

Active Learning of Constraints for Semi-supervised Text Clustering *

Ruizhang Huang Wai Lam Zhigang Zhang
Department of Systems Engineering & Engineering Management
The Chinese University of Hong Kong
Shatin, Hong Kong
{rzhuang, wlam, zgzhang}@se.cuhk.edu.hk

Abstract

This paper investigates active learning of constraints for semi-supervised document clustering. We make use of the intermediate clustering results to guide the document pair selection for obtaining user judgments for constraint generation. A gain function is designed for choosing the most informative document pairs given the current cluster assignments. This gain function measures how much we can learn by revealing the judgment of the document pairs. Two methods are investigated, namely, independent gain model and dependent gain model. In the independent gain model, we assume that the information learned by revealing the judgment of a document pair is independent of revealing the judgment of other document pairs. The dependent gain model also considers previously chosen documents to avoid redundant selection and maximize the gain collectively for a set of document pairs. Constrained semi-supervised clustering and gain directed document pair selection are conducted in an iterative manner. We have conducted extensive experiments on several real-world corpora. The results demonstrate that the intermediate clustering assignments and the interactions among a set of document pairs are useful for improving the clustering performance. Our approach is also superior to a recent existing work for this problem.

1 Introduction

Recently many studies have suggested that semi-supervised text clustering, which groups a collection of unlabeled text documents into clusters with a small amount of user provided information, is effective. The improvement on the clustering performance is mainly due to the incorporation of the user provided supervised data. Most traditional semi-supervised text clustering approaches generate queries for user feedback information in a passive manner. Therefore, it becomes crucial for a user to provide the most “valuable” information. However, it is not feasible for a user to browse all the text documents and select the most informative data. A solution to this problem is to let the clustering approach

play an active role in the process. Labeled information can be actively selected rather than chosen at random. Given a set of documents, users may have some criteria in mind for the underlying clusters. The grouping criteria may be different for different users according to the user preference or need. As shown in Figure 1, some users suggest d_1 , d_2 , and d_5 should be in the same cluster, d_3 and d_4 should be in another cluster. Some other users may suggest that d_1 , d_2 , and d_3 should be grouped together. Active learning offers a mechanism to select informative documents that best reveal the user grouping criteria. The user’s feedback on those selected documents contains the user’s opinion on the grouping criteria and offers good hints on guiding the clustering process for a better partition. Our goal is to select informative documents for obtaining user judgments so that the clustering performance can be improved with as few supervised data as possible. We consider pairwise constraints as the type of user provided supervised data to aid clustering. Pairwise constraints contain must-link or cannot-link information between two documents specifying that two documents must be in the same cluster or must be in different clusters respectively.

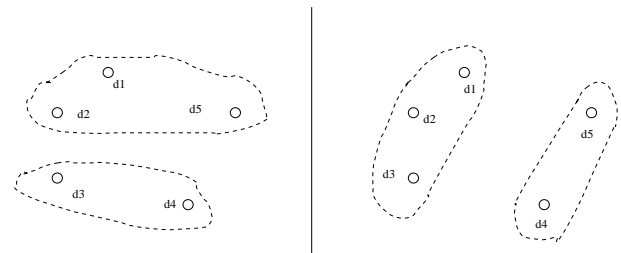


Figure 1: An example of different user criteria for the underlying clusters

Suppose that the true clustering assignments were known in advance. Those informative document pairs could then be easily chosen. In Figure 1, (d_1, d_5) is a useful document pair. If d_1 has a must-link constraint to d_5 , user may suggest the first clustering assign-

*The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Nos: CUHK 4179/03E and CUHK4193/04E) and the Direct Grant of the Faculty of Engineering, CUHK (Project Codes: 2050363 and 2050391). This work is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

ment rather than the second clustering assignment. In our framework, a preliminary clustering process is conducted to estimate the true clustering assignments. The intermediate cluster structure learned provides useful information for choosing informative document pairs. We investigate an approach to selecting informative document pairs automatically from the current cluster structure learned. A gain function is designed to measure how much information we can learn by revealing the judgments of document pairs. By formulating the gain function, the most informative document pairs are selected so that effective constraints can be generated. Two methods are designed for the gain function, namely, independent gain model and dependent gain model. In the independent gain model, we assume that the information learned by revealing the judgment of a document pair is independent of revealing the judgment of other document pairs. However, this design may not be effective in some situations. In an extreme case, the relative value obtained from two identical document pairs should be less than the value obtained from two different document pairs. Therefore, for the second method, we design a dependent gain model which considers previously chosen document pairs to avoid redundant selection and maximize the gain collectively for a set of document pairs.

We have conducted extensive experiments on our proposed approach comparing with a recent semi-supervised learning approach proposed by Basu et al. [2]. The experimental results demonstrate that our active learning framework is more effective.

2 Related Work

Semi-supervised clustering, which provides a way to incorporate the user knowledge or requirement in the form of constraints into the clustering process, is of great interest in recent years. In [19] [1], constraint-based methods which directly use constraints to guide the data partitioning are investigated. Sugota et al. [3] investigated a theoretically motivated framework for semi-supervised clustering that employs Hidden Random Markov Fields. Bilenko et al. [4] described an approach that unifies the constraint-based and metric-based semi-supervised clustering methods. In these methods, all the constraints are provided once before the clustering. All of these methods do not conduct clustering in an active feedback manner.

Active learning has been extensively studied in machine learning [7] [13] and has been applied to text classification. One of the important tasks of an active learning approach is to select informative samples to request the user to label it. Different principles of sample selection have been studied. Some

methods [9] [14] [18] choose the most uncertain samples which are closest to the current classification boundary. In [5], it is proposed to select the sample that yields the largest decrease of the margin between classes. In [7], it is suggested to select the sample that minimizes the expected future classification error. Some other methods combine clustering and active learning into text classification. In [10], a naive Bayes classifier is trained over both labeled and unlabeled data using an EM algorithm. For choosing query samples, some methods [22] [17] put emphasis on the uncertainty of samples. Some methods [20] [15] put emphasis on the representativeness of samples. Others [11] balance between these two factors. However active learning techniques in classification are not applicable in the clustering setting since some ideas used in sample selection criterion in classification context are not well-defined for clustering. In ad-hoc information retrieval area, Shen et al. [16] proposed a general framework for active feedback by defining the problem as a statistical decision problem. Based on the framework, several practical algorithms are derived.

