High-quality Training Data Selection using Latent Topics for Graph-based Semi-supervised Learning

Akiko Eriguchi
Ochanomizu University
2-1-1 Otsuka Bunkyo-ku Tokyo, Japan
g0920506@is.ocha.ac.jp

Ichiro Kobayashi
Ochanomizu University
2-1-1 Otsuka Bunkyo-ku Tokyo, Japan
koba@is.ocha.ac.jp

Abstract

In a multi-class document categorization using graph-based semi-supervised learning (GBSSL), it is essential to construct a proper graph expressing the relation among nodes and to use a reasonable categorization algorithm. Furthermore, it is also important to provide high-quality correct data as training data. In this context, we propose a method to construct a similarity graph by employing both surface information and latent information to express similarity between nodes and a method to select high-quality training data for GBSSL by means of the PageRank algorithm. Experimenting on Reuters-21578 corpus, we have confirmed that our proposed methods work well for raising the accuracy of a multi-class document categorization.

1 Introduction

Graph-based semi-supervised learning (GBSSL) algorithm is known as a useful and promising technique in natural language processings. It has been widely used for solving many document categorization problems (Zhu and Ghahramani, 2002; Zhu et al., 2003; Subramanya and Bilmes, 2008).

A good accuracy of GBSSL depends on success in dealing with three crucial issues: graph construction, selection of high-quality training data, and categorization algorithm. We particularly focus on the former two issues in our study.

In a graph-based categorization of documents, a graph is constructed based on a certain relation between nodes (i.e. documents). It is similarity that is often used to express the relation between nodes in a graph. We think of two types of similarity: the one is between surface information obtained by document vector (Salton and McGill, 1983) and the other is between latent information obtained by word probabilistic distribution (Latent Dirichlet Allocation (Blei et al., 2003)). Here, we propose a method. We use both surface information and latent information at the ratio of $(1-\alpha) : \alpha (0 \leq \alpha \leq 1)$ to construct a similarity graph for GBSSL, and we investigate the optimal α for raising the accuracy in GBSSL.

In selecting high-quality training data, it is important to take two aspects of data into consideration: quantity and quality. The more the training data are, the better the accuracy becomes. We do not always, however, have a large quantity of training data. In such a case, the quality of training data is generally a key for better accuracy. It is required to assess the quality of training data exactly. Now, we propose another method. We use the PageRank algorithm (Brin and Page, 1998) to select high-quality data, which have a high centrality in a similarity graph of training data (i.e. labeled data) in each category.

We apply our methods to solving the problem of a multi-class document categorization. We introduce PRBEP (precision recall break even point) as a measure which is popular in the area of information retrieval. We evaluate the results of experiments for each category and for the whole category. We confirm that the way of selecting the high-quality training data from data on a similarity graph based on both surface information and latent information is superior to that of selecting from a graph based on just surface information or latent information.

2 Related studies

Graph-based semi-supervised learning has recently been studied so much and applied to many applications (Subramanya and Bilmes, 2008; Subramanya and Bilmes, 2009; Subramanya et al., 2010; Dipanjan and Petrov, 2011; Dipanjan and Smith, 2012; Whitney and Sarkar, 2012).

Proceedings of the ACL Student Research Workshop, pages 136–141,
Sofia, Bulgaria, August 4-9 2013.

Subramanya and Bilmes (2008; 2009) have proposed a soft-clustering method using GBSSL and have shown that their own method is better than the other main clustering methods of those days. Subramanya et al. (2010) have also applied their method to solve the problem of tagging and have shown that it is useful. Dipanjan and Petrov (2011) have applied a graph-based label propagation method to solve the problem of part-of-speech tagging. They have shown that their proposed method exceeds a state-of-the-art baseline of those days. Dipanjan and Smith (2012) have also applied GBSSL to construct compact natural language lexicons. To achieve compactness, they used the characteristics of a graph. Whitney and Sarkar (2012) have proposed the bootstrapping learning method in which a graph propagation algorithm is adopted.

There are two main issues in GBSSL: the one is the way of constructing a graph to propagate labels, and the other is the way of propagating labels. It is essential to construct a good graph in GBSSL (Zhu, 2005). On the one hand, graph construction is a key to success of any GBSSL. On the other hand, as for semi-supervised learning, it is quite important to select better training data (i.e. labeled data), because the effect of learning will be changed by the data we select as training data.

