Diversity Measurement for the Email Content of Information Workers

by

Petch Manoharn

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

August 15, 2006

[September 2006]

Author____

Department of Electrical Engineering and Computer Science
August 15, 2006

Certified by____________________________________

Erik Brynjolfsson, Thesis Supervisor

Certified by____________________________________

Sinan Aral, Thesis Co-Supervisor

Certified by____________________________________

Marshall Van Alstyne, Thesis Co-Supervisor

Accepted by____________________________________

Arthur C. Smith
Chairman, Department Committee on Graduate Theses

Diversity Measurement for the Email Content of Information Workers

by
Petch Manoharn

Submitted to the
Department of Electrical Engineering and Computer Science

August 15, 2006

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

Since the introduction of computers and the Internet, information processing has evolved to be a major part of many businesses, but factors that contribute to the productivity of information workers are still understudied. Some social network studies claim that diverse information that passes through workers in the positions with diverse sources of information drives performance. However, such claims are rarely verified by empirical data. This study develops a measurement methodology for the diversity of the email content processed by information workers. The diversity values will be used for future productivity studies along with the results from social network analysis.

Erik Brynjolfsson
George and Sandi Schussel Professor of Management and Director, Center for Digital Business, MIT Sloan School of Management
Thesis supervisor

Sinan Aral
PhD Candidate, Sloan School of Management
Thesis co-supervisor

Marshall Van Alstyne
Associate Professor, Boston University School of Management
Visiting Professor, MIT Sloan School of Management
Thesis co-supervisor

This research was funded by the National Science Foundation under grant number IIS-0085725.

Contents

1. Introduction

Since the introduction of computers and internet, information processing has become a significant part of everyday life. Many successful companies have devoted their resources to information gathering and classification. Some companies even base their entire business around the various applications of information. Despite the significance of information the study of productivity in information-related sectors is still underdeveloped. In the manufacturing sector, economists have proposed many established models for production functions and productivity measurements, but one can hardly find a measurement for productivity of information workers such as lawyers, teachers, and so on.

Some recent studies have been conducted on the social network structure of information workers in order to understand the flow of information between people. The social network studies have found a correlation between social network structure and the productivity of information workers (Aral et al., 2006). Several theoretical arguments claim that access to diverse information that flows through and is exposed to workers in diverse structural positions drives the productivity of the workers. However, the claim is rarely verified by empirical data. An area that requires additional attention is the content analysis of the information flowing through social networks. While the content represents the actual information that is being communicated, the interpretation of the content is usually subjective and is hard to examine. The study of social networks concentrates on the flow of information, but the relationship between the flow of information and the content of information remains understudied.

The objective of our study is to provide a systematic method for measuring the diversity of the content of information exposed to information workers. Specifically, the information is represented by the emails circulated among the information workers as email represents a large portion of the information passed through most companies. The applications of the diversity measurement are not restricted to use in emails or for information workers. They can also be applied toward any set of text documents that require measurement of the amount of information contained in some subsets of the documents.

Since the diversity of information is a subjective concept, this study will not aim toward finding a universal measurement for diversity. Diversity metrics will be derived from many aspects or interpretations of diversity, and the results of the diversity metrics will be compared and are likely to be applied toward different occasions. Ideally, our diversity metrics should provide the results that behave similarly in diversity rankings. The diversity measurements will then be examined along with the results from social network study and the productivity data in order to determine the relevance of content diversity for the productivity of the information workers.

2. Theory and Literature

2.1 Data Representation and Similarity Coefficients

The most common way of modeling the content of text documents is the Vector Space Model (Salton, Wong, & Yang, 1975). The content of a document is represented by a multi-dimensional feature vector whose elements are commonly related to the term frequencies of the words in the document. The decision for the construction of the feature vectors is called term weighting. Term weight consists of three elements based on single term statistics: term frequency factor, collection frequency factor, and length normalization factor. The term frequency factor is usually based on the term frequencies in the document. The collection frequency factor is based on the frequencies of the term across documents in the collection of documents. The collection frequency factor provides a way to prioritize rare words or terms that do not appear often in all documents. The length normalization factor involves the length of the document in order to compensate for the fact that more words or terms are likely to appear in long documents.

A common weighting scheme is to use the term frequencies as described by Luhn (1958). The term frequency is usually referred to as content description for documents and is generally used as the basis of a weighted document vector. It is possible to use a binary document vector to represent the content of a document by associating the value one with the presence of a term in the document and the value of zero with the absence of the term. However, the results from using binary document vectors are not as good as results from vectors of term frequencies in the vector space model (Salton & Buckley, 1996). Many term-weighting schemes have been evaluated for the indexing of documents for search purpose.

Specifically, the TF-IDF weighting scheme (term frequency – inverse document frequency) is shown to provide an improvement in precision and recall (Salton & Buckley, 1996). The main concept of the TF-IDF weighting scheme is to provide high weights to terms that appear frequently in the document and infrequently across the documents in the collection. A similar concept is used by Amazon.com's Statistically Improbable Phrases (SIPs)[1] that automatically identify important phrases in each book in order to distinguish the book from other books with similar content. In this study, we will use the common weighting scheme based on the term frequencies because the effectiveness of the other weighting schemes is still unproven for our application. Future improvements will include the evaluation of other weighting schemes.

Based on the vector space model, many previous studies have been conducted on document similarity. The most common similarity coefficient is the cosine similarity (Rasmussen, 1992), which represents the similarity of two documents by the cosine of the angle between their feature vectors. There are other similarity coefficients that can be used as alternatives to cosine similarity such as Jaccard and Dice coefficients (Salton, 1988). Our study adapts these similarity coefficients to use as distance measurements for feature vectors to derive diversity measurements.

In addition to similarity coefficients for the vector space model, similarity coefficients have also been developed for concepts in taxonomy. Most of these coefficients are based on the information theoretic concept of 'information content' and the relationship between information content and similarity coefficients. The information content or self-information of a concept is usually defined by the amount of information the concepts added to someone's knowledge, and is normally expressed in the unit of information – a bit. It is defined as the negative of logarithm of the probability of the concept. Resnik (1995) proposes a measurement based on information content for documents in taxonomy, and Lin (1998) provides a generalized measurement of similarity, which is normalized to values between zero to one.

[1] The definition of the Statistically Improbably Phrases provided by Amazon.com is at http://www.amazon.com/gp/search-inside/sipshelp.html.

Zipf (1949) found that in any comprehensive text there is a hyperbolic relationship between the word frequency and rank of a word. The rank is determined in the order of high frequency to low frequency. For example, the most frequent word is of rank 1 and the second most frequent word is of rank 2 and so on. His observation implies that the product of the rank and the frequency is approximately constant for all words. This empirical finding is now known as Zipf's law. In general English text, the most frequent word is "THE", followed by "OF", and "AND". Zipf's law implies that the frequency of "THE" is approximately twice the frequency of "OF" or three times of the frequency of "AND". We see similar findings in our data set as presented and discussed in Appendix C.

2.2 Complexity Measurement

A closely related problem to diversity measurement is complexity measurement. One could argue, for example, that the complex email content implies greater content diversity. However, the veracity of this claim is empirical by nature. One measurement of complexity is the Kolmogorov complexity, also known as descriptive complexity (Li & Vitányi, 1997). It states that the complexity of an object is a measure of the resources needed to specify that object. For example, the number "1,000,000,000,000,000,000,000,000,000,000" can be described as "10^30" which is less complex than a number like "1,492,568,437,509,452,545,761,347,489" which is not easily described by any short description. With this logic, a good descriptive representation of email contents can be used as diversity measurement. However, we currently lack the model for the representation, so we will not apply Komogorov complexity in this study.

2.3 Data Clustering

One way to categorize information into manageable categories is to apply data clustering to the information. In the vector space model, the contents of documents are represented as feature vectors, and thus can be clustered into groups or categories. A common and efficient method of data clustering is called iterative clustering. This method produces clusters by optimizing an objective function defined either locally or globally by iteratively improving the result of the objective function. The most common objective function is the mean squared distance function. One of the most commonly known iterative clustering algorithms is the k-means clustering algorithm. The k-means algorithm is based on the relocation of documents between k clusters in order to locally optimize the mean squared distance function. The main concept of the k-means algorithm is as follows:

1. Divide documents into k clusters by a random initialization, and compute the centers of all clusters.
2. Reassign each document to the cluster whose center is the closest to the document.
3. Re-compute the cluster centers using the current assignment of documents.
4. If the stopping criterion is not met, go to step 2.

The typical stopping criteria are that there is no change in the assignment of documents to clusters or that the value of the objective functions falls below a pre-defined threshold. Once there is no document re-assignment, the algorithm has reached a local optimum of the objective function. The performance of the clustering is not easy to evaluate because the k-means algorithm is very sensitive to the initialization. If the initial assignment has not been properly chosen, the resulting clusters will converge toward a suboptimal local optimum (Duda & Hart, 1973).

This study utilizes output from a clustering algorithm called eClassifer, which was used in a previous stage of our study to classify topics in our email corpus. eClassifier is developed by IBM Almaden Research Center to be a semi-automatic tool for clustering unstructured documents such as problem ticket logs from a computer helpdesk. It uses the vector space model by constructing feature vectors from the documents by analyzing term frequencies with the use of a stop-word list, include-word list, synonym list, and stock-phrase list. The resulting terms are then examined by the user, and any modification can be made in order to guarantee the quality of the keywords. Once the feature vectors are constructed, the software applies a k-means clustering algorithm to cluster the documents. In order to solve the weakness of the k-means algorithm converging to local optimum, the user can reassign documents from the resulting assignment. The user can also merge clusters together or split a cluster into many smaller clusters. The k-means algorithm can then be applied in order to achieve a better assignment. eClassifier depends on some user interventions along every small step to overcome the weakness of the common automated algorithms used in clustering.

3. Background and Data

3.1 Background

The data used in this study is collected from a medium-sized executive recruiting firm over five years (Aral et al., 2006). The firm is headquartered in a large mid-western city and has thirteen regional offices in the United States. It consists of employees occupying one of the three basic positions – partner, consultant, and researcher. While the projects of the firm vary in details and complexities, they have a similar goal – to find and deliver suitable candidates with specific qualifications for upper-level executive positions requested by clients. The process of selection follows a standard guideline. A partner secures a contract with a client and assembles a team to work on the project. The team size ranges from one to five employees with the average team size of two, and the assignments are based on the availabilities of the employees. The project team identifies potential candidates based on the requested positions and specifications, and ranks them by their match with the job description. Based on the initial research, the team conducts internal interviews with potential candidates. After detailed evaluations, the team presents the final list of approximately six qualified candidates to the client along with detail background information. The client can then interview the candidates and make offers to one or more candidates if satisfied. In each project, the client has specific requirements about the skills and abilities of the candidates. In order to complete a contract, the search team must be able to present candidates who meet the minimum requirements of the client, and the candidate quality should satisfy the client.

The executive search process requires a significant amount of researching, and it is likely to involve a lot of information about the specific position and the qualifications of the potential candidates. Team members acquire information about potential candidates from various sources such as the firm's internal proprietary database of resumes, external proprietary databases, other employees in the firm, and other public sources of information. The team relies on the gathered information in order to make decisions during the process. The information exchanges within the firm also pay a significant role in helping search teams locate potential candidates that match the client's need.

3.2 Data

3.2.1 Email Data Set

We have acquired four data sets related to the operation of the firm – three data sets from the firm and one from outside the firm. The first data set is detailed internal accounting records regarding revenues, costs, contracts, project duration and composition, positions of the employees, etc. This data was collected during over five years of operation and included more than 1,300 projects. The second data set is the survey responses about information seeking behaviors such as experience, education, and time allocation. Due to the incentive for completing the survey, participation exceeded 85%. This information helps us establish the backgrounds of the employees. The third data set is a complete email history captured from the corporate mail server during the period from August 2002 to February 2004. The fourth data set is various independent controls for placement cities. This information allows normalization for the differences in the natures of different projects.

The data set that we are interested in for this thesis is the captured emails. The email data set consists of 603,871 emails that are sent and received by the participating employees of the firm. The contents of the emails are hashed to allow further studies of the email contents while preserving the privacy of the firm and the employees. Prior to the content hashing, a study has been conducted to perform data clustering on the contents of the emails into multiple groups that we will call "buckets" by using eClassifier. Since the clustering is performed on the original emails while the real contents can be verified, the bucket information is evaluated by human observers of the classification process and is therefore likely to be quite accurate. Due to time limitations clustering was performed on a large subset of the data, and therefore 118,185 of the 603,871 total emails contain bucket information. The further detail of the email data set is described in Appendix B, and the full description of the whole data set is discussed by Aral et al. (2006).