The notion of “dependent selection” is also studied in information retrieval problems. In [6], a method for combining query-relevance with information-novelty in the context of text retrieval and summarization is proposed. In [21], a non-traditional retrieval problem called subtopic-retrieval, is presented. The subtopic retrieval problem is concerned with finding documents that cover as many different subtopics of a general topic as possible.

There has been little work on investigating active learning for semi-supervised clustering. In [2], Basu et al. proposed a two-step active learning scheme based on the farthest-first traversal. However, in this method, the active learning is only a part of the preprocessing phase of the clustering and is not embedded in the clustering process. Therefore, it cannot make good use of intermediate clustering result to select more informative query samples. Klein et al. [8] also considered active learning in semi-supervised clustering, but this method provides cluster level queries rather than instance level queries.

3 Overview of Our Active Learning Framework

We attempt to partition the unlabeled documents into a set of clusters with the help of a small amount of constraints, in particular, must-link constraints and cannot-link constraints. In order to improve the clustering performance with a limited number of constraints, a good strategy for selecting those “valuable” document pairs is needed. The problem of active learning is essentially a selection problem. Document pairs are automatically selected from the dataset to form queries. Judgments

are then made by the user on selected document pairs and constraints are generated by assigning must-link or cannot-link labels according to the user judgment for each document pair. The intermediate clustering result can be used to guide the document pair selection. A gain function is designed for choosing document pairs. This gain function measures how much we can learn from revealing the judgments of document pairs. In the dependent gain model, it also considers previously selected documents to maximize the gain collectively for a set of document pairs and avoid redundant selections.

The outline of our active learning framework is depicted in Figure 2. Solid rectangles and solid lines indicate process steps and process flow. Broken rectangles and broken lines depict data and data flow. For the first

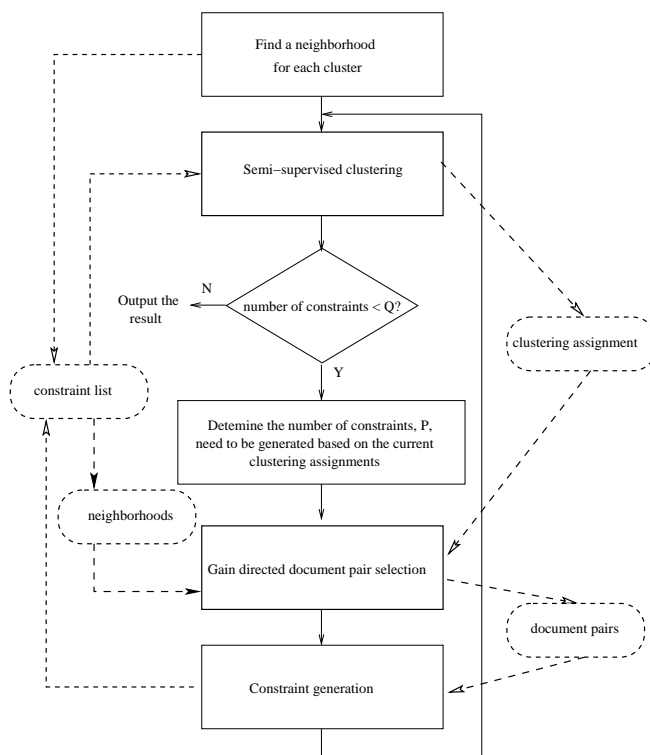


Figure 2: The outline of our active learning framework.

step of our framework, we employ a method similar to the “Explore” step in [2] for finding a skeleton structure of neighborhoods covering all the clusters using the farthest-first algorithm. We describe this step in the remaining content of this paragraph. A neighborhood is simply a set of documents within the same cluster. We make use of the pairwise constraints to generate the set of neighborhoods. Documents within the same neighborhood are must-link to each other. Any two documents in different neighborhood form a cannot-link constraint. In Figure 3, two neighborhoods are depicted.

The solid line represents a must-link constraint, and the broken line represents a cannot-link one. The farthest-first algorithm is adopted to find the set of neighborhoods for all the clusters at the beginning so that all the clusters have at least one representative document. Suppose K is the number of clusters. Initially, the set of neighborhoods are empty. The first document is picked randomly and added to the first neighborhood. When the number of neighborhoods has not reached to K and document pairs are allowed to be selected, a document d farthest from all the existing neighborhoods is selected. Document pairs are formed by combining d with a random document from each of the existing disjoint neighborhoods until a must-link is obtained. If a must-link is obtained for a particular existing neighborhood, d is added to that neighborhood. If all document pairs form cannot-link constraints, d forms a new neighborhood.

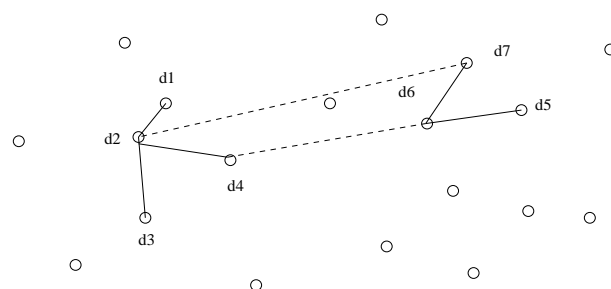


Figure 3: An example of neighborhoods

After the above step, a neighborhood should have been discovered for each cluster. A set of constraints is generated at the same time. Semi-supervised clustering and gain directed document pair selection are then conducted in an iterative manner until the number of constraints reaches a pre-specified value Q . For each iteration, P constraints are generated according to the current clustering assignment. In our current implementation, the number of iterations, θ , is determined first and the constraints are generated evenly in each iteration. In the gain directed document pair selection step, we attempt to calculate the value of a gain function for all possible document pairs and select the document pairs with the highest values. However, it is time consuming to compute the gain function for all possible document pairs in the dataset. The cluster assignments currently learned provide a clue for selecting document pairs. Instead of exploring all document pairs, we make use of the neighborhoods found in the previous step. Document pairs are formed by taking an unlabeled document d together with one of the documents in the neighborhood of the cluster that d currently belongs to. In the constraint generation step, queries are generated for the selected document pair on whether the two doc-

uments in the document pair should be in the same cluster or not. If the user judgment indicates that these two documents should be in the same cluster, a must-link constraint is assigned for this selected document pair. Otherwise, a cannot-link constraint is formed. When a cannot-link constraint is obtained, document pairs will be formed for d and other neighborhoods until a must-link constraint is generated. The document d is then involved in the neighborhoods. Next, the set of constraints are updated by the newly generated constraint and used in the subsequent semi-supervised clustering process. In the dependent gain model, the documents in the updated neighborhoods are also involved for selecting the most informative document pairs. The detailed description of gain directed document pair selection and constrained semi-supervised clustering are discussed in Section 4 and Section 5 respectively.