Considering the above mentioned, in our study, we focus on the way of selecting training data so as to be well propagated in a graph. We use the PageRank algorithm to select high-quality training data and evaluate how our proposed method influences the way of document categorization.

3 Text classification based on a graph

The details of our proposed GBSSL method in a multi-class document categorization are as follows.

3.1 Graph construction

In our study, we use a weighted undirected graph $G = (V, E)$ whose node and edge represent a document and the similarity between nodes, respectively. Similarity is regarded as weight. *V* and *E* represent nodes and edges in a graph, respectively. A graph G can be represented as an adjacency matrix, and $w_{ij} \in \boldsymbol{W}$ represents the similarity between nodes i and j. In particular, in the case of GBSSL method, the similarity between nodes are formed as $w_{ij} = sim(\boldsymbol{x}_i, \boldsymbol{x}_j)\delta(j \in K(i))$. $K(i)$ is a set of i's k-nearest neighbors, and $\delta(z)$ is 1 if z is true, otherwise 0.

3.2 Similarity in a graph

Generally speaking, when we construct a graph to represent some relation among documents, cosine similarity (sim_{cos}) of document vectors is adopted as a similarity measure based on surface information. In our study, we add the similarity (sim_{JS}) based on latent information and the similarity (sim_{cos}) based on surface information in the proportion of $\alpha : (1-\alpha)(0 \leq \alpha \leq 1)$. We define the sum of sim_{JS} and sim_{cos} as sim_{nodes} (see, Eq. (1)).

In Eq. (1), P and Q represent the latent topic distributions of documents S and T, respectively. We use Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to estimate the latent topic distribution of a document, and we use a measure Jensen-Shannon divergence (D_{JS}) for the similarity between topic distributions. Incidentally, sim_{JS} in Eq (1) is expressed by Eq. (2).

$$\begin{aligned} sim_{nodes}(S,T) &\equiv \alpha * sim_{JS}(P,Q) \\ &+(1-\alpha) * sim_{cos}(tfidf(S), tfidf(T)) \end{aligned} \quad (1)$$

$$sim_{JS}(P,Q) \equiv 1 - D_{JS}(P,Q) \quad (2)$$

3.3 Selection of training data

We use the graph-based document summarization methods (Erkan and Radev, 2004; Kitajima and Kobayashi, 2012) in order to select high-quality training data. Erkan and Radev (2004) proposed a multi-document summarization method using the PageRank algorithm (Brin and Page, 1998) to extract important sentences. They showed that it is useful to extract the important sentences which have higher PageRank scores in a similarity graph of sentences. Then, Kitajima and Kobayashi (2012) have expanded the idea of Erkan and Radev's. They introduced latent information to extract important sentences. They call their own method TopicRank.

We adopt TopicRank method in our study. In order to get high-quality training data, we first construct a similarity graph of training data in each category, and then compute a TopicRank score for each training datum in every category graph. We employ the data with a high TopicRank score as training data in GBSSL.

In TopicRank method, Kitajima and Kobayashi (2012) regard a sentence as a node in a graph on

surface information and latent information. The TopicRank score of each sentence is computed by Eq. (3). Each sentence is ranked by its TopicRank score. In Eq. (3), d indicates a damping factor. We, however, deal with documents, so we replace a sentence with a document (i.e. sentences) as a node in a graph. In Eq. (3), N indicates total number of documents, $adj[u]$ indicates the adjoining nodes of document u.

$$r(u) = d \sum_{v \in adj[u]} \frac{sim_{nodes}(u,v)}{\sum_{z \in adj[v]} sim_{nodes}(z,v)} r(u) + \frac{1-d}{N} \quad (3)$$

3.4 Label propagation

We use the label propagation method (Zhu et al., 2003; Zhou et al., 2004) in order to categorize documents. It is one of graph-based semi-supervised learnings. It estimates the value of label based on the assumption that the nodes linked to each other in a graph should belong to the same category. Here, $\boldsymbol{W}$ indicates an adjacency matrix. l indicates the number of training data among all n nodes in a graph. The estimation values $\boldsymbol{f}$ for n nodes are obtained as the solution (Eq. (6)) of the following objective function of an optimal problem (Eq. (4)). The first term in Eq. (4) expresses the deviation between an estimation value and a correct value of training data. The second term in Eq. (4) expresses the difference between the estimation values of the nodes which are next to another in the adjacency graph. $\lambda(> 0)$ is a parameter balancing both of the terms. Eq. (4) is transformed into Eq. (5) by means of $\boldsymbol{L}$. $\boldsymbol{L}(\equiv \boldsymbol{D} - \boldsymbol{W})$ is called the Laplacian matrix. $\boldsymbol{D}$ is a diagonal matrix, each diagonal element of which is equal to the sum of elements in $\boldsymbol{W}$'s each row (or column).

