (2012), is an information network with an additional structure which assigns a type to each node and edge of the network. The requirement is that all starting (or ending) points of edges of a certain type belong to the same type.

The data in a *text enriched heterogeneous information network* represents a fusion of two different data types: heterogeneous information networks and texts. Our data thus comprises of a heterogeneous information network with different node and edge types, where nodes of one designated type are text documents.

Network decomposition. In the first step of the methodology, for the designated node type (i.e., text documents), the original heterogeneous information network is decomposed into a set of homogeneous networks. In each homogeneous network, two nodes are connected if they share a particular direct or indirect link in the original heterogeneous network. Take an example of a network containing two types of nodes, *P*apers and *A*uthors, and two edge types, *Cites* (linking papers to papers) and *Written_by* (linking papers to authors). From it, we can construct two homogeneous networks of papers: the first in which two papers are connected if one paper cites another, and the second in which they are connected if they share a common author[5]. The choice of links to be used in the

[5] Depending on the application, any link between two papers, given by the heterogeneous network, may be used to construct either a directed or an undirected edge in the homogeneous network.

network decomposition step is the only manual step of the methodology: taking into account the real-world meaning of links, the domain expert will select only the decompositions relevant for the given task.

Feature vector construction. In the second step of the methodology, a set of feature vectors is calculated for each text in the original heterogeneous network: one bag-of-words vector constructed from the text document itself, and one feature vector constructed from every individual homogeneous network.

In bag-of-words (BOW) construction, each text is processed using traditional natural language processing techniques. Typically the following steps are performed: preprocessing using a tokenizer, stop-word removal, stemming, construction of N-grams of a certain length, and removal of infrequent words.

For each homogeneous networks, obtained through network decomposition, the personalized PageRank (P-PR) algorithm (Page et al., 1999) is used to construct feature vectors for each text in the network.

The personalized PageRank of node v (P-PR$_v$) in a network is defined as the stationary distribution of the position of a random walker which starts its walk in node v and at either selects one of the outgoing connections or travels to his starting location. The probability (denoted p) of continuing the walk is a parameter of the personalized PageRank algorithm and is usually set to 0.85. The PageRank vector is calculated iteratively. In the first step, the rank of node v is set to 1 and the other ranks are set to 0. Then, at each step, the rank is spread along the connections of the network using the formula

$$r^{(k+1)} = p(A^T r^{(k)}) + (1-p)r^{(0)} \tag{1}$$

where $r^{(k)}$ is the estimation of the PageRank vector after k iterations, and A is the coincidence matrix of the network, normalized so that the elements in each of its rows sum to 1.

Haveliwala and Kamvar (2003) have shown that the iteration, described by Equation 1, converges to the PageRank vector at a rate of p. In our experiments, the number of steps required ranged from 50 to 100, and since each step requires one matrix-vector multiplication, the calculation of a single P-PR vector may take several seconds for a large network, making the calculation of tens of thousands of P-PR vectors computationally very demanding.

Compared to Grčar et al. (2013), this work improves upon the original method by considerably decreasing the amount of computation for cases, where the size of the network taken into account during computation can be decreased. For each network node v, we can consider only the network G_v, composed of all the nodes and edges of the original homogeneous network that lie on paths leading from v. The P-PR$_v$ values, calculated on G_v, are equal to the P-PR values, calculated on the entire homogeneous network. If the network is strongly connected, G_v will be equal to the original network, yielding no change in the performance of the P-PR algorithm, However, if the network G_v is smaller, the calculation of the P-PR$_v$ algorithm will be faster as it is calculated on G_v instead of on the whole network. In our implementation we first estimate if the network

G_v contains less than 50% of the original nodes. This is achieved by expanding all possible paths from node v and checking the number of visited nodes in each step. If the number of visited nodes stops increasing after a maximum of 15 steps, we know we have found the network G_v and can count its nodes. If the number of nodes is still increasing, we abort the calculation of G_v. We limit the maximum number of steps because each step of G_v is computationally comparable to one step in the PageRank iterative algorithm which converges in about 50 steps. Therefore we can considerably reduce the computational burden if we do not perform too many steps in the search for G_v.

Once calculated, the resulting PageRank and BOW vectors are normalized according to the Euclidean norm.