3.2.2 Wikipedia.org Data Set

In order to conduct a sensible evaluation of our diversity measurements, we need to establish a systematic test that is amenable to verification. However, due to privacy protection, the contents of the emails in the email data set are encrypted. Since we do not have access to the readable contents, we are unable to directly evaluate the accuracy of the diversity measurements on readable content in the email data set itself. Therefore we validate our diversity measurement methodology by acquiring and testing our approach on another data set with readable contents – a collection of 291 articles from Wikipedia.org. Conveniently, the articles have been categorized in a hierarchical tree-like category structure with three major categories and 25 subcategories. As with most encyclopedic corpuses, this type of structure groups documents or entries in successively more detailed sub-categorizations. For example, a given Meta topic may catalogue documents on "Computers," subcategorized into topics such as "Computer Hardware," "Computer Software, and "Computer Peripherals" etc. This hierarchical structure enables us to compile groups of documents selected ex ante to contain either relatively similar topics or relatively dissimilar topics from diverse fields. For example, one grouping may contain articles from computing, biology, painting, and sports while another grouping contains the same number of articles from computing alone. We can then apply our diversity measurement methodology to the data and see whether our measures reliably characterize the relative diversity of the information contained in groups of documents that have been chosen ex ante as either 'diverse' or 'focused.' The detail of the Wikipedia.org data set appears in Appendix A.

3.2.3 Results from eClassifier

The setting for eClassifier output used in this study is to apply semi-automated clustering several times on the same set of emails using different numbers of resulting clusters. Specifically, the clustering is applied 11 times with the following numbers of clusters: 2, 3, 4, 5, 8, 16, 20, 21, 50, 100, and 200. Figure 1 demonstrates the results of the clustering. The results form a hierarchy-like structure as we may expect from a manual division of the emails based on content into categories and subcategories. However, all clustering is performed on the whole set of emails, not on individual clusters from the previous clustering, so the emails in a cluster are not derived solely from a single cluster in the previous clustering. Therefore, we are unable to assume that the clusters form a perfectly hierarchical tree structure.

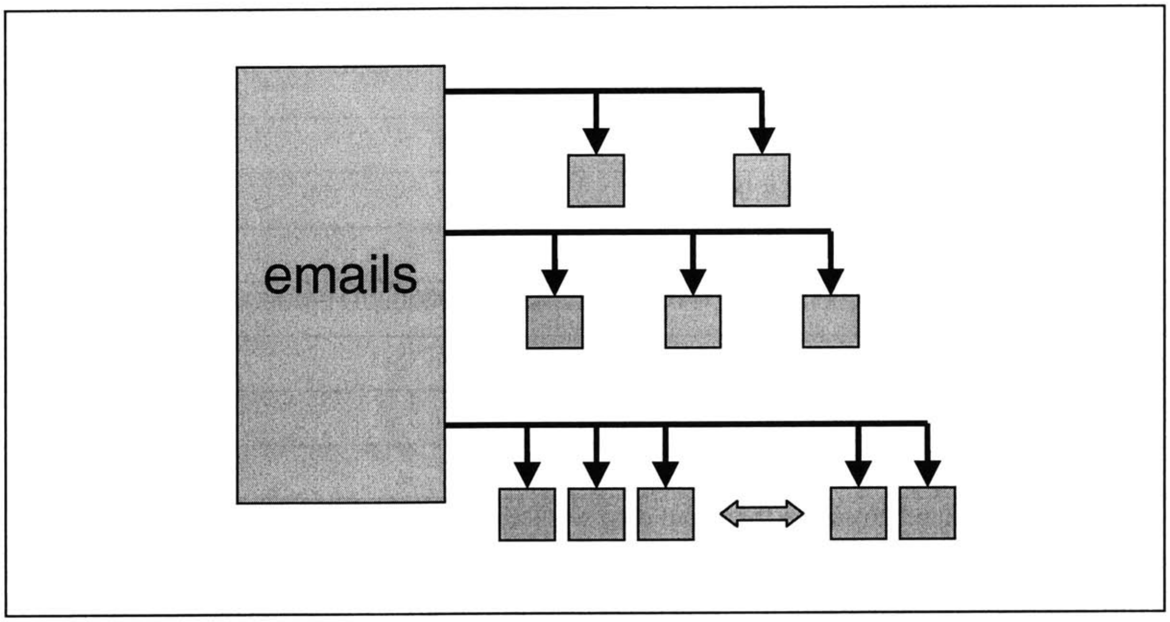


Figure 1: A hierarchical structure of the clustering results

4. Methods

4.1 Data Processing

The Email data set contains over six hundred thousand emails. However, there exist some duplicated emails with the same sender, recipients, timestamp, and content. The duplicated emails sometimes possess different unique identification numbers, so they are identified as being different in the data set. We eliminated the duplicated emails by removing emails with duplicated sender, recipients, and timestamp. Additionally, there are duplicated emails that are not entirely the same. One email may have fewer recipients than another email, but the sender, timestamp, and the content are the same as shown in Figure 2. Only one copy of these duplicated emails is included in the analysis. In order to achieve this objective, emails with same sender and timestamp as other emails and with the list of the recipients that is a subset of the list of the recipients of the other emails are removed. This method allows us to include only one copy of the duplicated email, which we choose as the copy which includes all the recipients. Out of 603,871 emails, there are 521,316 non-duplicated emails using this method of removing duplicates. Out of 118,185 emails with bucket information, there are 110,979 non-duplicated emails.

Email ID	00000000147E9BDD6D2B...A3F100000005EABB0000
Sender	E1
Recipients	E2, E3, E4
Timestamp	10/18/2002 4:00:15 PM

Email ID	00000000AAD8941CA365...A3F100000020D1940000
Sender	E1
Recipients	E2, E3, E4, E5
Timestamp	10/18/2002 4:00:15 PM

Figure 2: Duplicated emails in the email data set. Recipient initials have been changed to protect privacy.

The entire email data set includes both internal and external emails. A good number of external emails include mass emails sent by newsgroups or other sources, which may not be relevant to our analysis. However, the information from external sources may be important to the workers. Therefore, we would like to study the effect of including or excluding external emails in our analysis. For simplicity, we define internal emails to be emails sent by a person in the firm and the recipients of the emails include at least one person in the firm. Out of the 521,316 non-duplicated emails, there are 59,294 internal emails, and, out of the 110,979 non-duplicated emails with bucket information, there are 20,252 internal emails.

The emails in the email data set have been captured from the firm email server during the period between August 2002 and February 2004. However, a failure in the data capture procedure during a particular time period created some months during which statistically significantly fewer emails were captured than the periods of "normal" traffic. We suspect that the period with low numbers of emails is caused by a failure of the firm's email server which was reported to us during data collection. In order to reduce the effect of this problem, the emails during the period are excluded from the analysis. We therefore use emails collected during the period between 1 October 2002 and 3 March 2003 and between 1 October 2003 and 10 February 2004. During that period, there are 452,500 non-duplicated emails and 45,217 non-duplicated internal emails. Figure 3 shows the number of emails by month during our study period.

YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2002								3623	19169	27177	48127	45865
2003	53488	31586	2166	2583	3964	5108	6561	9564	15984	21911	54680	71989
2004	72468	26225										

(a) Non-duplicated emails

YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2002								557	3481	4969	6020	5307
2003	6037	3592	647	636	977	1072	1501	1817	3428	3639	4149	4538
2004	5205	2011										

(b) Non-duplicated internal emails

Figure 3: The number of emails by months

4.2 Topic Model and Feature Vector Creation

In this study, we represent the contents of documents using a Vector Space Model. Each document is represented by a multi-dimensional vector whose elements are the frequencies of the words in the document. We will call such a vector the "feature vector" of a given email or document. The number of dimensions is based on the number of keywords that we decide to select for representing the content of the email corpus. The common linguistic method for selecting keywords is to create a list of "stop-words," which are the words that are not to be included as keywords. The stop-word list usually contains common words such as "IS", "AM", "ARE", "THE", "AND", and so on. The construction of the stop-word list is likely time-consuming, and the performance is still unknown for each data set. Moreover, it is not applicable to our email data set because the data set only contains hashed words. Without the original words, the construction of the stop-word list is impossible. Therefore, we need a systematic approach for selecting keywords, which will be described in the next section. Once we construct feature vectors, the contents of the documents can be compared by using document similarity metrics such as cosine similarity and information theoretic measures such as entropy and information content.

As the contents of documents are modeled by the directions of feature vectors in n-dimensional space, topics can be modeled as probability distributions of the frequencies of the words that appear in documents in the topics. For example, one can sensibly mention that there is 20 percent chance that a word A appears with the frequency between 0.001 and 0.002 in a random document in topic B. The probability distributions of the frequency of a word in different topics are demonstrated in Figure 4. The probability distributions of all words in a topic are used to represent the content of that topic. We assume that the probability distribution of the frequency of a word has a unique mean for each topic. For example, in documents on the general topic of "Artificial Intelligence," the word "DOG" may have a mean frequency of 0.0001, which means that in an average document about artificial intelligence, there is likely to be one use of the word "DOG" out of 10,000 words. By using this topic model, we will develop a method to select keywords in the next section.

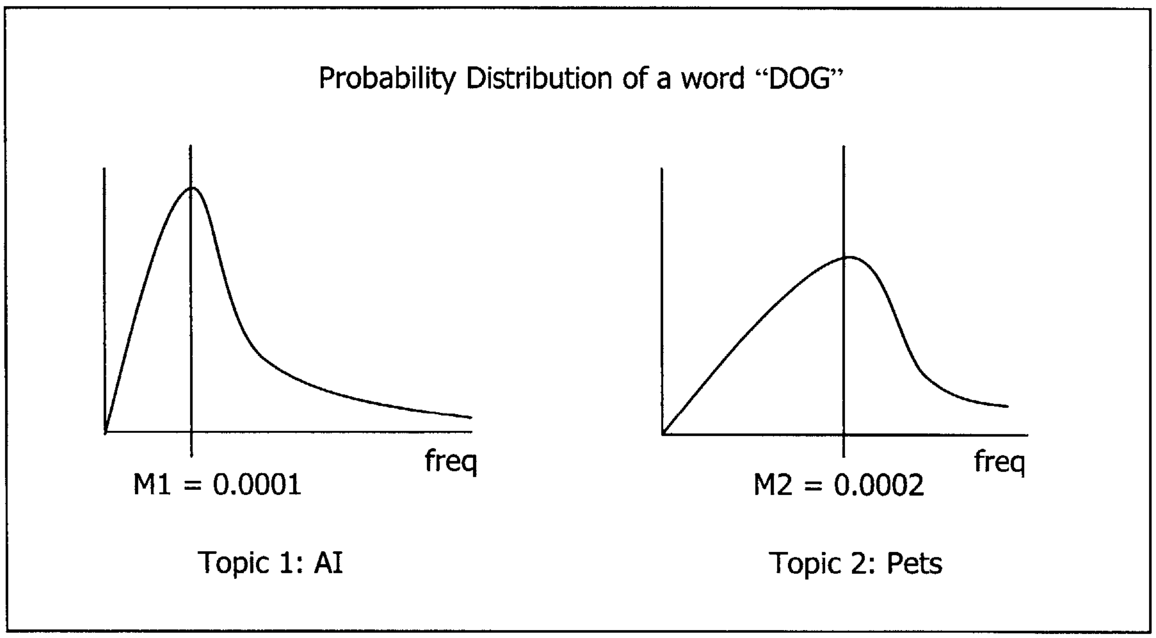


Figure 4: Modeled probability distribution of the frequency of a word in documents pertaining to different topics.

4.3 Keyword Selection

In order to construct feature vectors for each data set, we need to identify a set of keywords. The frequencies of the keywords in a document are then used as the elements of the feature vector, which becomes a representation of the content of that document. Later, we will use the feature vectors to measure the diversity of the content of the associated documents.

One way to select keywords is to pick random words. However, by doing so, we may select mostly words that rarely occur in any documents that we are interested in. The elements of the resulting feature vectors would be mostly zero. In the extreme case, the keywords may not appear in any documents. The feature vectors would be all zero, and they would no longer be able to represent the content of the documents. In order to prevent this outcome, we require that the keywords that we select have at least moderate number of occurrences.

On the other hand, there are words that occur regularly but do not contribute to the content of the documents. For example, the word "THE" usually occurs several times in any document. If such words are used as keywords, the directions of the feature vectors will be biased toward the dimensions associated with the common words due to their usually high frequencies. Thus the inclusion of common words could hinder the ability of the feature vectors to represent the content of the documents. Therefore, common words should not be included as keywords.