$$J(\boldsymbol{f}) = \sum_{i=1}^{l} (y^{(i)} - f^{(i)})^2 + \lambda \sum_{i<j} w^{(i,j)} (f^{(i)} - f^{(j)})^2 \quad (4)$$

$$= ||\boldsymbol{y} - \boldsymbol{f}||_2^2 + \lambda \boldsymbol{f}^T \boldsymbol{L} \boldsymbol{f} \quad (5)$$

$$\boldsymbol{f} = (\boldsymbol{I} + \lambda \boldsymbol{L})^{-1} \boldsymbol{y} \quad (6)$$

4 Experiment

4.1 Experimental settings

We use Reuters-21578 corpus data set[1] collected from the Reuters newswire in 1987 as target documents for a multi-class document categorization. It consists of English news articles (classified into 135 categories). We use the "ModApte" split to get training documents (i.e. labeled data) and test documents (i.e. unlabeled data), extract documents which have only its title and text body, and apply the stemming and the stop-word removal processes to the documents. Then, following the experimental settings of Subramanya and Bilmes (2008)[2] , we use 10 most frequent categories out of the 135 potential topic categories: *earn, acq, grain, wheat, money-fx, crude, trade, interest, ship,* and *corn*. We apply the one-versus-the-rest method to give a category label to each test document. Labels are given when the estimation values of each document label exceed each of the predefined thresholds.

We prepare 11 data sets. Each data set consists of 3299 common test data and 20 training data. We use 11 kinds of categories of training data: the above mentioned 10 categories and a category (*other*) which indicates 125 categories except 10 categories. The categories of 20 training data are randomly chosen only if one of the 11 categories is chosen at least once.

Selecting high-quality training data, we use the Gibbs sampling for latent topic estimation in LDA. The number of iteration is 200. The number of latent topics in the target documents is decided by averaging 10 trials of estimation with perplexity (see, Eq. (7)). Here, N is the number of all words in the target documents. w_{mn} is the n-th word in the m-th document. θ is an occurrence probability of the latent topics for the documents. ϕ is an occurrence probability of the words for every latent topic.

$$P(\boldsymbol{w}) = exp(-\frac{1}{N} \sum_{mn} log(\sum_{z} \theta_{mz} \phi_{z w_{mn}})) \quad (7)$$

In each category, a similarity graph is constructed for the TopicRank method. The number of nodes (i.e. $|V_{category}|$) in a graph corresponds to

[1] http://www.daviddlewis.com/resources/testcollections/reuters21578/

[2] Our data sets lack any tags and information excluding a title and a text body. Therefore, we cannot directly compare with Subramanya and Bilmes' results.

the total number of training data in each category, and the number of edges is $E = (|V_{category}| \times |V_{category}|)$. So, the graph is a complete graph. The parameter α in Eq (1) is varied from 0.0 to 1.0 every 0.1. We regard the average of TopicRank scores after 5 trials as the TopicRank score of each document. The number of training data in each category is decided in each target data set. We adopt training data with a higher TopicRank score from the top up to the predefined number.

In label propagation, we construct another kind of similarity graph. The number of nodes in a graph is $|V_{l+u}| = n(= 3319)$, and the similarity between nodes is based on only surface information (in the case of $\alpha = 0$ in Eq. (1)). The parameter k in the k-nearest neighbors method is $k \in \{2, 10, 50, 100, 250, 500, 1000, 2000, n\}$, the parameter λ in the label propagation method, is $\lambda \in \{1, 0.1, 0.01, 1e-4, 1e-8\}$. Using one of the 11 data sets, we decide a pair of optimal parameters (k, λ) for each category. We categorize the remaining 10 data sets by means of the decided parameters. Then, we obtain the value of precision recall break even point (PRBEP) and the average of PRBEP in each category. The value of PRBEP is that of precision or recall at the time when the former is equal to the latter. It is often used as an index to measure the ability of information retrieval.