4 Gain Directed Document Pair Selection

As mentioned in Section 3, a major component in our framework is the design of gain directed document pair selection as shown in Figure 2. Our framework attempts to choose the best set of document pairs by considering the current clustering assignments discovered by the clustering process. Constraints are generated from the selected document pairs for directing the clustering process in the next round. To formalize this problem, we treat it as the following optimization problem:

$$(4.1) \quad \Omega^* = \arg \max \Lambda(\Omega, \Theta, \Phi)$$

where Ω is a set of document pairs; Θ is the current clustering assignments; Φ is the current set of constraints; $\Lambda(\Omega, \Theta, \Phi)$ is a gain function reflecting how much we can learn by revealing the judgments of the selected document pairs.

The selection of the set of document pairs depends not only on the current clustering assignments, or the current set of constraints, but also on the judgment of the document pairs. Given a pair of documents, two possible judgments can be assigned to the document pair by a user, namely j_m representing the judgment that two documents must be in the same cluster or j_c representing the judgment that two documents must be in different clusters. Let $J = \{j_m, j_c\}$ be the set of all possible judgments that a user may assign to a pair of documents, the gain function can be written as:

$$(4.2) \quad \Lambda(\Omega, \Theta, \Phi) = \sum_{\vec{J} \in J^P} g(\Omega, \Theta, \Phi, \vec{J}) p(\vec{J} | \Omega, \Theta, \Phi)$$

where $\vec{J} = \{j_1, \dots, j_P\}$ is a set of possible judgments for the set of document pairs and j_i is a possible judgment for the i -th document pair $\omega_i(d_1, d_2)$ in Ω ; P is the number of document pairs that can be selected from

the current clustering assignments; $g(\Omega, \Theta, \Phi, \vec{J})$ is a judgment gain function which indicates how much we can learn from the judgment \vec{J} of the document pairs; $p(\vec{J} | \Omega, \Theta, \Phi)$ is the probability that the judgment \vec{J} would be assigned to Ω .

We investigate two methods for the design of the gain function for selecting document pairs. The first method, called independent gain model, assumes that document pairs are selected independently. The previously selected documents in the current constraints do not affect the document pair selection in the later stage. The second method, called dependent gain model, attempts to select document pairs based on both the current clustering assignments and the current set of constraints. The following two subsections present the independent gain model and the dependent gain model in detail.

4.1 Independent Gain Model The independent gain model assumes that the document pairs are selected independently without considering the documents involved in the constraints. In other words, given a set of selected document pairs, users judge each document pair independently without considering the current clustering assignment and the current set of constraints. The $g(\Omega, \Theta, \Phi, \vec{J})$ and $p(\vec{J} | \Omega, \Theta, \Phi)$ in Equation 4.2 become as follows:

$$(4.3) \quad g(\Omega, \Theta, \Phi, \vec{J}) = \sum_{\omega_i(d_1, d_2) \in \Omega} g^{IG}(\omega_i(d_1, d_2), \Theta, j_i)$$

$$(4.4) \quad p(\vec{J} | \Omega, \Theta, \Phi) = \prod_{i=1}^P p(j_i | \omega_i(d_1, d_2))$$

where $g^{IG}(\omega_i(d_1, d_2), \Theta, j_i)$ is the independent judgment gain function for single document pair $\omega_i(d_1, d_2)$; $\omega_i(d_1, d_2)$ is the i -th document pair in Ω ; j_i is a possible judgment of document pair $\omega_i(d_1, d_2)$; $p(j_i | \omega_i(d_1, d_2))$ is the probability that the judgment j_i would be assigned to document pair $\omega_i(d_1, d_2)$.

Therefore, the goal of active learning for independent gain model is to select the optimal set of document pairs according to the following gain function:

$$(4.5) \quad \Omega^* = \arg \max \sum_{\omega_i(d_1, d_2) \in \Omega} \sum_{j_i} g^{IG}(\omega_i(d_1, d_2), \Theta, j_i) p(j_i | \omega_i(d_1, d_2))$$

This optimization problem can be viewed as a ranking problem which tries to select the top P document pairs with the highest value on the following pairwise gain function:

$$(4.6) \quad G^{IG}(\omega_i(d_1, d_2)) = \sum_{j_i} g^{IG}(\omega_i(d_1, d_2), \Theta, j_i) p(j_i | \omega_i(d_1, d_2))$$

Instead of ranking all possible document pairs, we make use of the set of neighborhoods to generate document pairs. This idea is formulated via a document gain function. Given the current clustering assignments, we select the document pairs from each currently discovered intermediate cluster. We calculate the value of the document gain function for all unlabeled documents in the dataset. The documents with the highest value are selected paired with one of the documents in the neighborhoods. The document gain function for the independent gain model, $G^{IG}(d)$, is derived from the pairwise gain function given in Equation 4.6. Specifically, $G^{IG}(d)$ is formulated as follows:

$$(4.7) \quad G^{IG}(d) = \sum_{j_d} g^{IG}(\omega_d(d, \mu_{\kappa_d}), j_d) p(j_d | \omega_d(d, \mu_{\kappa_d}))$$

where κ_d is the cluster to which the document d currently belongs; μ_{κ_d} is the centroid of the cluster κ_d and can be regarded as a pseudo-document representing the cluster κ_d ; $\omega_d(d, \mu_{\kappa_d})$ is a document pair formed by the document d and μ_{κ_d} ; j_d is the judgment of the document pair $\omega_d(d, \mu_{\kappa_d})$, in particular, either must-link j_m or cannot-link j_c indicating that d belongs to or does not belong to the cluster κ_d ; $p(j_d | \omega_d(d, \mu_{\kappa_d}))$ is the probability that the document pair $\omega_d(d, \mu_{\kappa_d})$ would be assigned the judgment j_d ; $g^{IG}(\omega_d(d, \mu_{\kappa_d}), j_d)$ is the independent judgment gain function which indicates how much we can learn from the judgment j_d on the document pair $\omega_d(d, \mu_{\kappa_d})$. We remove Θ from g^{IG} in Equation 4.6 and obtain $g^{IG}(\omega_d(d, \mu_{\kappa_d}), j_d)$ as each document d is selected from the cluster it currently belongs to.