Similar to common words, there may be words that happen to occur with relatively uniform frequencies across all documents in the data set. These words effectively cause the same problem as common words, and should be excluded from the set of keywords.

In summary, we require that keywords have the following properties in order to prevent us from selecting words that make it difficult for the feature vectors created to represent the content of the documents in the data set:

- A keyword should occur at least moderately often in the data set.
- A keyword should not be a common word. For example, the word "THE" should not be a keyword.
- A keyword should not occur similarly often in all topics or documents. The differences in the frequencies of keywords enable us to distinguish the contents of the documents.

Since the email data set contains hashed contents, we are unable to select keywords based on human appraisal. We therefore utilize the classification data that we have obtained from eClassifier in order to select keywords that possess the properties that we desire. Similarly, in the Wikipedia.org data set, we use the category information to select the keywords.

From our topic model, a topic is represented by a probability distribution of words. In order to determine whether a word is suitable to be a keyword, we consider the word's distribution across all the topics that we are interested in. In a topic that the word appears often, the probability distribution of the word is modeled as shown in Figure 5(a), as opposed to a probability distribution of the word in a topic that it appears less often as shown in Figure 5(b). Without any context, we model that an article has a fixed probability, $\Pr[topic]$, to be of a certain topic. By using Bayes's rule:
$f(x) = \Pr[topicA] \cdot f(x \mid topicA) + \Pr[topicB] \cdot f(x \mid topicB) + \ldots$, the probability distribution of the word across multiple topics $f(x)$ is a linear combination of the probability distributions of the word in the topics $f(x \mid topic)$. The combined probability distribution is shown in Figure 5(c).

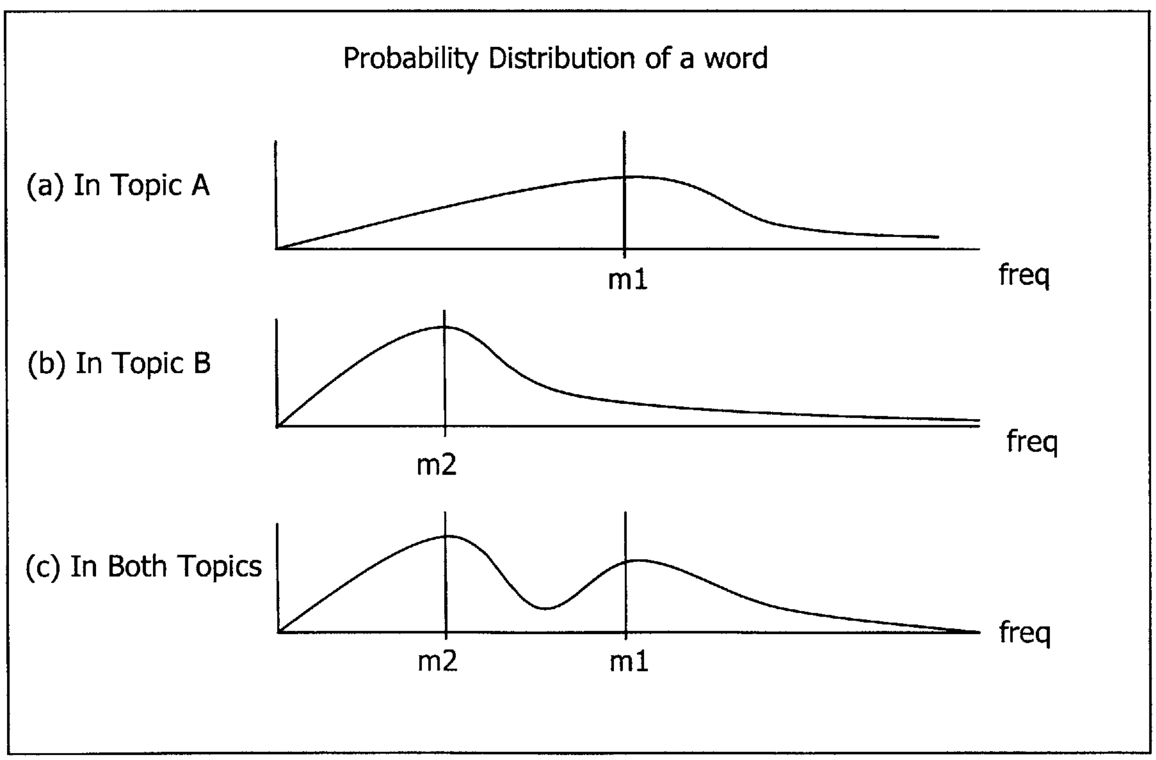


Figure 5: A model of probability distribution of a word in multiple topics

In order to construct feature vectors that successfully represent the content of the documents, keywords should have distinguishable probability distributions across topics as shown in Figure 6(a), as opposed to common words that are likely to appear in all topics with similar frequencies as shown in Figure 6(b). Figure 6(a) shows a probability distribution of a word across multiple topics, which is a linear combination of the separated probability distributions of the word in those topics. Many peaks at various frequencies imply that the word is likely to occur with different frequencies across different topics. On the other hand, the peaks of the distribution in Figure 6(b) are at similar frequencies, implying that the word is likely to occur with similar frequency across different topics. Therefore, the word in Figure 6(a) is more suitable to be a keyword because it enables us to *distinguish* the content of topics.

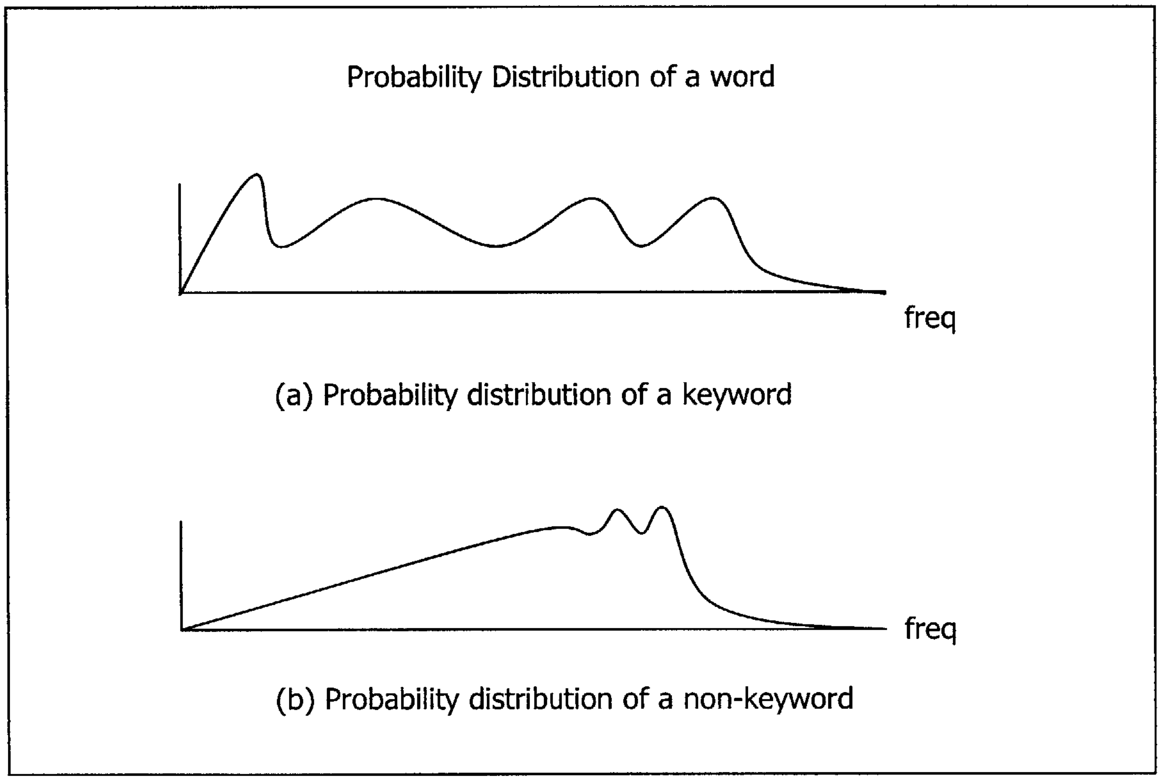


Figure 6: Probability distributions of keywords and non-keywords

In reality, we do not know the actual probability distributions of words. To approximate the probability distribution of a word, we evaluate the frequencies of the word in all buckets and use the values of the frequencies to estimate the mean frequencies of the word in the buckets or topics. We expect the dispersion of the mean frequencies across topics to be high in keywords. In our study, we find the coefficient of variation (the ratio of the standard deviation to the mean) of the mean frequencies across topics to be a good indicator of this dispersion. A desirable property of the coefficient of variation is that it compensates for the effect of the mean value on the dispersion. Because of this property of scale-invariance, many studies have used the coefficient of variation to provide measures of dispersion that are comparable across multiple data sets with heterogeneous mean values (Allison, 1978, Pfeffer & O'Reilly, 1987, Ancona & Caldwell, 1992). In Appendix C, we will see that variance does not have this property and is not suitable for measuring the dispersion for keyword selection. We define this inter-topic frequency deviation to be the value of variance over mean squared (which is the square of the coefficient of variation) as follows:

$$\text{Dinter} = \frac{1}{M^2} \sum_{b \in buckets} (m_b - M)^2$$

The main reason for using the squared value is to reduce unnecessary computation. We are only interested in the ranking of words based on the coefficient of variation, which is the same as the ranking based on Dinter due to the fact that the square function is monotonic. More detailed information about the threshold and the keyword selection process can be found in Appendix C.

The downside of the coefficient of variation is that its value can be unreasonably high when the mean frequency is low, which is likely the case for most of the words in our study. For example, the coefficient of variation of a word that occurs only once in a topic and nowhere else is going to be as high as the coefficient of variation of another word that occurs a hundred times uniformly in a topic and never occurs in other topics. We do not wish to select the words with extremely low frequencies because such words are not likely to successfully represent the content of any entire topics. Unfortunately, words with low frequencies are likely to have high coefficient of variation because of the division by the square of the frequencies. In order to solve this problem, instead of selecting words based only on high coefficient of variation, we eliminate words with low coefficient of variation, and then select words with high frequencies in equal numbers from all topics.

Additionally, we need to confirm that the words, which possess varying frequencies across topics, actually distinguish the topics by having uniform frequencies across documents within topics. For example, a word like "GOAL" is likely to have uniform frequencies in topic

"SOCCER" because it is likely to appear in a relatively large number of documents about soccer. However, there are words that are unique to documents. Even though they seem to appear frequently in a topic, they only appear often in a few specific documents in the topic. For example, a name of a person may appear regularly in his biography, but it may not appear in any other documents of the same topic. These words are not suitable for representing the content of topics. Therefore, we define an intra-topic frequency deviation for each word as follows:

$$\text{Dintra} = \frac{1}{M^2} \sum_{b \in buckets} \sum_{d \in b} (f_d - m_b)^2$$

In order to be a keyword, a word needs to have high inter-topic frequency deviation and low intra-topic frequency deviation. We decide to select keywords using inter-topic deviations and frequencies as mentioned before, and we eliminate the remaining words with high intra-topic deviations as described in details in Appendix C. We find that the keywords selected by this method represent and distinguish all the topics well. We present evidence of the ability of this method to represent and distinguish all topics or categories in the Wikipedia.org Data Set in Section 5.1.