4.2 Result

Table 1 shows a pair of the optimal parameters (k, λ) in each category corresponding to the value of α ranging from 0.0 to 1.0 every 0.1. Figures from 1 to 10 show the experimental results in using these parameters in each category. The horizontal axis indicates the value of α and the vertical axis indicates the value of PRBEP. Each figure shows the average of PRBEP in each category after 10 trials for each α. Fig. 11 shows how the relative ratio of PRBEP changes corresponding to each α in each category, when we let the PRBEP at $\alpha = 0$ an index 100. Fig. 12 shows the macro average of PRBEP after 10 trials in the whole category corresponding to each α. Error bars indicate the standard deviations.

In all figures, the case at $\alpha = 0$ means that only surface information is used for selecting the training data. The case at $\alpha = 1$ means that only latent information is used. The other cases at $\alpha \neq 0$ or 1 mean that both latent information and surface information are mixed at the ratio of $\alpha : (1 - \alpha)$ $(0 < \alpha < 1)$.

First, we tell about Fig. 1-10. On the one hand, in Fig. 4, 5, 6, 8, 10, the PRBEPs at $\alpha \neq 0$ are greater than that at $\alpha = 0$, although the PRBEP at $\alpha = 1$ is less than that at $\alpha = 0$ in Fig. 4. On the other hand, in Fig. 2, 7, the PRBEPs at $\alpha \neq 0$ are less than that at $\alpha = 0$. In Fig. 1, 3, 9, the PRBEPs at $\alpha \neq 0$ fluctuate widely or narrowly around that at $\alpha = 0$. In addition, the PRBEPs at $\alpha = 0$ range from 7.7 to 74.3 and those at $\alpha = 1$ range from 8.0 to 72.6 in all 10 figures. It is hard to find significant correlation between PRBEP and α.

Secondly, in Fig. 11, some curves show an increasing trend and others show a decreasing trend. At best, the maximum value is three times as large as that at $\alpha = 0$. At worst, the minimum is one-fifth. Indexes at $\alpha \neq 0$ are greater than or equal to an index 100 at $\alpha = 0$ in most categories.

Finally, in Fig. 12, the local maximums are 46.2, 46.9, 45.0 respectively at $\alpha = 0.2, 0.6, 0.9$. The maximum is 46.9 at $\alpha = 0.6$. The minimum value of the macro average is 35.8 at $\alpha = 0$, though the macro average at $\alpha = 1$ is 43.4. Hence, the maximum macro average is greater than that at $\alpha = 1$ by 3.5% and still greater than that at $\alpha = 0$ by 11.1%. The macro average at $\alpha = 1$ is greater than that at $\alpha = 0$ by 7.6%. Furthermore, the macro average increases monotonically from 35.8 to 46.2 as α increases from 0.0 to 0.2. When α is more than 0.2, the macro averages fluctuate within the range from 40.3 to 46.9. It follows that the macro average values at $0.1 \leq \alpha \leq 1$ are greater than that at $\alpha = 0$. What is more important, the macro averages at $\alpha = 0.2, 0.4, 0.6, 0.7, 0.9$ are greater than that at $\alpha = 1$ and of course greater than that at $\alpha = 0$.

5 Discussion

Looking at each Fig. 1-10, each optimal α at which PRBEP is the maximum is different and not uniform in respective categories. So, we cannot simply tell a specific ratio of balancing both information (i.e. surface information and latent information) which gives the best accuracy.

From a total point of view, however, we can see a definite trend or relationship. In Fig. 11, we can see the upward tendency of PREBP in half of categories. Indexes of the PRBEP at $\alpha \geq 0.1$ are greater than or equal to 100 in most categories.