Data fusion. The result of running both the text mining procedure and the personalized PageRank is a set of vectors $\{v_0, v_1, \ldots, v_n\}$ for each node v, where v_0 is the BOW vector, and where for each i ($1 \leq i \leq n$, where n is the number of network decompositions), v_i is the personalized PageRank vector of node v in the i-th homogeneous network. In the final step of the methodology, these vectors are combined to create one large feature vector. Using positive weights $\alpha_0, \alpha_1, \ldots, \alpha_n$ which sum to 1, a unified vector is constructed which fully describes the publication from which it was calculated. The vector is constructed as

$$v = \sqrt{\alpha_0} b \oplus \sqrt{\alpha_1} v_1 \oplus \cdots \oplus \sqrt{\alpha_n} v_n.$$

where the symbol $\oplus$ represents the concatenation of two vectors. The values of the weights α_i can either be set manually using a trial-and-error approach or can be determined automatically.

A simple way to automatically set weights is to use an optimization algorithm such as the multiple kernel learning (MKL), presented in (Rakotomamonjy et al., 2008) in which the feature vectors are viewed as linear kernels. For each i, the vector v_i corresponds to the linear mapping $\overline{v_i} : x \mapsto x \cdot v_i$.Another possibility is to determine the optimal weights using a general purpose optimization algorithm, e.g., differential evolution (Storn and Price, 1997).

4 Application and experiment description

In previous work, Grčar et al. (2013) used the described methodology to assist in the categorization of video lectures, hosted by the VideoLectures.net repository. The methodology turned out useful because of the rapid growth of the number of hosted lectures and the fact that there is a relatively large number of possible categories into which the lectures can be categorized. In this paper, the methodology is applied to a much larger network which allowed us to see 1) how the methodology scales up to big data and 2) if the information contained in the network structure is necessary at all when the textual data is abundant.

We collected data for almost one million scientific publications from the field of psychology. Like the video lectures, the publications belong to one or more

categories from a large set of possible categories. The motivation is to construct a classifier which is capable to find appropriate categories for new publications with more probable categories listed first. Such a classifier can be used to assist in the classification of new psychology articles. The same methodology and data set could be exploited to form reading recommendations based on selected paper and to assist authors in submitting their papers to the most appropriate journal.

We first describe the structure and origin of the analyzed data set. Then, we describe creation of heterogeneous network of publications and authors and experiments performed on the data set.

Data collection. The first step in the construction of a network is data collection. To the best of our knowledge, there is no freely available central database containing publications in the field of psychology. Because of this, we decided to crawl the pages connected with psychology on Wikipedia.

Wikipedia pages are grouped into categories which form a hierarchy. We visited the hierarchical tree of Wikipedia's subcategories of the category Psychology. We examined all categories up to level 5 in the hierarchy. The decision was based on the difference between the number of visited categories and the number of articles at depths 4, 5 and 6. We crawled through all Wikipedia pages, belonging to the visited categories, and extracted the DOIs (digital object identifiers) of all publications, referenced in the pages.

We queried Microsoft Academic Search (MAS) for each of the collected DOIs. If a publication was found on MAS, we collected the information about the title, authors, year of publication, the journal, ID of the publication, IDs of the authors, etc. Whenever possible, we also extracted the publication's abstract. Additionally, we collected the same information for all the publications that cite the queried publications.

Dataset. The result of our data collection process is a network consisting of 953,428 publications of which 63,862 "core publications" were obtained directly from Wikipedia pages. Other publications were citing the core publications. Each of the core publications was labelled with one or more Wikipedia categories from which it was collected. The categories at levels 3, 4 and 5 were transformed into higher level categories by climbing up the category hierarchy to level 2. This was done to decrease the total number of classes. We collected 93,977 abstracts of the publications, of which 4,551 belong to the core publications.

The heterogeneous network was decomposed into three homogeneous networks: the *paper-author-paper* (PAP) network, the *paper-cites-paper* (PP) network and a symmetric copy of the PP network in which directed edges are replaced by undirected edges (PPS).

Experiment description. In all the experiments we used the same settings to obtain the feature vectors. As in (Grčar et al., 2013), n-grams of size up to 2 and a minimum term frequency of 0 was used to calculate the BOW vectors. For

the calculation of personalized PageRank vectors the damping factor was set to 0.85 (the standard setting also used by Page et al. (1999)). In the experiments with more than one feature vector, the vectors were concatenated using weights determined by the differential evolution optimization (Storn and Price, 1997). In all the experiments we used the centroid classifier using the cosine similarity distance. This classifier first calculates the centroid vector for each class (or category) by summing and normalizing all vectors belonging to instances of that class. For a new instance with feature vector w, it then calculates the cosine similarity distance

$$d(c_i, w) = 1 - c_i \cdot w$$

which represents the proximity of the instance to class i. The class (category) with the minimal distance is selected as the prediction outcome. We also use the "top n" classifier, where the classifier returns n classes with the minimal distances. As in (Grčar et al., 2013), we consider a classifier successful if it correctly predicts at least one label of an instance.