To evaluate whether a set of keywords represents the data set and whether it is able to distinguish the contents of topics, we define:

$$CosDist(A,B) = 1 - CosSim(A,B) = 1 - \cos(angle\_between\_A\_and\_B)$$

$$\text{Adhesion of a bucket B} = \min_{b \in buckets - B} \{CosDist(m_b, m_B)\}$$

$$\text{Inverse Cohesion (InvCohesion) of a bucket B} = \underset{d \in B}{average}\{CosDist(v_d, m_B)\}$$

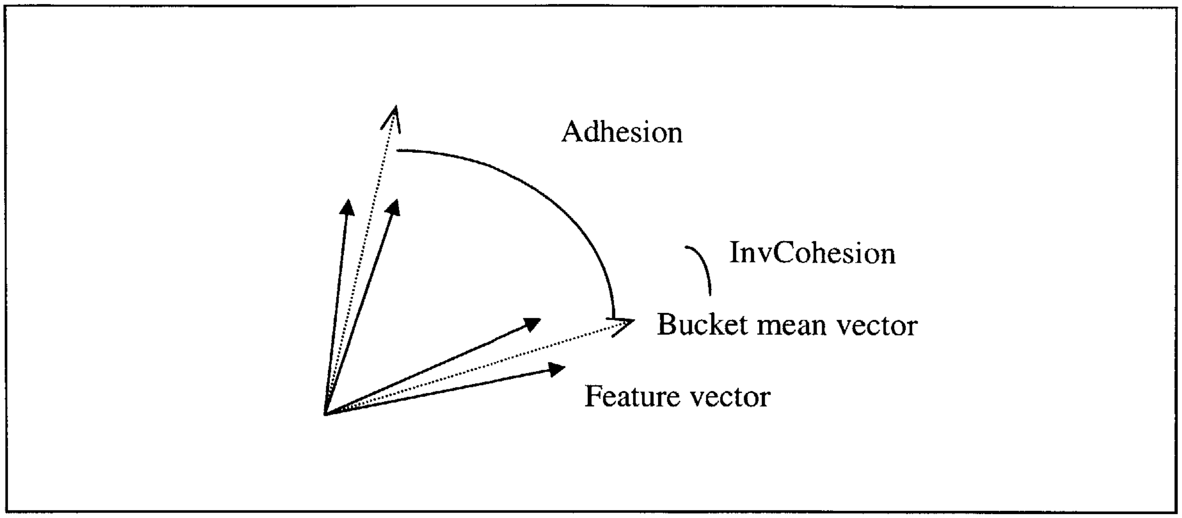


Figure 7: Adhesion and inverse cohesion

Adhesion:

Adhesion measures the distance from a specific bucket to its closest bucket based on the cosine distances between the means of the buckets' feature vectors. High adhesion indicates that the bucket is different (distinguishable) from other buckets, which is a desirable property for a good set of keywords chosen to distinguish topics.

Cohesion:

Cohesion (1 - InvCohesion) is the average of cosine similarities between every feature vector in the bucket and the mean of the bucket. It measures the similarity of the contents of the documents in bucket as represented by the feature vectors. High cohesion indicates that the documents have similar contents and shows that the mean of the feature vectors in the bucket is a good representative of the collective content of the bucket. On the other hand, inverse cohesion is the average of cosine distances from each vector to the mean of the bucket. It can be used interchangeably with cohesion as the indicator for the similarity of the contents of the documents within a bucket or a topic. The difference is that low inverse cohesion is desirable. We decide to use inverse cohesion instead of cohesion because both adhesion and inverse cohesion are measurements of distances separating content, so they are more comparable than adhesion and cohesion.

In summary, high average adhesion of buckets indicates that the set of keywords produces feature vectors that are able to distinguish buckets or topics. However, to be certain of the property, low average inverse cohesion of buckets is needed to confirm the uniformity of the contents within buckets.

4.4 Diversity Metrics

4.4.1 Feature-vector-based Diversity Metrics

When properly constructed, feature vectors provide accurate representations of the contents of documents. Many well-known metrics, for example, cosine similarity and Dice's coefficient, measure the similarity of documents based on their feature vectors. These pair-wise similarity metrics can be used to derive diversity metrics for document sets.

Cosine Deviation from Mean

The main idea of this metric is to derive a measure of diversity by aggregating cosine distances. In order to measure the deviation of the feature vectors, we explore the concept of variance, which is a well-known measurement for deviation in a data set. As the first diversity metric, we use the variance of feature vectors based on cosine distances to represent diversity. The mean vector is the average of all feature vectors, and the variance is computed by averaging the square distances from every feature vector in the document set to the mean vector.

$$MeanDistCos(d) = CosDist(d, M)$$

$$VarCos = \frac{1}{number\_of\_documents} \sum_{d \in documents} (MeanDistCos(d))^2$$

Figure 8 demonstrates the distances from the feature vectors to the mean vector. The VarCos value is the average of the square of the distances.

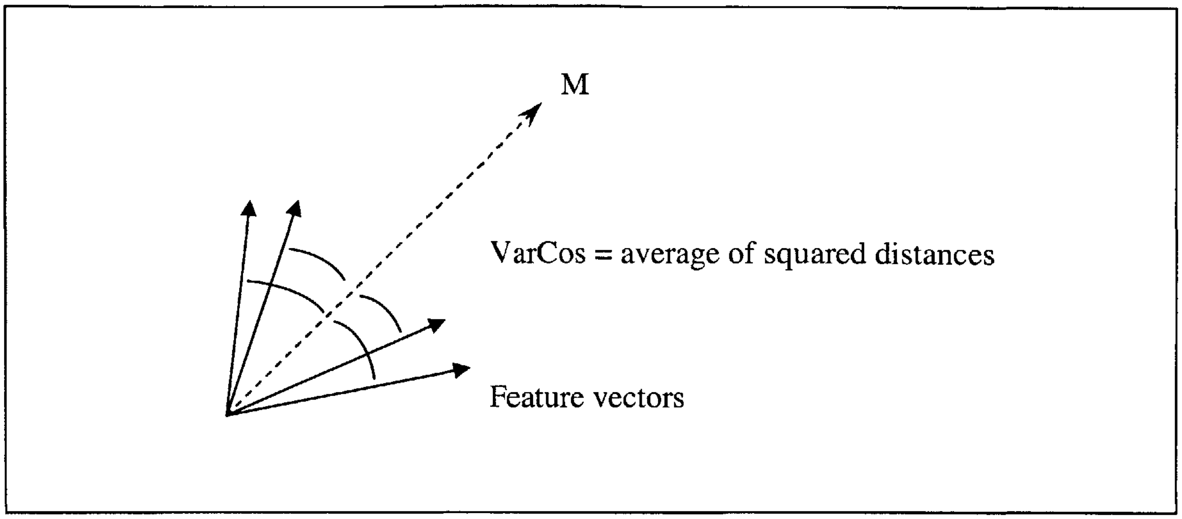


Figure 8: Variance of Cosine Distances

Dice's Coefficient Deviation from Mean

Dice's coefficient is another metric for measuring similarity between two feature vectors. Similar to cosine distance, the variance of feature vectors based on Dice's coefficient distances represents the diversity of the document set.

$$MeanDistDice(d) = DiceDist(d, M) = 1 - DiceCoeff(d, M)$$

$$VarDice = \frac{1}{number\_of\_documents} \sum_{d \in documents} (MeanDistDice(d))^2$$

The results from vector-based diversity metrics will be compared to evaluate the relative performance of different distance metrics for our purposes. Additionally, the results will be compared against the results from the bucket-information-based diversity metrics defined below in order to evaluate how the results based on different concepts perform. The goal is to derive a set of metrics that measures the diversity of content in employees email.

4.4.2 Bucket-information-based Diversity Metrics

Feature vectors are not the only method by which we can derive diversity metrics. The information available in the eClassifier data enables additional methods for capturing content diversity. By using multiple methods for measuring diversity, we can establish the robustness of our measurement by comparing diversity measures produced by various measures. The feature vectors generated by keywords are used to construct distance based diversity metrics and they also provide additional information regarding the contents of individual documents. However it could be that feature vector based diversity metrics may also introduce additional errors. In order to test and address this issue, the feature-vector-based diversity metrics will be compared against diversity metrics derived entirely or mostly based on eClassifier bucket information.

Average Common Bucket

The Average Common Bucket diversity metric is derived from the basic idea that two documents classified into the same eClassifier bucket are likely to be more similar to each other than two documents not classified into the same bucket. This assumption can be incorrect if there exist two buckets that contain documents with similar contents and another bucket that contains documents with comparatively diverse contents. However, in general, the assumption will hold. To control for the potential bias created by different levels of diversity within buckets, we also utilize an Average Common Bucket measure of diversity that takes the 'information content' of the buckets into consideration (this metric is described in the next section). For every two documents, the Common Bucket Similarity is defined to be the number of levels that D1 and D2 are in the same bucket over the total number of levels.

$$CommonSim(D_1, D_2) = \frac{number\_of\_levels\_in\_same\_bucket}{total\_number\_of\_bucket\_levels}$$

On the other hand, the Common Bucket Distance is the number of levels that D1 and D2 are in the different buckets over the total number of levels.

$$CommonDist(D_1, D_2) = 1 - CommonSim(D_1, D_2)$$

The Average Common Bucket diversity metric is defined to be the average of the Common Bucket Distances between every two documents in the document set.

$$AvgCommon = \underset{d_1, d_2 \in documents}{average} \{CommonDist(d_1, d_2)\}$$

This metric represents the average difference in the contents of two documents in the document set based on the Common Bucket Distance.

Average Common Bucket with Information Content

The Average Common Bucket diversity metric assumes that the amount of information conveyed by the fact that two documents reside in the same bucket is the same for all buckets in all levels. However, this assumption is unlikely to hold true. For example, documents related to animals may contain more diverse contents than documents related to dogs. Different levels of diversity across different buckets will bias the results of bucket-information-based diversity metrics that assume that a pair of documents in a given bucket is as similar (to each other) as another pair of documents in another bucket. In order to correct for this potential bias, we consider the 'information content' of the buckets. The information content, usually represented by the log value of the probability of a concept, allows us to evaluate the amount of information conveyed by the coexistence of documents in a specific bucket. The idea of the information content can be combined with Common Bucket Similarity by defining the Common Bucket Similarity with Information Content to be the normalized sum of the information content of all bucket levels, in which the two documents share the same bucket.

$$CommonICSim(D_1, D_2) = \frac{1}{\log(\frac{1}{\|all\_documents\|})} \cdot \frac{\sum_{D_1, D_2 in\_same\_bucket} \log\left(\frac{\|documents\_in\_the\_bucket\|}{\|all\_documents\|}\right)}{total\_number\_of\_bucket\_levels}$$

We also define:

$$CommonICDist(D_1, D_2) = 1 - CommonICSim(D_1, D_2)$$

Similar to the previous metric, the Average Common Bucket diversity metric with Information Content is defined to be the average of the Common Bucket Distance with Information Content between every two documents in the document set.

$$AvgCommonIC = \underset{d_1, d_2 \in documents}{average} \{CommonICDist(d_1, d_2)\}$$

The potential problem with the implementation of this metric is that the information content [IC = log(p)] is theoretically ranged from 0 to infinity. However, we would like the diversity measurement to range from 0 to 1. Our attempt to normalize the information content by dividing by the minimum possible value log(1/#total docs) which is comparatively larger than the information contents for most buckets, resulting in small similarity measurements and close-to-one diversity measurements. Fortunately, the actual numerical values are not important in our application. The comparisons to other metrics will determine whether this metric performs well in ranking the diversity.

Average Bucket Distance

The idea of this metric is built upon the idea of the Average Common Bucket metric. The Average Common Bucket assumes that two documents are alike (with the distance of 0) only when they are in the same bucket. Otherwise, they have the distance of 1 no matter which buckets they reside in. However, more information can be drawn from the fact that the contents of every two buckets or topics are not completely different. The contents in two buckets can be more similar than the contents of other two buckets. In this metric, the level of similarity or dissimilarity between buckets is measured by the cosine distance between the mean vectors of the buckets.

$$BucketDist(B_1, B_2) = CosDist(m_{B_1}, m_{B_2})$$

In the document level, the average of the bucket distances across all levels of buckets represents the dissimilarity between two documents:

$$DocBucketDist(D_1, D_2) = \frac{1}{\|bucket\_levels\|} \cdot \sum_{i \in bucket\_levels} \left(BucketDist(B_{level=i,D_1}, B_{level=i,D_2}) \right)$$

Similar to other average metric, we average the document distances for every two documents in the document set to measure diversity of the document set:

$$AvgBucDiff = \underset{d_1, d_2 \in documents}{average} \{DocBucDist(d_1, d_2)\}$$

This metric provides an alternative extension to the Average Common Bucket metric. Compared to the AvgCommonIC, AvgBucDiff captures the idea that two documents in different buckets are not entirely dissimilar instead of the information content. The results from the three bucket-information-based metrics will be compared in later sections.

Metric	Description and Purpose
VarCos	Variance based on cosine distance (cosine similarity): $MeanDistCos(d) = CosDist(d, M)$ $VarCos = \frac{1}{number\_of\_documents} \sum_{d \in documents} (MeanDistCos(d))^2$ The value of variance reflects the deviation of the data, which, in this case, are the feature vectors. The distance measurement is derived from a well-known similarity measurement, cosine similarity.
VarDice	Variance based on Dice distance (Dice coefficient): $MeanDistDice(d) = DiceDist(d, M) = 1 - DiceCoeff(d, M)$ $VarDice = \frac{1}{number\_of\_documents} \sum_{d \in documents} (MeanDistDice(d))^2$ Similar to VarCos, variance is used to reflect the deviation of the feature vectors. Dice coefficient is used as an alternative to cosine similarity.
AvgCommon	AvgCommon measures the level to which the documents in the document set reside in different eClassifier buckets: $CommonSim(D_1, D_2) = \frac{number\_of\_levels\_in\_same\_bucket}{total\_number\_of\_bucket\_levels}$ $CommonDist(D_1, D_2) = 1 - CommonSim(D_1, D_2)$ $AvgCommon = \underset{d_1, d_2 \in documents}{average} \{CommonDist(d_1, d_2)\}$ AvgCommon is derived from the concept that documents are similar if they are in the same bucket, and they are dissimilar if they are not in the same bucket. Therefore, every two documents in the document set that are in different buckets contribute to high diversity value.