Table 1: the optimal parameters (k, λ) for each category

Category\α	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
earn	(500, 1)	(50, 1)	(1000, 1)	(1000, 1)	(50, 1)	(50, 1)	(50, 1)	(50, 1)	(50, 1)	(50, 1)	(50, 1)
acq	(100, 0.01)	(100, 0.01)	(100, 0.01)	(2, 1)	(100, 0.01)	(100, 0.01)	(100, 1e-8)	(100, 1e-8)	(100, 1e-8)	(100, 1e-8)	(100, 1e-8)
money-fx	(250, 0.01)	(100, 1e-8)	(10, 1e-4)	(100, 1e-8)	(2, 0.1)	(2, 0.1)	(2, 1e-8)	(250, 1e-8)	(2, 0.1)	(2, 1e-8)	(250, 1e-8)
grain	(250, 0.1)	(2000, 1e-4)	(100, 1)	(250, 0.1)	(100, 1)	(50, 1)	(250, 1)	(50, 1)	(50, 1)	(50, 1)	(100, 1)
crude	(50, 0.1)	(2, 1)	(250, 0.01)	(50, 1e-8)	(10, 0.01)	(250, 0.01)	(250, 0.01)	(250, 1e-8)	(10, 0.01)	(250, 0.01)	(250, 0.01)
trade	(2, 1)	(10, 0.1)	(50, 0.01)	(10, 1e-8)	(10, 1e-8)	(10, 1e-8)	(50, 1e-8)	(10, 1e-8)	(10, 1e-4)	(10, 0.1)	(10, 0.1)
interest	(10, 1)	(50, 1e-8)	(50, 1e-8)	(10, 1)	(2, 0.1)	(250, 1e-8)	(250, 0.01)	(250, 0.01)	(2, 1)	(2, 0.1)	(500, 1e-8)
ship	(3318, 1)	(50, 1)	(50, 1)	(250, 0.1)	(50, 0.1)	(50, 0.1)	(50, 1e-8)	(50, 1e-8)	(100, 0.1)	(100, 0.1)	(50, 0.01)
wheat	(500, 1e-8)	(500, 1e-8)	(250, 1e-8)	(500, 1e-8)	(500, 0.01)	(1000, 0.01)	(500, 1e-8)	(250, 1e-8)	(250, 1e-8)	(250, 1e-8)	(250, 1e-8)
corn	(10, 1e-8)	(100, 1e-8)	(250, 1e-8)	(10, 1e-8)	(250, 1e-8)	(250, 1e-4)	(500, 1e-8)	(100, 1e-8)	(250, 1e-8)	(50, 0.01)	(250, 1e-4)