We use the centroid classifier for two reasons. First, Grčar et al. (2013) show that it performs just as well as the SVM and the k-nearest neighbor classifier. Second, for large networks calculating all the personalized PageRank vectors is computationally very expensive. As shown in (Grčar et al., 2013), the centroids of each class can be calculated in a single iteration of the PageRank algorithm.

We performed three sets of experiments using different number of papers and different homogenization of the heterogeneous network.

In the first set, we use the publications for which abstracts are available. Because most of the 93,977 qualifying papers are not core publications, we construct only two feature vectors for each publication: a bag-of-words (BOW) vector and a personalized PageRank vector obtained from the PAP network. We examine how the predictive power of the classifier increases as the number of publications used increases. We use 10,000, 20,000, 30,000, 40,000, 50,000, 70,000 and 93,977 publications.

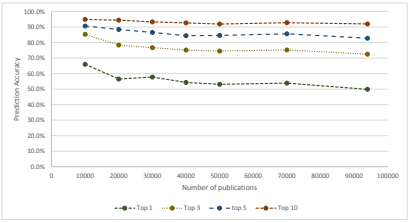
In the second round of experiments all the collected papers are used (953,428 papers). Because the papers are labelled using citations the PP and PPS networks are not used because the links in this network were used to label the papers. Since the abstracts are not available for most of these papers, only the personalized PageRank vectors obtained from the PAP network are used in the classification.

In the third round of experiments, we use only the core publications for which an abstract is available (4,551 papers). While this is the smallest data set, it allows us to use all of the feature vectors the methodology provides: the BOW vectors and the personalized PageRank obtained from all three networks (PP, PPS and PAP).

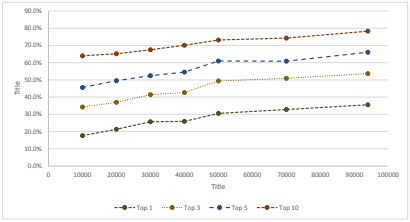
5 Evaluation and results

In each of the experiments, described in Sect. 4, we predicted the labels of the analyzed publications. The classification accuracy was measured for the top 1,

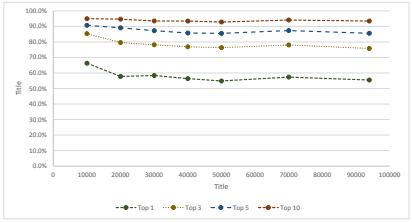
3, 5 and 10 labels, returned by the classifiers. For each experiment the data set was split into a training set, a validation set and a test set. In the first and third round of experiments, the sizes of the testing and validation set were fixed to 2500 instances, all the remaining instances were used for training. In the second round of experiments, the size of the validation and testing set was set to 1500 instances. The centroids of all classes were calculated using the training set and concatenated according to the weights optimized using the validation set. The performance of the algorithm (the percentage of papers for which the label is correctly predicted) was calculated using the test set.



(a) The centroid classifier using BOW.



(b) The centroid classifier using PAP.



(c) The centroid classifier using both BOW and PAP.

Fig. 1. The classification accuracy of classifiers using different number of publications to predict labels.

The results of the first round of experiments are shown in Fig. 1. The performance of the classifier using BOW vectors does not increase with more instances, while the classifier using PAP vectors is steadily improving as we increase the number of publications. The classifier using both BOW and PAP vectors consistently outperforms the individual classifiers. This shows that combining the network structural information and the content of the publication is useful. As the performance of the PAP classifier increases, the gap between the BOW classifier and the classifier using both vectors also increases. The results obtained with all the 93,977 publications are shown also in Table 1.

The classifier using the full PAP network (calculated in the experiment 2), also shown in Table 1, outperforms the classifiers using all other networks, showing that increasing the network size does help the classification. However, its performance is still lower than that of the BOW classifier for smaller networks.

Table 1. The classification accuracy of the centroid classifiers in the first and second round of experiments (the publications with abstracts).

top N	BOW+PAP	PAP	BOW	PAP
1	55.5	35.6	49.9	38.8
3	75.8	53.7	72.6	59.3
5	85.6	66.0	82.8	71.0
10	93.5	78.3	92.0	81.4

It appears that authors in the field of psychology are not strictly limited to one field of research, making prediction using co-authorship information difficult.