AvgCommonIC	AvgCommonIC uses information content to measure the level to which the documents reside in different buckets: $CommonICSim(D_1, D_2) = \frac{1}{\log(\frac{1}{\lVert all\_documents \rVert})} \cdot \frac{\sum_{D_1, D_2 in\_same\_bucket} \log\left(\frac{\lVert documents\_in\_the\_bucket \rVert}{\lVert all\_documents \rVert}\right)}{total\_number\_of\_bucket\_levels}$ $CommonICDist(D_1, D_2) = 1 - CommonICSim(D_1, D_2)$ $AvgCommonIC = \underset{d_1, d_2 \in documents}{average} \{CommonICDist(d_1, d_2)\}$ AvgCommonIC extends the concept of AvgCommon by compensating for the different amount of information provided by the fact that documents reside in the same bucket for different buckets. For example, the fact that two documents are both in a bucket with low intra-bucket diversity is likely to imply more similarity between the two documents than the fact that two documents reside in a bucket with high intra-bucket diversity.
AvgBucDiff	AvgBucDiff measures diversity using the similarity/distance between the buckets that contain the documents in the document set: $BucketDist(B_1, B_2) = CosDist(m_{B_1}, m_{B_2})$ $DocBucketDist(D_1, D_2) = \frac{1}{\lVert bucket\_levels \rVert} \cdot \sum_{i \in bucket\_levels} (BucketDist(B_{level=i, D_1}, B_{level=i, D_2}))$ $AvgBucDiff = \underset{d_1, d_2 \in documents}{average} \{DocBucDist(d_1, d_2)\}$ AvgBucDiff extends the concept of AvgCommon by using the similarity/distance between buckets. While AvgCommon only differentiates whether two documents are in the same bucket, AvgBucDiff also considers the distance between the buckets that contain the documents.

Figure 9: Summary of diversity metrics

4.5 Triplet Test on Wikipedia.org Data Set

We construct the Wikipedia.org data set so that we are able to evaluate the diversity metrics against the pre-classified documents. In order to utilize the structure of the categories, we construct test cases, each of which contains three documents from various combinations of categories. There are ten distinct configurations of categories for three-document test cases as show in Figure 10.

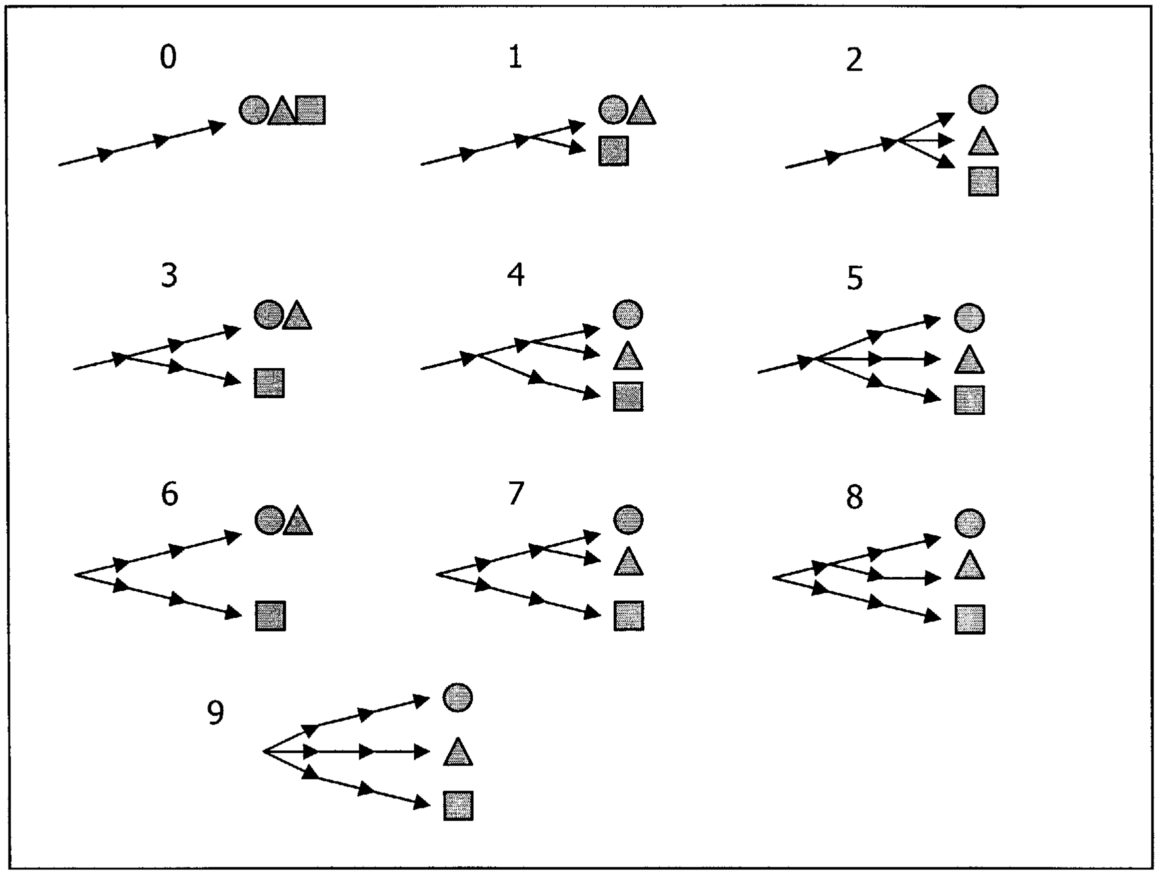


Figure 10: All ten distinct category configurations for three-document test cases

The numbers associated with the configurations are called "configuration types" or "types." The configurations are numbered in the order of our estimation of their diversity. By selecting documents from the same subcategories of the hierarchy of topics we aim create a cluster of documents with low diversity (Type 0). By selecting documents from maximally different subcategories, we aim create a cluster of documents with high content diversity (Type 9). As the topic hierarchy is defined a priori to group like documents, we can use these clusters to test whether our diversity metrics can accurately describe the diversity of clusters that are created to either be maximally diverse or minimally diverse. The order is loosely defined because in some cases, the order of diversity is not obvious. For example, type-5 and type-6 configurations cannot be trivially ranked. Type-5 configuration contains documents that are in different second-level categories. Even though type-6 configuration contains a document that is in a different first level category, the other two documents are in the same category in all levels. Nevertheless, it is unquestionable that the type-9 configuration is likely to be much more diverse than the type-0 configuration. The order of the configuration types will be used as a reference to evaluate the performance of diversity metrics.

To implement this evaluation, we generate all combinations of three documents as test cases. For each test case of three documents, we compute the diversity scores for the test case. Eventually, we compute the average of the diversity scores of the test cases for each configuration type. We call this test, the triplet test. Upon obtaining the average diversity scores for all configuration types, we use the correlations between diversity scores and configuration types to show the performance of diversity metrics. The correlations of diversity scores across diversity metrics indicate the relationships between diversity metrics. We will show and interpret the results in later sections.

We also explore the effect of the number of emails in test cases. The three-document test cases are restrictive, so we implement a test, which we call the extended triplet test. The extended triplet test mimics the process of the triplet test. The difference is that each configuration branch represents multiple documents instead of one document. For example, if a branch represents two documents, the number of documents in the test case becomes six. This method increases the number of documents, while preserving the configuration types. To implement the extended triplet test, we are unable to compute the average of all combinations of documents because the number of combinations grows exponentially, so we restrict the computation by limiting the number of combinations for each configuration.

5. Results

5.1 Results from Wikipedia.org Data Set

5.1.1 Keyword Selection on Wikipedia.org Data Set

Ideally, words with common roots should be grouped together. For example, all tenses of a verb or single and plural forms of a noun should not be considered as separated words. However, to simplify this process, we decide to remove the letter "S" at the end of all words to eliminate the common noun single-plural-form indicator with the assumption that the keywords are likely to be nouns that indicate specific objects. Using this method, we list all words and compute the frequencies and deviations of frequencies for every word.

Out of 15,349 distinct words in the 291 Wikipedia articles, we decide to select approximately 400 words to use as keywords. By choosing different thresholds of the inter-topic frequency deviations, the resulting cohesion-adhesion measurements are shown in Figure 11. We notice a significant improvement in adhesion at a very low threshold, and the adhesion improves at a diminishing rate as the threshold increases. Inverse cohesion also increases, but it increases at a lower rate than adhesion. We find that the initial increase in both adhesion and inverse cohesion results from the exclusion of the words "THE" and "AND", which are the two most frequent words in the data set. By including the two words, the feature vectors are inclined toward the two dimensions represented by the two words, and thus cluster more closely than they should.

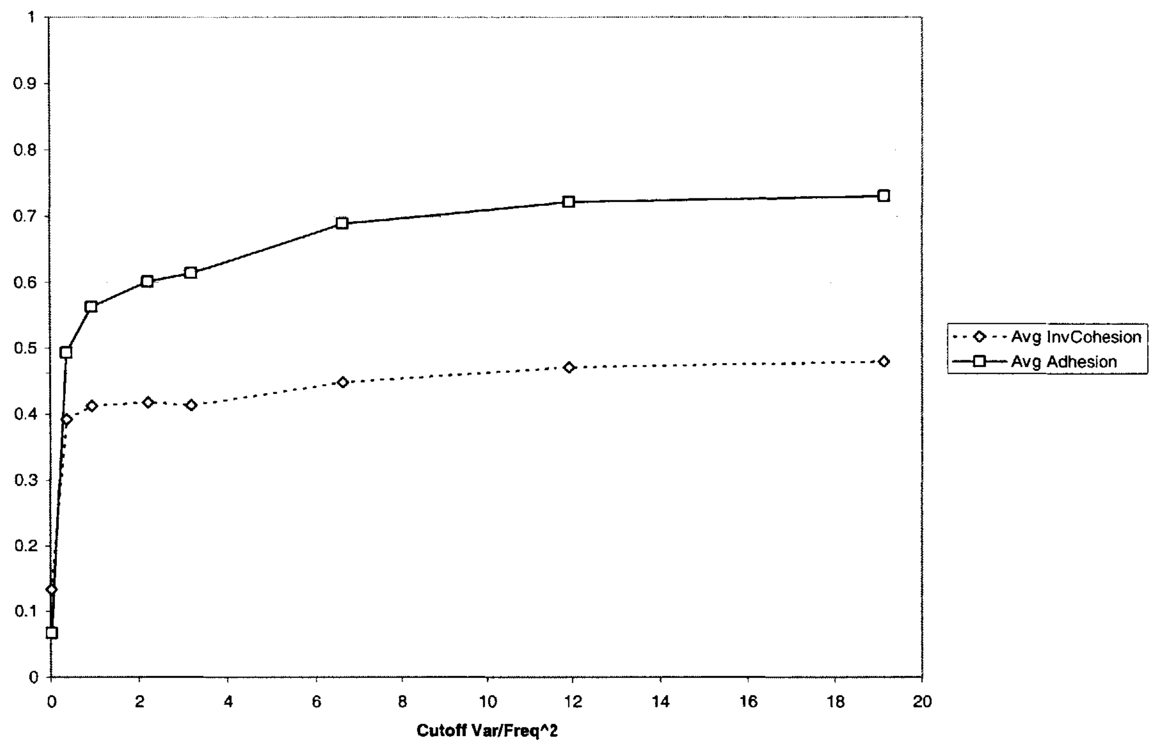


Figure 11: Inverse cohesion and adhesion across multiple thresholds in Wikipedia.org data set

Adhesion and inverse cohesion increase at a diminishing rate. We use the threshold where they start to remain unchanged. In order to study the effect of using different thresholds on diversity rankings, we select keywords by using three different thresholds, and find the correlations between scores generated from our diversity metrics. The result in Figure 12 shows that different thresholds within a suitable range of thresholds create highly correlated diversity results and therefore do not have a significant effect on the diversity rankings.