We attempt to choose the document d that are not certain on the judgment j for the cluster to which d is currently assigned. The well-known entropy function can be employed to model this idea. Given two documents $\omega_i(d_1, d_2)$, the entropy $H(j_i | \omega_i(d_1, d_2))$ is expressed as follows:

$$(4.8) \quad H(j_i | \omega_i(d_1, d_2)) = \sum_{j_i} -\log(p(j_i | \omega_i(d_1, d_2))) p(j_i | \omega_i(d_1, d_2))$$

We regard a cluster centroid as a pseudo-document representing the cluster. Therefore, the entropy of the document-to-cluster judgment, $H(j | \omega(d, \mu_{\kappa_d}))$, is calculated as shown in Equation 4.9. The documents in each cluster are ranked by the descending order of this entropy.

$$(4.9) \quad H(j_d | \omega_d(d, \mu_{\kappa_d})) = \sum_{j_d} -\log(p(j_d | \omega_d(d, \mu_{\kappa_d}))) p(j_d | \omega_d(d, \mu_{\kappa_d}))$$

Comparing $H(j_d | \omega_d(d, \mu_{\kappa_d}))$ with the independent document gain function shown in Equation 4.7, we design

$g^{IG}(\omega_d(d, \mu_{\kappa_d}), j_d)$ for choosing the document for each cluster as follows:

$$(4.10) \quad g^{IG}(\omega_d(d, \mu_{\kappa_d}), j_d) = -\log(p(j_d | \omega_d(d, \mu_{\kappa_d})))$$

To estimate the probability that a document is must-link or cannot-link to a cluster, we make use of the similarity of the document to the cluster centroids. If the document d is uncertain on the current assignment, there must be at least one other cluster to which d is very likely related. Otherwise, if the similarity of d to the currently assigned cluster is dramatically larger than all the other clusters, d is very likely related to the current cluster. The probability that the document d and the assigned cluster is must-link or cannot-link is calculated as follows:

$$(4.11) \quad p(j_m | \omega_d(d, \mu_{\kappa_d})) = \frac{s(d, \mu_{\kappa_d})}{s(d, \mu_{\kappa_d}) + s(d, \mu_{\kappa'_d})}$$

$$(4.12) \quad p(j_c | \omega_d(d, \mu_{\kappa_d})) = 1 - p(j_m | \omega_d(d, \mu_{\kappa_d}))$$

where $\mu_{\kappa'_d}$ is the centroid of the cluster κ'_d to which the document d most likely belongs besides the cluster κ_d ; $s(d, \mu)$ is the cosine similarity between the document d and the cluster centroid μ and it is calculated as follows:

$$(4.13) \quad s(d_i, d_j) = \frac{\nu_{d_i} \nu_{d_j}}{\|\nu_{d_i}\| \|\nu_{d_j}\|}$$

where ν_{d_i} and ν_{d_j} denote the vector representations of the document d_i and the document d_j ; d_i and d_j can also be pseudo-documents such as the cluster centroid μ ; $\|\cdot\|$ denotes the L_2 norm.

4.2 Dependent Gain Model In the independent gain model, we assume each document pair is selected independently. Therefore, both the judgment gain function and the probability of the judgment assigned to a document pair do not consider the previously selected documents as shown in Equation 4.3 and Equation 4.4. The previously selected documents are involved in the constraints and the set of neighborhoods. Recall that we convert the set of constraints into a set of neighborhoods. However, this assumption may be too simplistic. For example, if a document pair is identical to the documents in the neighborhoods, the value gained by revealing its judgment is not valuable. Therefore, the set of constraints is useful for selecting document pairs and avoiding redundancy selection. We involve the current set of constraints, Φ , in the judgment gain function. The probability of the judgment for a set of document pairs is formulated as follows:

$$(4.14) \quad g(\Omega, \Theta, \Phi, \vec{J}) = \sum_{\omega_i(d_1, d_2) \in \Omega} g^{DG}(\omega_i(d_1, d_2), \Theta, \Phi, j_i)$$

$$(4.15) \quad p(\vec{J} | \Omega, \Theta, \Phi) = \prod_{i=1}^P p(j_i | \omega_i(d_i, d_j), \Phi)$$

where $g^{DG}(\omega_i(d_1, d_2), \Theta, \Phi, j_i)$ is the dependent judgment gain function for the single document pair $\omega_i(d_1, d_2)$; j_i is a possible judgment of the document pair $\omega_i(d_1, d_2)$.

The optimization problem can be viewed as a ranking problem which tries to select the top P document pairs with the highest value on the following pairwise gain function:

$$(4.16) \quad G^{DG}(\omega_i(d_1, d_2)) = \sum_{j_i} g^{DG}(\omega_i(d_1, d_2), \Theta, \Phi, j_i) p(j_i | \omega_i(d_1, d_2), \Phi)$$

Similar to the independent gain model, we make use of the neighborhoods found in the previous step to generate the pair of documents. A dependent document gain function is formulated as follows:

$$(4.17) \quad G^{DG}(d) = \sum_{j_d} g^{DG}(\omega_d(d, \mu_{\kappa_d}), \Phi, j_d) p(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$$

where κ_d is the cluster to which the document d currently belongs; μ_{κ_d} is the centroid of the cluster κ_d and can be regarded as a pseudo-document representing the cluster κ_d ; $\omega_d(d, \mu_{\kappa_d})$ is a document pair formed by the document d and μ_{κ_d} ; j_d is the judgment of the document pair, in particular, either must-link j_m or cannot-link j_c indicating that d belongs to or does not belong to the cluster κ_d ; $p(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$ is the probability that the document pair $\omega_d(d, \mu_{\kappa_d})$ would be assigned the judgment j_d ; $g^{DG}(\omega(d, \mu_{\kappa_d}), \Phi, j_d)$ is the dependent judgment gain function which indicates how much we can learn from the judgment j_d on the document pair $\omega_d(d, \mu_{\kappa_d})$. We remove Θ from $g^{DG}(\omega(d, \mu_{\kappa_d}), \Theta, j_d)$ as each document d is selected from the cluster it currently belongs to.