Table 2. The classification accuracy of the centroid classifiers in the second round of experiments, (the core publications with abstracts).

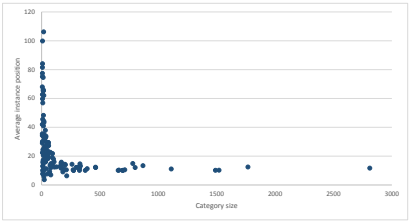
top N	All	noBOW	noPAP	noPP	noPPS	BOW+PAP	BOW+PP	BOW+PPS	PAP+PP	PAP+PPS	PP+PPS
1	64.9	49.5	64.7	61.3	65.9	57.7	59.1	62.8	50.3	49.0	44.3
3	84.3	64.6	82.5	74.3	83.5	80.0	78.4	82.0	65.6	63.7	56.7
5	90.2	72.5	90.0	88.6	90.6	88.1	86.4	89.6	72.7	72.0	64.0
10	95.4	81.7	95.4	94.7	95.9	94.9	94.4	95.1	81.5	81.4	73.2

top N	BOW	PP	PPS	PAP
1	55.4	43.5	42.9	30.6
3	78.8	55.8	54.2	47.5
5	87.4	62.4	61.5	58.9
10	93.8	72.1	72.8	72.7

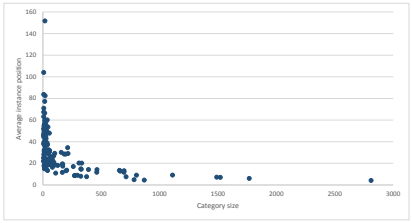
Table 2 shows the accuracies, obtained in the third round of experiments. Because more information was extracted from the network, these results are the most comprehensive overview of the methodology. The results show that using a symmetric citation network (PPS), i.e. allowing the PageRank to use both directions of a citation yields better results than using the unidirectional citation network (PP). Combining both the PP and PPS vectors does not improve the performance of the classifier, which means that vectors, obtained from the PP network, carry no information that is not already contained in the PPS network. However, this is an exception and training classifiers with other vectors combinations increases the prediction accuracy over single vectors: using both BOW and PAP is better than using only BOW, and adding also PP increases the performance even further.

We also analyze the performance of classifiers for different class values. We analyze each class c in the following way. First, we obtain the ordered list of labels that the classifier returns for each test instance from class c. In this list, class values are ordered according to the distance between the instance and the (already computed) class centroids. The first element in the list is the class value whose centroid is closest to the given instance. We then find the rank of class c on this list. For each instance, we compute the minimum value of n for which the top-n classifier predicts class c for this instance. For each class value, we average the obtained ranks n over test instances with this class. This gives us an estimate of the ranking error.

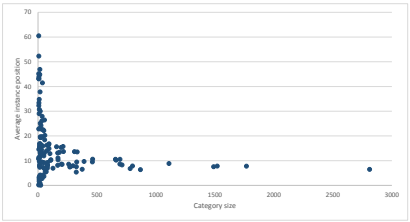
We plot these average ranks versus number of instances with each class value. The results are shown in Fig. 2. The graphs are similar for classifiers using BOW, PPS and PAP vectors. We can see that classes containing a small number of instances have considerably higher average ranks than classes with many instances, meaning that prediction is much less successful for underrepresented class values. The classifier using PP vectors is the only classifier for which this trend does not appear. For the PP classifier, the results for small classes show much more noise than for larger classifiers, but average ranks (i.e., error) does not decrease with increasing number of instances.



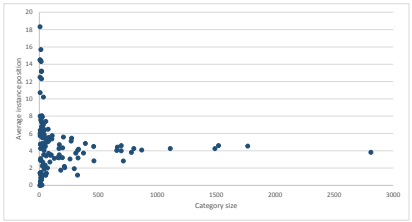
(a) The centroid classifier using BOW.



(b) The centroid classifier using PAP.



(c) The centroid classifier using PPS.



(d) The centroid classifier using PP.

Fig. 2. Graphs showing the average index of a class versus the class size.

6 Conclusions and further work

While network analysis in general is an established field of research, analysis of heterogeneous networks is a much newer field. Methods taking the heterogeneous nature of the networks into account show an improved performance, as shown in Davis et al. (2011). Some methods like RankClus and others presented in (Sun and Han, 2012) are capable of solving tasks that cannot even be defined on homogeneous information networks (like clustering two disjoint sets of entities). Another important novelty is joining network analysis with the analysis of data, either in the form of text documents or results obtained from various experiments (Dutkowski and Ideker, 2011), (Hofree et al., 2013) and (Grčar et al., 2013).