Correlation – Cutoff	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
14000-13000	0.9960	0.9968	0.9920	0.9974	0.9914
14000-14500	0.9972	0.9976	0.9926	0.9906	0.9879
13000-14500	0.9961	0.9960	0.9914	0.9950	0.9891

Figure 12: Correlations of diversity scores from different set of keywords generated by three different thresholds: the numbers of words that pass the thresholds are 13000, 14000, and 14500 words

Upon examination, the keywords selected by this method are words that exemplify different topics. Examples are shown in Figure 13.

- accessed
- adopted
- africa
- age
- agp
- air
- aircraft
- albedo
- altitude
- angle
- api
- apollo
- appearance
- appeared
- approache
- arena
- arm
- array
- asia
- assembly
- assume
- assumption
- asteroid
- astronomy
- ati
- atla
- atmosphere
- atmospheric
- attack
- audio
- australia
- authentication
- automatic
- autonomou
- balance
- base
- beacon
- beagle
- behavior
- best
- binary
- biological
- biomedical
- bit
- block
- board
- building
- burke
- camera
- cape
- capture
- card
- cartography
- catalyst
- cbc
- celestial
- cell
- challenge
- champion
- character
- chart
- chemical
- chip
- chipset
- chosen
- cipher
- ciphertext
- circuit
- classification
- client
- climate
- closer
- cloud
- cluster
- clustering
- coast
- codec
- cold
- color
- colour

Figure 13: Examples of keywords from Wikipedia.org data set

The keyword selection for Wikipedia.org data set is described in details in Appendix C.

5.1.2 Triplet Test on Wikipedia.org Data Set

The diversity scores derived from our metrics are computed on many combinations of documents from the Wikipedia.org data set. Since the diversity scores are to be compared with the configuration types, which are directly derived from the category structure, we decide against using category information directly as bucket information for the bucket-information-based diversity metrics. Instead, the bucket information is generated by performing K-Means clustering on the feature vectors. The averages of the diversity scores grouped by configuration types are shown in Figure 14.

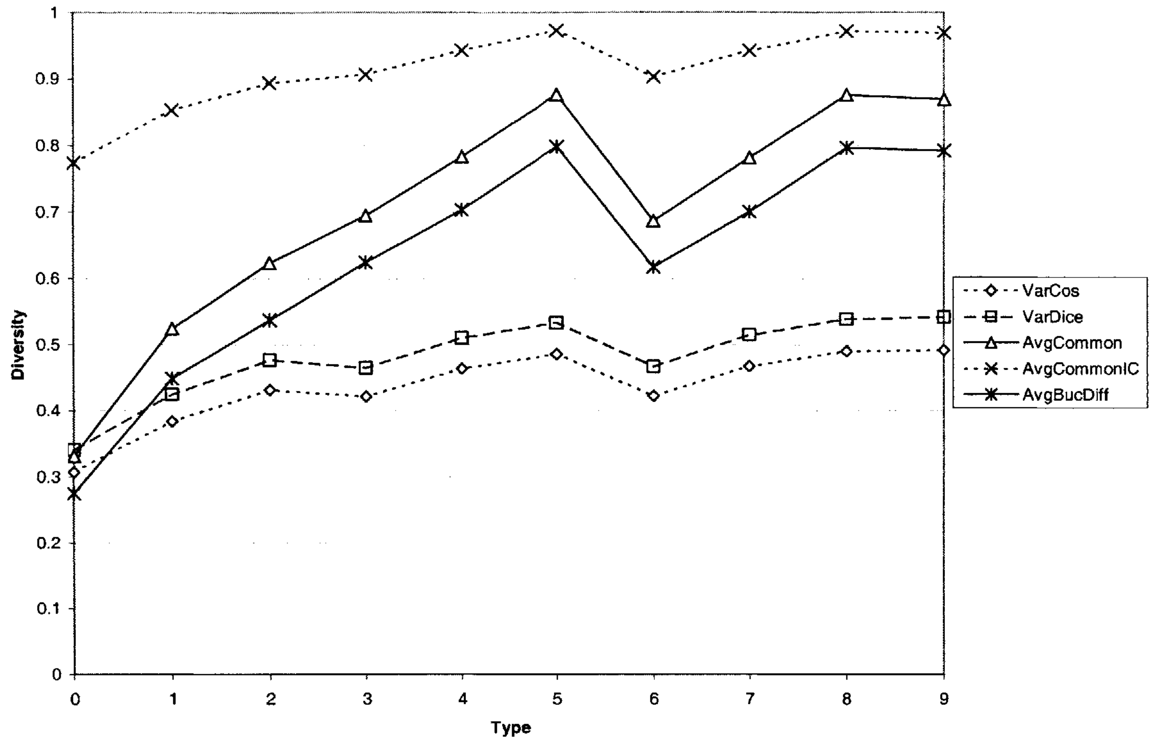


Figure 14: Averages of diversity scores grouped by configuration types

The chart in Figure 14 shows that all diversity metrics behave similarly. However, an unexpected result is that the average diversity score of type-5 configuration is approximately as high as type-9 configuration. The type-5 configuration contains three documents which are in the same main category but are all in different second-level categories. The type-9 configuration contains three documents that are all in different main categories. Ideally, we predict that type-9 configuration should possess higher average diversity score than type-5 configuration. However, the result contradicts this prediction. We explain the result by assuming that in this Wikipedia.org data set, the main categories contain so highly diverse contents that the dissimilarities between documents in the same main category may be close to the dissimilarities of documents across main categories. For example, in the "Technology" main category, documents in Robotics are dissimilar to documents in Video and Movie Technology, while those documents may contain similarities with documents in the Computer Science main category. If this assumption is true, the main categories will have little effect toward diversity scores. Therefore, type-3 and type-6 configurations should have similar scores. Type-4 and type-7 configurations should have similar scores, and so should type-5, type-8 and type-9 configurations. The results demonstrate that these similarities and dissimilarities exist, giving us confidence on our explanation.

Despite the unexpected shape of the chart in Figure 14 caused by the diverse contents of the main categories, we notice increasing diversity from type-0 to type-5 and from type-6 to type 8 in the bucket-information-based diversity scores as we expect. The assumption that the main categories contain diverse contents explains the decrease in diversity scores from type-5 to type-6 and the similar scores between type-3 and type-6, between type-4 and type-7, and between type-5, type-8, and type-9 in all diversity metrics. We also find decreases in the feature-vector-based diversity scores from type-2 to type-3. This result can be explained by the fact that the comparison of the diversity of type-2 and type-3 configurations is not trivial. The type-3 configuration contains three documents in the same second-level category, but all three documents are in different third-level categories. The type-2 configuration contains two documents in the same third-level category and the other document in a different second-level category from the first two documents as shown in Figure 10. The diversity ranking of type-2 and type-3 depends on the difference of the contents of documents across the third-level categories. If the difference is small compared to the difference of contents across the second-level categories, the type-2 configuration is likely to possess a lower diversity score because all three documents are in the same second-level category. If the differences are comparable, the fact that two documents in type-3 share the same third-level category causes type-3 to possess a lower diversity score. Therefore, it is not surprising that different diversity metrics rank the diversity of type-2 and type-3 in different orders.

Figure 15 shows high correlations of diversity scores derived from our diversity metrics. They confirm that the diversity metrics behave similarly. Despite the unexpected result above, we still achieve high correlations between diversity scores and the configuration types.

Correlations	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	0.9999	1.0000			
AvgCommon	0.9855	0.9845	1.0000		
AvgCommonIC	0.9943	0.9937	0.9973	1.0000	
AvgBucDiff	0.9790	0.9778	0.9993	0.9939	1.0000
Type	0.8292	0.8307	0.8539	0.8339	0.8609

Figure 15: Correlations of diversity scores across multiple metrics

In the extended triplet test, we compute average diversity scores of 6-document document sets and 9-document document sets grouped by configuration types. Figure 16 shows high correlations across different sizes of document sets, indicating that the increasing size does not affect the performance of diversity rankings. Figure 17 confirms that the diversity scores behave similarly. The diversity scores increase at a diminishing rate as the number of documents increases. The property is intuitive because the increase in the number of the documents introduces additional dissimilarities to the document set, and the effect is weaker as the number of documents increases.

Correlation - #docs	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
3docs-6docs	0.9990	0.9992	0.9998	0.9997	0.9998
3docs-9docs	0.9977	0.9982	0.9997	0.9995	0.9997
6docs-9docs	0.9996	0.9997	0.9996	0.9997	0.9997

Figure 16: Correlations of diversity scores across multiple sizes of document sets

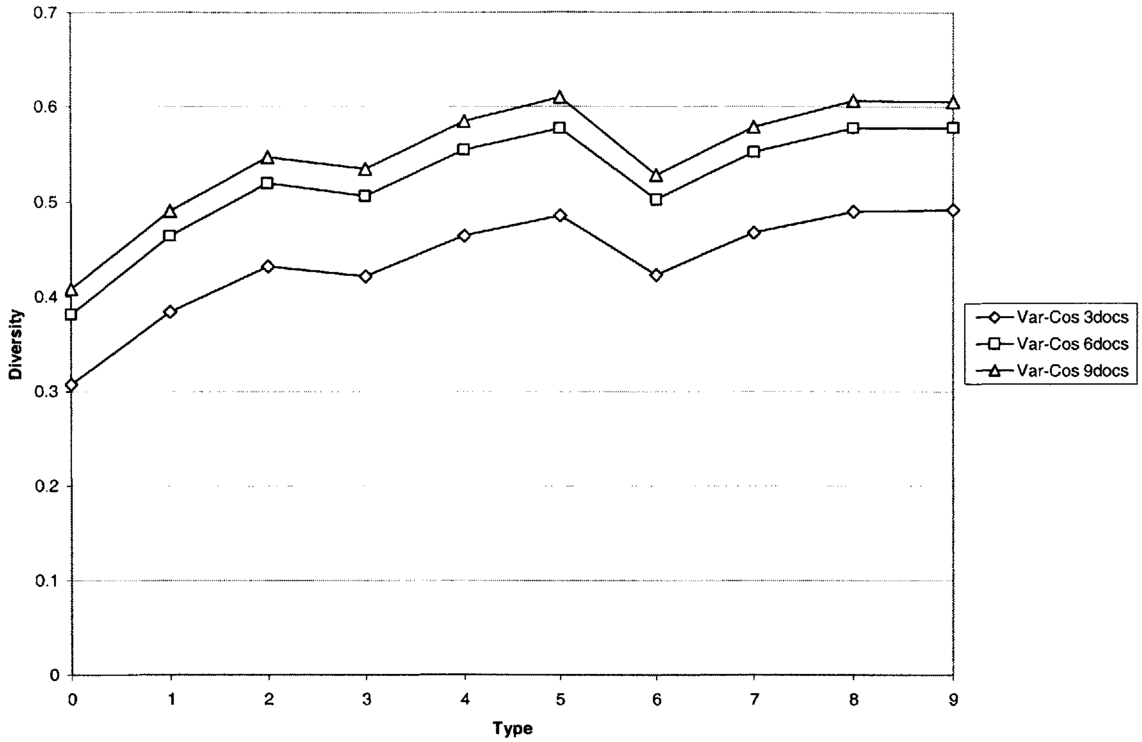


Figure 17: VarCos scores vs configuration types across multiple sizes of document sets

The chart in Figure 17 indicates a similar trend across multiple sizes of document sets used to compute diversity scores. The overall shape can be explained by the highly diverse contents of the main categories in our Wikipedia.org data set. We also notice that the VarCos metric indicates that type-2 configuration is more diverse than type-3 configuration in all sizes of the document sets, similar to the previous results from the feature-vector-based diversity metrics.