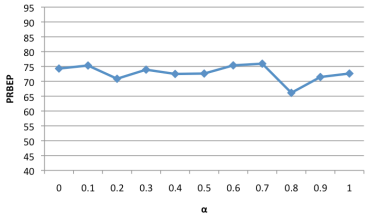

Figure 1: *earn*

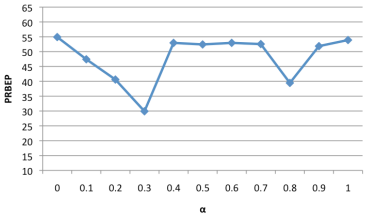

Figure 2: *acq*

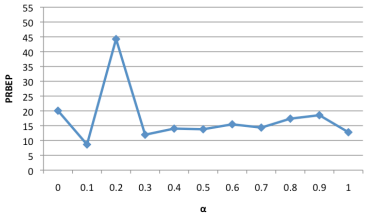

Figure 3: *money-fx*

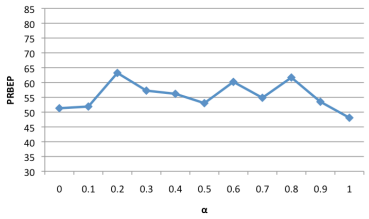

Figure 4: *grain*

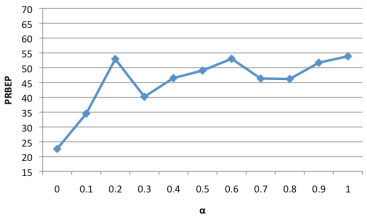

Figure 5: *crude*

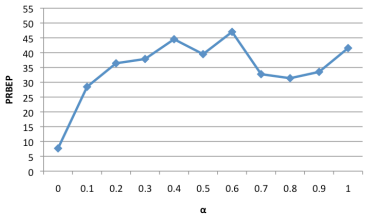

Figure 6: *trade*

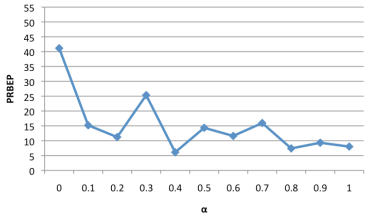

Figure 7: *interest*

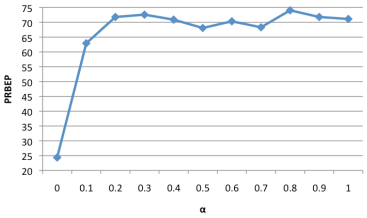

Figure 8: *ship*

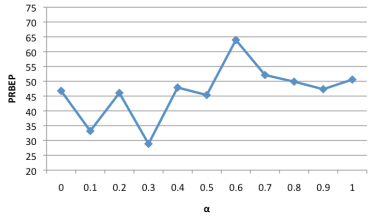

Figure 9: *wheat*

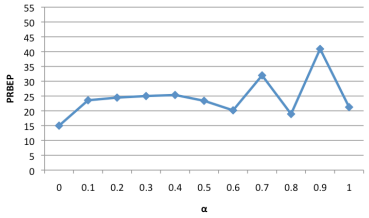

Figure 10: *corn*

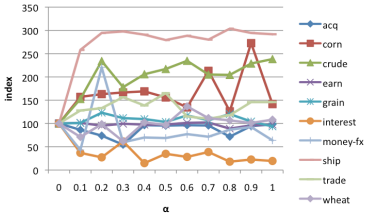

Figure 11: Relative value

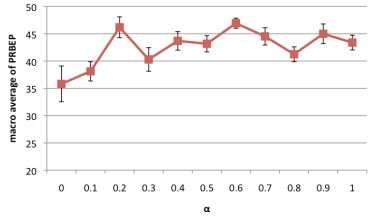

Figure 12: Macro average

The macro average of the whole category is shown in Fig. 12. Regarding the macro average at $\alpha = 0$ as a baseline, the macro average at $\alpha = 1$ is greater than that at $\alpha = 0$ by 7.6% and still more, the maximum at $\alpha = 0.6$ is greater by 11.1%. Besides, five macro averages at $0.1 \leq \alpha \leq 1$ are greater than that at $\alpha = 1$. Therefore, we can say that using latent information gives a higher accuracy than using only surface information and that using both information gives a higher accuracy than using only latent information. So, if a proper α is decided, we will get a better accuracy.

6 Conclusion

We have proposed methods to construct a similarity graph based on both surface information and latent information and to select high-quality training data for GBSSL. Through experiments, we have found that using both information gives a better accuracy than using either only surface information or only latent information. We used the PageRank algorithm in the selection of high-quality training data. In this condition, we have confirmed that our proposed methods are useful for raising the accuracy of a multi-class document categorization using GBSSL in the whole category.

Our future work is as follows. We will verify in other data corpus sets that the selection of high-quality training data with both information gives a better accuracy and that the optimal α is around 0.6. We will revise the way of setting a pair of the optimal parameters (k, λ) and use latent information in the process of label propagation.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*.

Sergey Brin and Lawrence Page. 1998. The Anatomy of a Large-scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, pages 107–117.

Das Dipanjan and Noah A. Smith. 2012. Graph-based lexicon expansion with sparsity-inducing penalties. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL HLT '12, pages 677–687.

Das Dipanjan and Slav Petrov. 2011. Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* - Vol. 1, pages 600–609.

Güneş Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research 22*, pages 457-479.

Güneş Erkan. 2006. Language Model-Based Document Clustering Using Random Walks. *Association for Computational Linguistics*, pages 479–486.

Risa Kitajima and Ichiro Kobayashi. 2012. Multiple-document Summarization baed on a Graph constructed based on Latent Information. In *Proceedings of ARG Web intelligence and interaction*, 2012-WI2-1-21.

Gerard Salton and Michael J. McGill. 1983. Introduction to Modern Information Retrieval. McGraw-Hill.

Amarnag Subramanya and Jeff Bilmes. 2008. Soft-Supervised Learning for Text Classification. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1090–1099.

Amarnag Subramanya and Jeff Bilmes. 2009. Entropic graph regularization in non-parametric semi-supervised classification. In *Proceedings of NIPS*.

Amarnag Subramanya, Slav Petrov and Fernando Pereira. 2010. Efficient graph-based semi-supervised learning of structured tagging models. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 167–176.

Dengyong Zhou, Oliver Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with Local and Global Consistency. In *NIPS 16*.

Xiaojin Zhu and Zoubin Ghahramani. 2002. Learning from Labeled and Unlabeled Data with Label Propagation. Technical report, Carnegie Mellon University.

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. 2003. Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In *Proceedigns of the International Conference on Machine Learning (ICML)*.

Xiaojin Zhu. 2005. Semi-Supervised Learning with Graphs. PhD thesis, Carnegie Mellon University.

Max Whitney and Anoop Sarkar. 2012. Bootstrapping via Graph Propagation. *The 50th Annual Meeting of the Association for Computational Linguistics*.