This paper presents a more efficient implementation of the methodology by (Grčar et al., 2013), which combines the information from heterogeneous networks with textual data. By improving the computational efficiency of the approach we were able to address a novel application, i.e. the analysis of a large citation network of psychology papers. Our contribution is also the analysis of performance with different number of instances and different types of network structures included. The results show that relational information hidden in the network structure is beneficial for classification, while the errors are shown to be mostly due to low number of instances for some categories.

In the work presented, we only use a part of the information we collected about the publications. In future, we will to examine how to incorporate the temporal information into our methodology; we have already collected the year of publication, which allows us to observe the dynamics of categories, aiming to improve the classification accuracy. In addition, we plan to use a combination of network analysis and data mining on PubMed and DBLP articles. We will also address biological networks enriched with experimental data and texts.

References

Burt, R. and Minor, M. (1983). *Applied Network Analysis: a Methodological Introduction*. Sage Publications.

Davis, D., Lichtenwalter, R., and Chawla, N. V. (2011). Multi-relational link prediction in heterogeneous information networks. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 281–288.

D'Orazio, V., Landis, S. T., Palmer, G., and Schrodt, P. (2014). Separating the wheat from the chaff: Applications of automated document classification using support vector machines. *Polytical Analysis*, 22(2):224–242.

Dutkowski, J. and Ideker, T. (2011). Protein networks as logic functions in development and cancer. *PLoS Computational Biology*, 7(9).

Grčar, M., Trdin, N., and Lavrač, N. (2013). A methodology for mining document-enriched heterogeneous information networks. *The Computer Journal*, 56(3):321–335.

Han, E.-H. and Karypis, G. (2000). Centroid-based document classification: Analysis and experimental results. In *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 424–431. Springer.

Haveliwala, T. and Kamvar, S. (2003). The second eigenvalue of the Google matrix. Technical report, Stanford InfoLab.

Hofree, M., Shen, J. P., Carter, H., Gross, A., and Ideker, T. (2013). Network-based stratification of tumor mutations. *Nature Methods*, 10(11):1108–1115.

Hwang, T. and Kuang, R. (2010). A heterogeneous label propagation algorithm for disease gene discovery. In *Proceedings of SIAM International Conference on Data Mining*, pages 583–594.

Jeh, G. and Widom, J. (2002). SimRank: A measure of structural-context similarity. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 538–543. ACM.

Ji, M., Sun, Y., Danilevsky, M., Han, J., and Gao, J. (2010). Graph regularized transductive classification on heterogeneous information networks. In *Proceedings of the 25th European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pages 570–586.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

Kondor, R. I. and Lafferty, J. D. (2002). Diffusion kernels on graphs and other discrete input spaces. In *Proceedings of the 19th International Conference on Machine Learning*, pages 315–322.

Kwok, J. T.-Y. (1998). Automated text categorization using support vector machine. In *Proceedings of the 5th International Conference on Neural Information Processing*, pages 347–351.

Manevitz, L. M. and Yousef, M. (2002). One-class SVMs for document classification. *Journal of Machine Learning Research*, 2:139–154.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Rakotomamonjy, A., Bach, F., Canu, S., and Grandvalet, Y. (2008). SimpleMKL. *Journal of Machine Learning Research*, 9:2491–2521.

Storn, R. and Price, K. (1997). Differential evolution; a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359.

Sun, Y. and Han, J. (2012). *Mining Heterogeneous Information Networks: Principles and Methodologies*. Morgan & Claypool Publishers.

Sun, Y., Yu, Y., and Han, J. (2009). Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806.

Tan, S. (2006). An effective refinement strategy for KNN text classifier. *Expert Syst. Appl.*, 30(2):290–298.

Vanunu, O., Magger, O., Ruppin, E., Shlomi, T., and Sharan, R. (2010). Associating genes and protein complexes with disease via network propagation. *PLoS Computational Biology*, 6(1).

Wong, T.-T. (2014). Generalized Dirichlet priors for Naïve Bayesian classifiers with multinomial models in document classification. *Data Mining and Knowledge Discovery*, 28(1):123–144.

Zachary, W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. (2004). Learning with local and global consistency. *Advances in Neural Information Processing Systems*, 16(16):321–328.