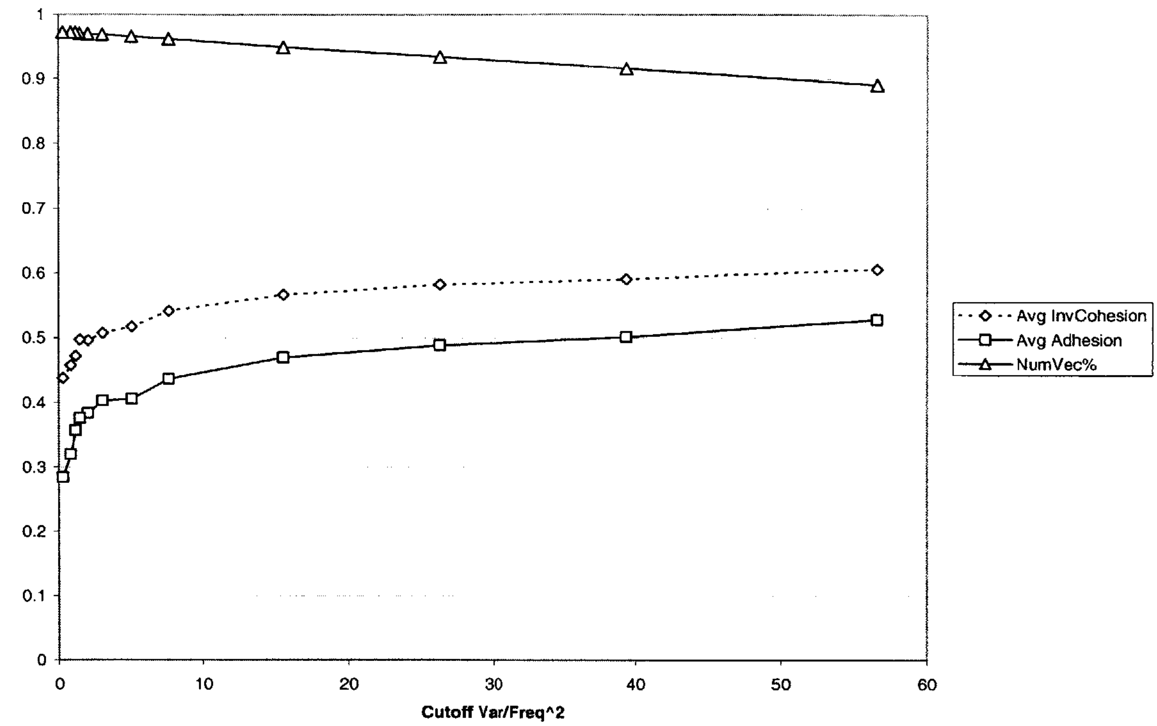
In summary, our diversity metrics are shown to behave similarly in every situation we have encountered. Despite some unexpected outcome, they still correlate well with our manually-assigned configuration types. They also demonstrate an expected behavior as the number of documents in the document set increases. The triplet test and the Wikipedia.org data set have shown that our diversity metrics are appropriate measurements for diversity.

5.2 Results on Email Data Set

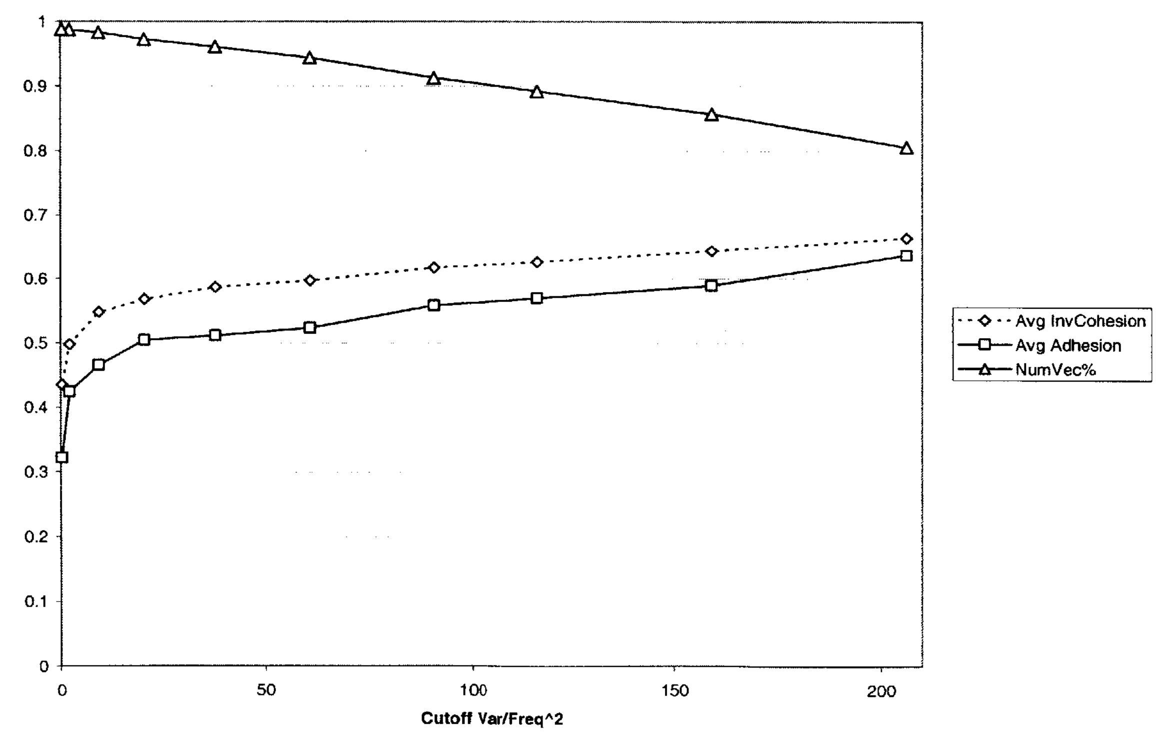
5.2.1 Keyword Selection on Email Data Set

While the Wikipedia.org data set contains category information, the email data set contains bucket information from the eClassifier output. The difference is that the email data set does not contain the bucket information for all emails. According to our findings, there are 110,979 non-duplicated emails (BucSetAll) with bucket information, out of which, 20,252 emails are internal emails (BucSetInt). The larger set of emails is likely to be a better set to use for selecting keywords, but it is possible that the inclusion of the external emails may deteriorate the quality of the resulting keywords. Therefore, both sets of emails are used to select different sets of keywords, and the resulting keywords are compared.

For both sets of emails, approximately 1,500 keywords are selected. Unlike in the Wikipedia.org data set, we are no longer able to generate enough keywords so that every document contains at least one keyword. Therefore, while constructing feature vectors, emails with no keywords are ignored. It should be noted however that emails differ from Wikipedia entries in that emails may be one line long and therefore not contain any keywords. Therefore the lack of keywords in emails may be an accurate description of the topic content of these emails. As the threshold increases, the number of emails without keywords increases because the use of variance over frequency squared eliminates words with high frequencies. This issue raises a concern that high thresholds may lead to keywords with very low frequencies, which are not desirable. Therefore, we only use thresholds, whose resulting keywords are able to construct feature vectors for most of the emails. Using a set of keywords, we construct feature vectors by using the frequencies of the keywords in the emails. We disregard emails that do not contain any keywords. The percentage of feature vectors generated from emails (the number of generated feature vectors over the total number of emails) is one of the factors that show us whether the set of keywords is sufficient to represent the content of the emails. Figure 18 shows the effect of different thresholds on inverse cohesion, adhesion, and the percentage of feature vectors constructed from emails. Expectedly, the percentage of feature vectors decreases as the threshold increases. Adhesion and inverse cohesion also increase at a diminishing rate. As before, we decide to select the threshold at the point at which adhesion and inverse cohesion start to remain unchanged.



(a) Keyword selection on BucSetAll



(b) Keyword selection on BucSetInt

Figure 18: Adhesion and InvCohesion during the keyword selection on email data set

In order to study the effect of including external emails for keyword selection, we use two sets of keywords generated from BucSetAll and BucSetInt to construct feature vectors. The diversity scores of the employees are then computed from their incoming and outgoing internal emails. The correlations between the scores from the different sets of keywords are shown in Figure 19. The scores are expectedly correlated, but the correlations are only moderate. Therefore, the inclusion of the external emails does affect the keywords selected by our method. The keywords selected from BucSetInt are restricted to the content of internal emails, which are more likely to contain work-related content. On the other hand, the keywords selected from BucSetAll are likely to contain some keywords that are related to additional content such as news provided by the inclusion of external emails. The effect of the additional content on keywords still remains to be studied.

Correlation - Keyword	**VarCos**	**VarDice**	**AvgCommon**	**AvgCommonIC**	**AvgBucDiff**
BucSetInt-All	0.3683	0.4802	0.5232	0.5306	0.5351

Figure 19: Correlations of diversity scores from different sets of keywords generated from BucSetAll and BucSetInt

The keyword selection for email data set is described in details in Appendix D.

5.2.2 Diversity Scores on Email Data Set

The result from Wikipedia.org data set shows that the diversity scores generated by our diversity metrics are relevant to the diversity that we aim to measure. Therefore, we will apply the diversity metrics on the email data set in a similar way that we did on the Wikipedia.org data set. However, only some emails have bucket information, but the bucket-information-based diversity metrics require bucket information for all emails. In order to solve this problem, bucket information is generated for all emails. Emails with original bucket information do not require new assignments. For each bucket, a mean feature vector is computed to represent the main content of the bucket. An email without original bucket information is then assigned to the bucket whose mean vector is the closest to the feature vector of the email. This method generates missing bucket information based on the modeled contents so that the diversity scores can be computed.

The email data set has been processed and filtered into different sets. The relevant sets are BucSetAll and BucSetInt (as defined in Section 5.2.1) used in the keyword selection. The diversity scores of the employees are computed based on the feature vectors created from the set of 452,500 non-duplicated emails (EmSetAll), during the period in which we know the data to be robust, and the set of 45,217 non-duplicated internal emails (EmSetInt) during the same period. In order to compute the diversity scores of the employees, the contents of the emails are assigned to employees. There are multiple ways to assign the contents. The content of an email can be linked to the sender (outgoing emails: OUT), the recipients (incoming emails: INC), or both the sender and the recipients (both incoming and outgoing emails: IO). The table in Figure 20 summarizes the different sets of emails used in this study.

Emails for diversity scores Emails for keyword selection	EmSetInt	EmSetAll
BucSetInt	INC OUT IO	-
BucSetAll	INC OUT IO	INC OUT IO

Figure 20: Different sets of emails used in the computation of diversity scores

The table in Figure 21 shows the correlations of the diversity scores derived from our multiple diversity metrics. The high correlations confirm that our diversity metrics behave similarly on the email data set as we have encountered a similar result on the Wikipedia.org data set.

Correlation - MetricInt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	0.9695	1.0000			
AvgCommon	0.8010	0.7868	1.0000		
AvgCommonIC	0.8568	0.8161	0.9235	1.0000	
AvgBucDiff	0.7020	0.6720	0.9507	0.8787	1.0000

(a) Feature vectors generated from the keywords selected by BucSetInt

Correlation - MetricAll	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	0.9741	1.0000			
AvgCommon	0.8285	0.8379	1.0000		
AvgCommonIC	0.8269	0.8214	0.9406	1.0000	
AvgBucDiff	0.7104	0.7292	0.9201	0.8606	1.0000

(b) Feature vectors generated from the keywords selected by BucSetAll

Figure 21: Correlations of diversity scores across multiple diversity metrics

The diversity scores are computed on the set of all emails (EmSetAll) and on the set of internal emails (EmSetInt). EmSetInt beneficially excludes mass emails and junk emails that usually originate from sources outside the firm. These mass emails are likely to interfere with our analysis because they are sent to many people, but they usually do not contribute important contents to the recipients. On the other hand, one may also argue that EmSetAll contains all information the workers have obtained, including the information from the external sources. Therefore, both sets of emails are used to compute diversity scores, and the scores are compared. The table in Figure 22 shows the correlations of the diversity scores on EmSetAll and EmSetInt. The positive correlations confirm that our diversity rankings still follow the same trend. However, the correlations are only moderate, showing that the inclusion of the external emails does affect the diversity scores as it does in the keyword selection. The inclusion of the external emails brings a large number of additional emails and presumably a large amount of additional content that may or may not be related to the content produced inside the firm. Therefore, it is not surprising that the inclusion of the external emails affect diversity scores of the employees due to the different levels of their exposure to external emails.

Correlation - IntExt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
EmSetInt-All	0.4529	0.5560	0.5187	0.6462	0.3411

Figure 22: Correlations of diversity scores computed from all emails and only internal emails

There are many ways to model the information that the employees process through their emails. Outgoing emails are a good representative of the information that the employees deem to be important enough to pass to other people. However, in reality, an employee does not always need to send important information once he or she receives it. Therefore, the incoming emails usually capture more information arriving to the employee. However, incoming emails also contain more information that can interfere with our analysis such as junk emails and so on. To capture even more information, both incoming and outgoing emails can be used to compute diversity scores. The tables in Figure 23 show the correlations of the diversity scores computed from the above concepts.

Correlation - InOutInt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
IO-INC	0.8184	0.8449	0.7688	0.8030	0.7517
IO-OUT	0.8709	0.8886	0.9414	0.9413	0.9189
INC-OUT	0.5399	0.6062	0.5435	0.5986	0.4666

(a) Correlations of diversity scores on EmSetInt

Correlation - InOutAll	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
IO-INC	0.8274	0.8018	0.6565	0.6429	0.6847
IO-OUT	0.3220	0.4334	0.5037	0.6628	0.2864
INC-OUT	-0.0007	0.0619	0.1082	0.2270	-0.0254

(b) Correlations of diversity scores on EmSetAll

Figure 23: Correlations of diversity scores computed from incoming, outgoing, and both incoming and outgoing emails.