The entropy function $H(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$, which measures the uncertainty of the judgment for a document pair in the current clustering assignments considering the current set of constraints, is employed to model the dependent document gain function. The document d that is not certain on the judgment j_d for the cluster and is not similar with those documents involved in the neighborhoods is selected. The entropy $H(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$, is calculated as shown in Equation 4.18. The documents in each cluster are ranked by the descending order of this entropy.

$$(4.18) \quad H(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi) = \sum_{j_d} -\log(p(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)) p(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$$

Comparing $H(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi)$ with the document gain function shown in Equation 4.17, we design $g^{DG}(\omega_d(d, \mu_{\kappa_d}), \Phi, j_d)$ for choosing the document for

each cluster as follows:

$$(4.19) \quad g^{DG}(\omega_d(d, \mu_{\kappa_d}), \Phi, j_d) = -\log(p(j_d | \omega_d(d, \mu_{\kappa_d}), \Phi))$$

When a document d is not clear on the current assignment but is close to a certain document d' in the corresponding neighborhood, the judgments of the previously selected document d' provide useful clues for the judgment of document d . The assignment of d is clear and may be easily obtained in the subsequent clustering process. As the document d is near to d' and d' is certainly assigned to cluster κ_d , the probability that d has a must-link constraint to the currently assigned cluster κ_d is high. We formulate this idea as follows:

$$(4.20) \quad p(j_m | \omega_d(d, \mu_{\kappa_d}), \Phi) = \frac{s_d(d, \mu_{\kappa_d})}{s_d(d, \mu_{\kappa_d}) + s(d, \mu_{\kappa'_d})}$$

where $\mu_{\kappa'_d}$ is the centroid of the cluster κ'_d to which document d most likely belongs besides cluster κ_d ; $s(d, \mu)$ is the cosine similarity between the document d and the cluster centroid μ ; $s_d(d, \mu)$ is the dependent cosine similarity between the document d and the cluster centroid μ . $s_d(d, \mu)$ measures how similar a document is to a certain cluster given a set of documents involved in the corresponding neighborhood of the cluster. $s_d(d, \mu)$ is estimated as follows:

$$(4.21) \quad s_d(d, \mu) = (1 - \varepsilon) \frac{Q}{|D|} s(d, \mu) + \varepsilon \frac{Q}{|D|} \max_{d' \in n} s(d, d')$$

where n is the neighborhood associated with the cluster; Q is the number of constraints that can be posted in the semi-supervised learning; $|D|$ is the total number of documents in the document collection; ε controls the tradeoff between the similarity of the document d to the cluster centroid and to the closest previously selected document. When Q is small compared with $|D|$, there is a small amount of selected documents in the neighborhoods. The probability that a document is similar to a previously seen document is small. Therefore, the cluster centroid contributes more for calculating the similarity. When Q is large, there is a large amount of selected documents. It is relatively more important not to select a redundant document. Therefore, previously selected documents play an important role.

5 Constrained Semi-supervised Clustering

Our active learning framework involves the constrained semi-supervised clustering as shown in Figure 2. Pairwise constraints, updated by the gained directed document pair selection, are considered as the type of user feedback information to aid clustering. In this step, unsupervised clustering is employed to yield a better partitioning of the data according to the user feedback information.

Clusters are discovered by minimizing an objective function. The objective function is composed of two parts, namely, the document distance function considering the documents and the penalty function considering constraints. An optimal partition is obtained when the overall distance of the documents from the cluster centroids is minimized while a minimum number of constraints are violated.

The document distance function Υ measures the distance of the documents from the cluster centroids. This function is formulated as follows:

$$(5.22) \quad \Upsilon = \sum_{k=1}^K \sum_{d \in D_k} \psi(d, \mu_k)$$

where d is a document; μ_k is the centroid of the cluster k ; K is the number of clusters; D_k is the set of documents that belong to the cluster k ; $\psi(d, \mu_k)$ measures the distance between the document d to the cluster centroid μ_k . Since the amount of constraints is small compared with unlabeled documents, the document distance function Υ is mainly related to the unlabeled documents. The distance between two documents or between a document and a cluster centroid is calculated as follows:

$$(5.23) \quad \psi(d_i, d_j) = 1 - \frac{\nu_{d_i} \nu_{d_j}}{\|\nu_{d_i}\| \|\nu_{d_j}\|}$$

where ν_{d_i} and ν_{d_j} denote the vector representations of the document d_i and the document d_j ; d_i and d_j can be pseudo-documents such as the cluster centroid μ ; $\|\cdot\|$ denotes the L_2 norm.

The penalty function Δ is related to the constraints. It measures the cost of violation of the constraints. We denote the set of must-link constraints as M and the set of cannot-link constraints as C . We make use of the distance of the two documents in the constraints as the penalty cost for violating the constraints in M and C . The penalty function Δ is formulated as follows:

$$(5.24) \quad \Delta = \sum_{(d_i, d_j) \in M} \psi(d_i, d_j) \delta(\kappa_{d_i} \neq \kappa_{d_j}) + \sum_{(d_i, d_j) \in C} (1 - \psi(d_i, d_j)) \delta(\kappa_{d_i} = \kappa_{d_j})$$

where κ_d is the cluster to which d belongs; δ is the indicator function.

Consequently, the objective function Ξ is formulated as follows:

$$(5.25) \quad \Xi = \rho \Upsilon + (1 - \rho) \Delta$$

where ρ is a parameter for balancing the contribution of the document distance and constraints in the objective function. Therefore the semi-supervised clustering task

is to minimize the following objective function:

$$(5.26) \quad \Xi = \rho \sum_{k=1}^K \sum_{d \in D_k} \psi(d_i, \mu_k) + (1 - \rho) \left(\sum_{(d_i, d_j) \in M} \psi(d_i, d_j) \delta(\kappa_{d_i} \neq \kappa_{d_j}) \right) + \sum_{(d_i, d_j) \in C} (1 - \psi(d_i, d_j)) \delta(\kappa_{d_i} = \kappa_{d_j})$$

The optimization for finding the clusters minimizing the objective function expressed in Equation 5.26 is achieved by an iterative process. The outline of the algorithm is presented in Figure 4.

-
- 1 Initialization: the centroid of each cluster is initialized.
 - 2 Repeat until convergence
 - 3 Assign_cluster: Given the centroid for each cluster, update the document assignment to clusters.
 - 4 Estimate_means: Given the current cluster assignment, re-calculate the centroid of each cluster.
-

Figure 4: The constrained semi-supervised clustering algorithm

The cluster centroids are initialized by the set of neighborhoods. In the Assign_cluster step, the assignments of the documents to the clusters are updated. Each unlabeled document is assigned to the cluster that minimizes the objective function. In the Estimate_means step, every cluster centroid μ_k is re-estimated using the current document assignment as follows:

$$(5.27) \quad \mu_k = \frac{\sum_{d \in D_k} d}{\|\sum_{d \in D_k} d\|}$$

where D_k is the set of documents that belong to the cluster k .

6 Experimental Result

6.1 Datasets Two real-word text corpora were used for conducting extensive experiments. The first corpus is derived from the 20-Newsgroups collection¹. This corpus contains text messages from 20 different Usenet newsgroups, about 1,000 messages from each newsgroup. For the first dataset, following the ones used in Basu et al. [2], we first created a subset which has 100 messages from each of the 20 newsgroups. Two datasets, namely *Small-News-Similar-3*

¹The description of the 20-Newsgroup corpus can be found at <http://people.csail.mit.edu/jrennie/20Newsgroups>

and *Small-News-Different-3*, were then derived from the subset by selecting newsgroups with different level of similarity. *Small-News-Similar-3* consists of 300 messages from 3 similar newsgroups (comp.graphics, comp.os.ms.windows.misc, and comp.windows.x) where cross-posting occurs often. *Small-News-Different-3* consists of 300 messages posted by 3 newsgroups of quite different topics (alt.atheism, rec.sport.baseball, and sci.space). Since we also want to conduct experiments in large scale, another two datasets, namely *News-Similar-3* and *News-Different-3*, were generated using the original full set of 20-Newsgroup corpus. *News-Similar-3* contains 2938 messages from the 3 similar newsgroups. *News-Different-3* contains 2780 messages from the 3 different newsgroups.

The second dataset was derived from Reuters RCV1 corpus [12]. The RCV1 corpus is an archive of over 800,000 manually categorized newswire stories. The news stories are organized differently based on three category codes, in particular, topics, industries, and regions. We derived a dataset from the RCV1 based on the region category code. We selected those large categories which contain a large number of news stories. 2,000 news stories were selected randomly which are grouped into 12 region categories.

6.2 Evaluation Metric The pairwise F-measure ζ similar to the one used in [2] is used for the evaluation metric. It is defined as a combination of recall R and precision P for the document pairs without user provided constraints as follows:

$$(6.28) \quad P = n_c/n_s$$

$$(6.29) \quad R = n_c/n_f$$

$$(6.30) \quad \zeta = \frac{2 \times P \times R}{P + R}$$

where n_c is the number of document pairs that are correctly predicted as in the same cluster; n_s is the number of document pairs that are predicted as in the same cluster; n_f is the number of document pairs that are actually in the same cluster.

6.3 Experimental Setup We make use of the set of manually annotated cluster assignments to simulate the user judgments on the proposed document pairs chosen by active learning models. If the documents belong to the same user labeled cluster, a must-link constraint is assigned to the document pair. Otherwise, a cannot-link constraint is assigned.

To determine the value of the parameters of our models, we conducted a parameter tuning process. We varied different settings of the parameters using a small dataset. The set of parameters which achieve the best performance are chosen for conducting the full-scale ex-

periments. The dataset used for the tuning process is *Small-News-Similar-3*. The number of constraints is set as 200. Both independent gain and dependent gain models are investigated. In our framework, there are three parameters to be tuned. The first parameter is the number of iterations, θ , mentioned in Section 3. Figure 5 depicts the performance for different θ . The best performance is obtained for both dependent and independent gain model when θ is set to 8. When the number of iteration is increasing, the performance improves at the beginning. The reason is that with better cluster assignments, document pairs can be chosen more precisely. After 8 iterations, the performance starts to decrease because of the limited number of constraints allowed to be generated. With too many iterations, the number of constraints that can be generated in each iteration is too little to obtain useful information from the current clustering assignments. The second parameter

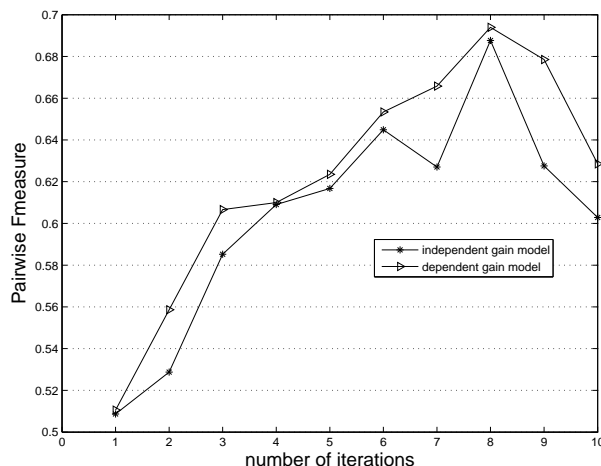


Figure 5: Parameter tuning for the number of iterations θ

is ρ in Equation 5.25. We find that ρ is not sensitive to the clustering performance. The clusters discovered are more or less stable with different values of ρ . We decide ρ to be 0.1 after the tuning process. The third parameter is ε in Equations 4.21. Figure 6 depicts the performance of clustering performance with different values of ε from 0.1 to 1. We conducted experiments only with the dependent gain model since ε is related to the dependent cosine similarity in the dependent gain model. The best performance is obtained when ε is equal to 0.5. As a result, these tuned parameters are used in the full-scale experiments.

We conducted experiments for our framework including both the independent and dependent gain mod-

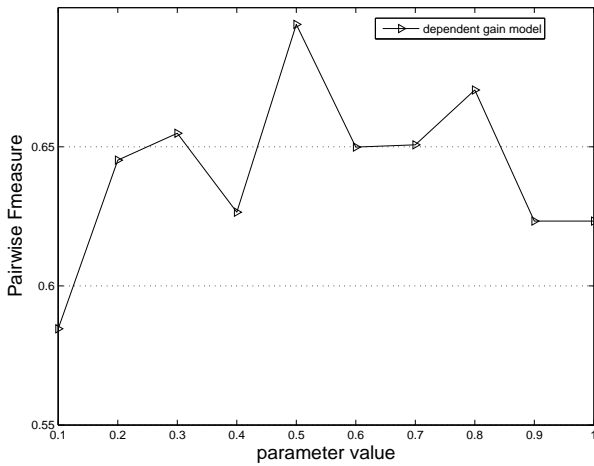


Figure 6: Parameter tuning for the parameter ε

els². We investigated the clustering performance by varying the number of constraints. For comparative investigation, we also run experiments for the recent active learning method proposed by Basu et al. [2], labeled as explore-consolidate. The constrained semi-supervised clustering which does not actively learn the constraints is treated as the baseline. One each dataset, 5-fold cross-validation was conducted. The pairwise constraints are generated from the training portion of the dataset. We make use of a sample size S for choosing documents for document pairs. For each document pair, S documents are chosen from the dataset randomly and are ranked based on the document gain function. S was set to 300 for the *Small-News-Different-3* and the *Small-News-Similar-3* dataset, and was set to 500 for RCV1, *News-Different-3*, and *News-Similar-3* datasets.

6.4 Performance of the Our Models From Figure 7 to Figure 11, we present experimental results on our proposed independent and dependent gain models. Figure 7 and Figure 8 depict the results for the *Small-News-Similar-3* and the *Small-News-Different-3* datasets. Figure 9 and Figure 10 depict the results for the *News-Similar-3* and the *News-Different-3* datasets. Figure 11 depicts the results on the RCV1 dataset.

The experimental results demonstrate that our proposed independent gain model and dependent gain model are effective for improving the clustering performance. Both the independent and dependent gain models outperform the baseline model and the recent existing work of Basu et al. [2] (explore-consolidate). The dependent gain model generally achieves the best per-

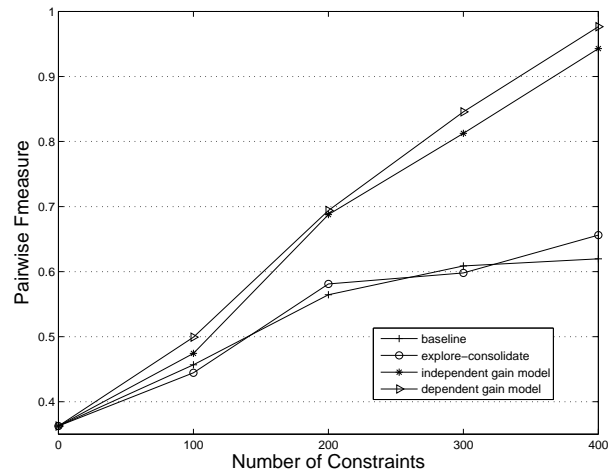


Figure 7: Clustering performance measured by pairwise F-Measure for our models on the *Small-News-Similar-3* dataset

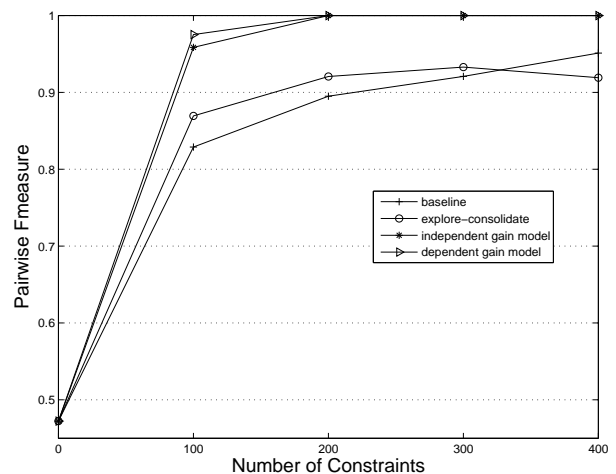


Figure 8: Clustering performance measured by pairwise F-Measure for our models on the *Small-News-Different-3* dataset

²The experiments were conducted on a 3.6 GHz PC with 2 GB memory. The operating system is Linux Fedora Core 4.

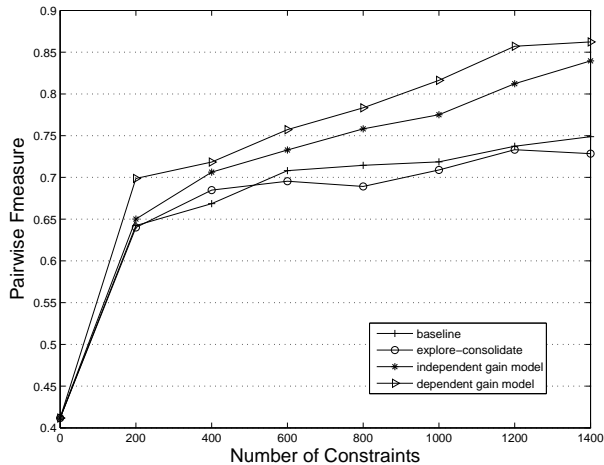


Figure 9: Clustering performance measured by pairwise F-Measure for our models on the *News-Similar-3* dataset

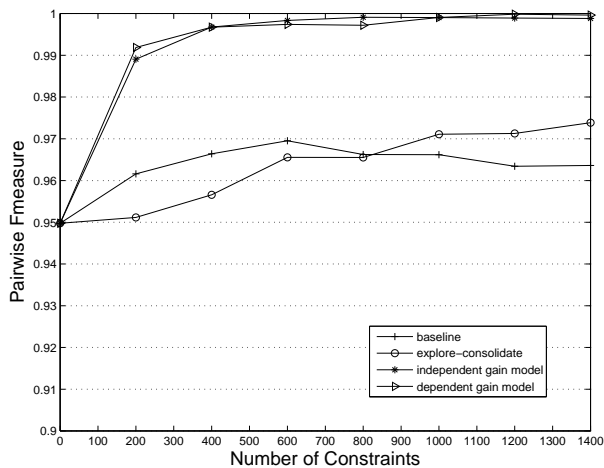


Figure 10: Clustering performance measured by pairwise F-Measure for our models on the *News-Different-3* dataset

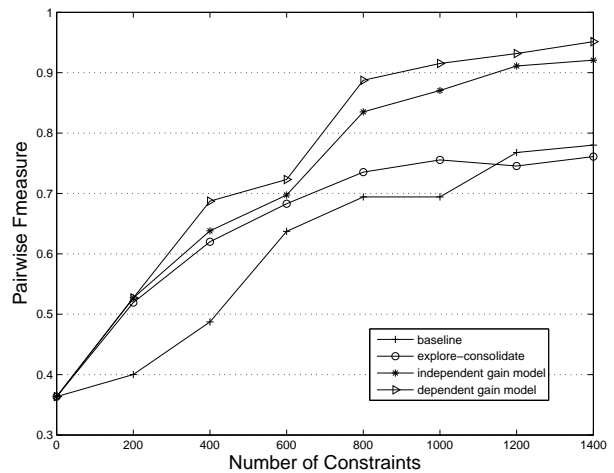


Figure 11: Clustering performance measured by pairwise F-Measure for our models on the RCV1 dataset

formance. For the *Small-News-Different-3* and *News-Different-3* datasets, our proposed models almost reach to a perfect result. The intermediate clustering assignment is helpful for selecting document pairs. The clustering performance can be improved dramatically with the constraints generated from informative document pairs actively selected. Compared with the baseline model, the improvement of our models is obvious in all range of number of constraints. Compared with the explore-consolidate, the improvement is more and more dramatically when the number of constraints is increasing. The dependent gain model generally performs better than the independent gain model. The improvement is more obviously with a large number of constraints. The reason is that documents are more likely to be redundant when there are a large amount of previously selected documents.

6.5 Discussions The number of constraints does not equal to the number of documents involved in the neighborhoods. In an optimal situation, given V documents in the dataset and K clusters, a user needs to indicate either V different class labels to the documents or $V-K$ must-link constraints and $((K-1)(K-2))/2$ cannot-link constraints to provide a perfect clustering manually. However, in our problem, it is infeasible to find the related cluster for each selected document to generate pairwise constraints at the first time. In the worst case, $K-1$ constraints are generated to involve a document into the set of neighborhoods. As shown in Table 1, the percentage of the documents involved in the neighborhoods is small. Compared with the whole dataset, users only provide a small amount of information especially in *News-Similar-3* and RCV1 datasets. In the *News-*

Different-3 corpus, the percentage of the documents in the neighborhood is larger compared with two other corpora. The reason is that the selected documents usually have correct assignments. Therefore, it does not generate many cannot-link constraints. However, the clustering performance of *News-Different-3* reaches to 0.99 with 200 pairwise constraints under the dependent gain model. The percentage of the number of documents involved in the neighborhoods is only 4.3%.

number of constraints	<i>News-Similar-3</i>	<i>News-Different-3</i>	RCV1
200	3.4%	4.3%	2.4%
400	6.5%	10.7%	4.5%
600	9.8%	17.8%	7.2%
800	13.8%	25.1%	9.4%
1000	17.9%	31.6%	13.8%
1200	21.7%	38.9%	16.9%
1400	25.9%	45.7%	21.1%

Table 1: The percentage of the documents involved in the neighborhoods

We also compared the computational time required for our proposed dependent gain model and the explore-consolidate model [2] with 400 constraints as shown in Table 2. Our dependent gain model requires more time than the explore-consolidate model since document pair selection and semi-supervised clustering are conducted in an interactive manner. Nevertheless, the execution time of our dependent gain models is still satisfactory in practice. The computational time should be worthwhile in exchange for significantly improved performance. For example, as shown in Figure 11, our dependent gain model achieves 10.8% increase in performance over the explore-consolidate model.

	<i>News-Different-3</i>	<i>News-Similar-3</i>	RCV1
explore-consolidate	26	28	21
dependent gain model	229	240	120

Table 2: The computational time (in second) for our dependent gain model and the explore-consolidate model

7 Conclusions and Future Work

We have presented a semi-supervised text clustering approach that actively selects informative document pairs from the intermediate clustering results for user feedback. The user judgments are used to generate constraints for guiding the clustering process in the next round. A gain function is designed for choosing “valuable” document pairs automatically by measuring

how much we can learn by revealing the judgment of document pairs. Two methods have been developed, namely, independent gain model and dependent gain model. Experimental results show that our proposed active learning approach is effective. The clustering performance is dramatically improved with the updated constraints generated. Our approach also outperforms the recent method mentioned in [2].

There are several directions for further study. One direction is to incorporate the document distribution into the gain function. In our current framework, we attempt to choose document pairs which are uncertain on the cluster assignment. However, another factor which affects the clustering performance is the density of the documents in the dataset. The documents can be weighed by the document density. A selected document with low density is not indicative to the other documents. The worst case is that the constraint generated from a selected document pair is only useful for specific documents. On the contrary, a selected document with high density may be more useful for guiding the clustering process since it provides useful information for a large number of unlabeled documents.

Another direction is to discover the hierarchical relationship of the documents. Currently, we consider pairwise constraints as the user provided information. It is feasible for users to provide hierarchical constraints for indicating the parent-to-child relationships between documents. This information can be used to guide hierarchical clustering. The hierarchical constraints can be obtained randomly as a preprocessing process before clustering, and can also be actively learned from the currently discovered hierarchical structure.

References

- [1] S. Basu, A. Banerjee, and R. Mooney. Semi-supervised clustering by seeding. In *Proceedings of the Nineteenth International conference on Machine Learning*, pages 27–34, 2002.
- [2] S. Basu, A. Banerjee, and R. J. Mooney. Active semi-supervision for pairwise constrained clustering. In *Proceedings of the SIAM International Conference on Data Mining*, pages 333–344, 2004.
- [3] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 59–68, 2004.
- [4] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the International Conference on Machine Learning*, pages 81–88, 2004.
- [5] C. Campbell, N. Cristianini, and A. Smola. Query learning with large margin classifiers. In *Proceedings of*

- the *Seventeenth International Conference on Machine Learning*, pages 111–118, 2000.
- [6] Jaime Carbonell and Jade Coldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336, 1998.
- [7] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.
- [8] D. Klein, S. D. Kamvar, and C. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 307–314, 2002.
- [9] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the Seventeenth ACM International Conference on Research and Development in Information Retrieval*, pages 3–12, 1994.
- [10] A. K. McCallum and K. Nigam. Employing EM in pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358, 1998.
- [11] H. T. Nguyen and A. Smeulders. Active learning using pre-clustering. In *Proceedings of the 21st International Conference on Machine Learning*, pages 623–630, 2004.
- [12] T. Rose, M. Stevenson, and M. Whitehea. The Reuters Corpus Volume 1— from yesterday’s news to tomorrow’s language resources. In *Proceedings of Third International Conference on Language Resources and Evaluation*, pages 29–31, 2002.
- [13] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the International Conference on Machine Learning*, pages 441–448, 2001.
- [14] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 839–846, 2000.
- [15] S. Shen and C. Zhai. Active feedback - UIUC TREC-2003 HARD experiments. In *The Twelve Text Retrieval Conference, TREC*, pages 662–666, 2003.
- [16] X. Shen and C. Zhai. Active feedback in ad hoc information retrieval. In *Proceedings of the 28th ACM International Conference on Research and Development in Information Retrieval*, pages 59–66, 2005.
- [17] M. Tang, X. Luo, and S. Roukos. Active learning for statistical natural language parsing. In *Proceedings of the Association for Computational Linguistics 40th Anniversary Meeting*, pages 120–127, 2002.
- [18] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proceedings of the Ninth ACM International Conference on Multimedia*, pages 107–118, 2001.
- [19] K. Wagstaff and C. Cardie. Clustering with instance-level constraints. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 1103–1110, 2000.
- [20] Z. Xu, K. Yu, V. Tresp, X. Xu, and J. Wang. Representative sampling for text classification using support vector machines. In *Proceedings of the 25th European Conference on Information Retrieval Research*, pages 393–407, 2003.
- [21] Chen Zhai, William W. Cohen, and John Lafferty. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 10–17, 2003.
- [22] C. Zhang and T. Chen. An active learning framework for content-based information retrieval. *IEEE Trans. on Multimedia*, 4:260–268, 2002.