Figure 23(a) shows the correlations of scores on the internal emails (EmSetInt). The scores using both incoming and outgoing emails are highly correlated with the scores using only incoming emails and only outgoing emails. We believe that this effect is due to the fact that they share many emails. The scores using only incoming emails is moderately correlated with the scores using only outgoing emails. This result shows that there is a difference between the information one receives and sends.

Figure 23(b) shows the correlations of scores on all emails (EmSetAll). It yields an interesting result that is different from the result in Figure 23(a). The scores using both incoming and outgoing emails continue to be highly correlated with the scores using only incoming emails due to the large number of common emails. However, the scores using both incoming and outgoing emails is only moderately correlated with the scores using only outgoing emails. We believe that this effect is due to the inclusion of a large number of external emails that are mostly incoming emails from outside sources. The contents of the additional emails are also not likely to coincide with the original outgoing emails as the scores using only incoming emails show no correlation with the scores using only outgoing emails. When the external emails are included, the decision of using incoming email and/or outgoing emails becomes increasingly influential to the resulting diversity scores.

6. Discussion and Conclusion

The goal of our research is to identify and evaluate techniques for measuring the diversity of content in the email of information workers. The diversity scores derived from the measurement will be used in future productivity studies along with the results from social network analysis. In order to evaluate the diversity scores, many diversity metrics are defined based on different aspects of diversity. The metrics are evaluated against a set of articles from Wikipedia.org based on their manually assigned categories and subcategories. The results from the metrics on the Email data set are also compared. Our finding is that the rankings based on our diversity metrics follow similar trends. Also, the diversity metrics are able to successfully rank the diversity of content in selected Wikipedia.org articles.

Our topic model uses the probability distributions of the frequencies of words to represent topics. In order to represent the content of a document, we construct a feature vector based on the frequencies of keywords. Keywords need to be carefully selected so that the resulting feature vectors are able to represent the content of the associated documents well. Our method for keyword selection is based on the coefficient of variation of the mean frequencies across topics. The high correlations between the diversity scores and the manually-defined type in the triplet test on the Wikipedia.org data set in Figure 15 show that our method for keyword selection enables the automated selection of keywords that are useful in representing the contents of the documents in our data set. The high correlations in Figure 16 in the extended triplet test show that the diversity metrics provide sensible results when applied to several tests using different numbers of documents.

Our diversity metrics derived from different points of view exhibit similar results in diversity rankings as shown by the high correlations of the diversity scores across diversity metrics. Moreover, the diversity scores are able to predict the diversity ranking based on the manually assigned configuration types, proving that they can be used to rank diversity. The correlations of diversity scores across diversity metrics in the email data set also confirm the findings in the Wikipedia.org data set. These results satisfy our objective to find a set of diversity metrics that are developed around different aspects of diversity.

An interesting observation arising from the study is the effect of the inclusion of the external emails toward diversity rankings. The resulting diversity scores still show moderate correlations with the diversity scores computed without the external emails. However, the fact that the correlations are not high implies that the inclusion of the external emails does have an effect on both keyword selection and diversity ranking. The effect is likely due to the fact that the external emails contain a large amount of content in addition to the content of the internal emails.

Finally, this study has provided several sets of diversity scores of employees in the firm, derived from multiple diversity metrics on multiple sets of emails. Different sets of diversity scores are positively correlated, confirming the consistency of our results. The diversity scores will be studied further along with the results from social network analysis and the productivity data. The relationship between the results will provide further understanding about the effect of the different sets of emails used to compute diversity scores.

7. Limitations and Future Work

Some improvements can be applied toward the method in this study to improve the quality of the keyword selection and the computation of the diversity scores. Latent Semantic Indexing applies the use of singular value decomposition (SVD) on the document-term matrix generated from the term frequencies in the documents in order to algorithmically determine words with similar usages (Berry, Dumais, & O'Brien, 1995). Those words are then grouped together along the same dimension of the feature vectors in order to represent the same concept. Effectively, it reduces the dimensions of the feature vectors or, alternatively, increases the number of useful keywords if the number of dimensions remains the same.

Moreover, instead of using term frequencies as the elements of the feature vectors, many term weighting techniques as mentioned before can be applied. Although the effectiveness of the term weighting schemes in our application is still unknown, they are proven to be effective in document indexing in the way that they significantly increase the recall rate of the important documents (Salton & Buckley, 1996).

Appendix A: Wikipedia.org Data Set

The Wikipedia.org data set consists of 291 articles from Wikipedia. The articles have been categorized into three main categories, nine second-level categories, and 25 third-level categories as show in Figure 24.

Wikipedia's Categories

Figure 24: Categories of documents in Wikipedia.org data set

Articles in the Wikipedia.org data set are captured directly from Wikipedia.org in HTML format. An excerpt from an article, titled "Algorithmic learning theory" under categories: Computer science/Artificial intelligence/Machine learning, is shown in Figure 25. The syntaxes of HTML and Wikipedia are likely to interfere with our content analysis. Moreover, there are phrases that appear in essentially all articles but do not contribute to the content of the articles; for example, the phrase "From Wikipedia, the free encyclopedia" at the beginning of all articles. These issues potentially degrade the performance of our data representation. We rely on our keyword selection described in section 4.3 and Appendix C to identify words that are relevant to the contents of the articles, and the ones that are not relevant need to be excluded from the analysis.

Please read Wikipedia founder Jimmy Wales's personal appeal
<http://wikimediafoundation.org/wiki/Personal_Appeal>.

Algorithmic learning theory

From Wikipedia, the free encyclopedia.

Jump to: navigation <#column-one>, search <#searchInput>

Algorithmic learning theory (or *inductive inference*) is a framework
for machine learning </wiki/Machine_learning>.

The framework was introduced in E. Mark Gold
</w/index.php?title=E._Mark_Gold&action=edit>'s seminal paper "Language
identification in the limit
</wiki/Language_identification_in_the_limit>". The objective of language
identification </w/index.php?title=Language_identification&action=edit>
is for a machine running one program to be capable of developing another
program by which any given sentence can be tested to determine whether
it is "grammatical" or "ungrammatical". The language being learned need
not be English </wiki/English_language> or any other natural language
</wiki/Natural_language> - in fact the definition of "grammatical" can
be absolutely anything known to the tester.

Figure 25: An excerpt from an article in Wikipedia.org data set

Appendix B: Email Data Set and Data Processing

The email data set consists of 603,871 emails that are sent and received by the participating employees of the firm. With an aid of a semi-automatic clustering software called eClassifier, a previous study has performed clustering on this data set. Due to the difference between the current data set and the data set used at the time of the study, there is a clustering result for only 118,185 emails. The study was conducted before the contents of the emails are hashed to preserve the privacy of the firm and the employees, so the user intervention in the clustering by eClassifier is likely to result in an accurate clustering. This study uses the clustering result of the previous study in order to algorithmically select keywords to construct feature vectors that can well represent the contents of the emails. The emails have been clustered 11 times, each time with a different number of clusters: 2, 3, 4, 5, 8, 16, 20, 21, 50, 100, and 200. The clustering result forms a structure as in Figure 1.

Figure 2 confirms the existence of duplicated emails in the email data set. However, the duplication can be removed by eliminating additional emails with the same sender, recipients, and timestamps. Moreover, we also eliminate emails that share the same sender and timestamps, while their recipient list is a subset of the recipient list of others existing emails. This extra measure eliminates some additional duplicated emails with the special circumstance. The elimination of duplicated emails reduces the number emails in the data set to 521,316 non-duplicated emails, and the number of non-duplicated emails with bucket information is 110,979 emails.

As suggested in section 4.1, we separate internal emails and external emails to study the effect of the inclusion of external emails. Our criterion for an internal email is that it is sent by an employee and is received by at least one employee. This way the information is circulated within the firm, and the content of the internal email is likely to be related to the work of the firm. In the email data set, there are 59,294 non-duplicated internal emails, out of which, 20,252 emails have bucket information.

YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2002								557	3481	4969	6020	5307
2003	6037	3592	647	636	977	1072	1501	1817	3428	3639	4149	4538
2004	5205	2011										

Figure 26: The number of non-duplicated internal emails by months

Figure 3 and Figure 26 show the number of emails in each month from August 2002 to February 2004. The period between March 2003 and September 2003 contains significantly fewer emails than the other months. Both internal and external emails share the same trend, so it is not likely to be an effect of a decrease in external emails. This inconsistency is assumed to be caused by a failure of the capturing software on the corporate email server. In order not to let this inconsistency affect our analysis, the period of low email activities is excluded from the analysis. Specifically, our analysis includes emails during the periods from 1 October 2002 to 3 March 2003 and from 1 October 2003 to 10 February 2004. During the time period, there are 452,500 non-duplicated emails and 45,217 non-duplicated internal emails.

The set of 110,979 non-duplicated emails with bucket information is called BucSetAll, and the set of 20,252 non-duplicated internal emails with bucket information is called BucSetInt. We use these two email sets to generate keywords based on the bucket information. The reason for using both sets arises from the hypothesis that the inclusion of the external emails potentially has negative effect toward keyword selection due to the inclusion of email contents that are not related to the work of the employees. On the other hand, the larger set of emails will provide more contents for keyword selection. Therefore, both results are provided by this study and will be used for further analysis in future studies.

Similar to the sets of emails used for keyword selection, there are two sets of emails used to generate feature vectors and to compute diversity scores. During the period from 1 October 2002 to 3 March 2003 and the period from 1 October 2003 to 10 February 2004, the set of 452,500 non-duplicated emails is called EmSetAll, and the set of 45,217 non-duplicated internal emails is called EmSetInt. The reason for using the two sets of emails for computing diversity scores is also the same as the reason for the sets of emails for keyword selection, which is to study the effect of the inclusion of the external emails.

Appendix C: Keyword Selection and Results on Wikipedia.org Data Set

The articles in the Wikipedia.org data set is in the HTML format as described in Appendix A. There are unavoidably many fractions of words and words that are used of the syntaxes of HTML on the Wikipedia's website. In order to reduce the number of those words, we exclude the words that are shorter than three characters. Another problem that affects all content representations is the existence of synonyms and different word forms. Synonyms and different word forms cause two or more words to represent the same concept; for example, exclude, excludes, excluded, and exclusion have different spellings but represent the same meaning. Ideally, those words need to be grouped together under the same concept. One way to solve this issue is to create a massive list of synonyms and word forms. However, such method is extremely time-consuming and may not worth the effort in our application. Alternatively, we opt for a simple solution by removing a letter "s" from the end of all words. The reason for this method is that we hypothesize that most content-bearing words that we are interested in are in noun forms. By removing "s", we eliminate the most common plural indicator for nouns. This method has its flaws, but the resulting keywords in Figure 13 show many nouns that are likely to be affected positively by this method.

After implementing the methods mentioned above, we find that there exists 15,349 unique words in the Wikipedia.org data set. Out of these words, we decide to select approximately 400 keywords to represent the contents of the articles. Keywords are commonly selected based on their frequencies in the documents and over the set of all documents. In this study, the Wikipedia.org data set includes category information based on the categorization by Wikipedia. Similarly, the email data set contains the bucket information from the clustering results in a previous study. Therefore, we derive a method to algorithmically select keywords based on the category information or the bucket information in order to achieve a set of keywords that can well represent the contents of the documents.

The probability distributions of a keyword and a non-keyword in Figure 6 show the characteristics of the candidates for keywords. Variance is a common measurement of deviation of a set of observed data. The variance of the mean frequencies of the buckets potentially distinguishes keywords from non-keywords based on the observed frequencies from the probability distributions of the word across topics. Therefore, we initially explore the use of the variance of the mean frequencies of the buckets as a threshold for keyword selection. However, the chart in Figure 27 shows that the common words with high frequencies tend to have high variances. In addition, Zipf's law implies that the first few most frequent words have much higher frequencies than the other words. In general English text, the three most frequent words are "THE", "OF", and "AND". In our study, we decide not to include words with length less than three, so "OF" is not included in Figure 27, but it is noticeable that the frequency of "THE" is approximately three times of the frequency of "AND" as implied by Zipf's law. This observation indicates that, in most kinds of texts, there are always a few common words that occur with much higher frequencies than the other words, and the effect of the mean frequencies of the words on their variances is not negligible. In order to reduce this effect, we decide to use the squared value of the coefficient of variation as the threshold instead:

$$\text{Dinter} = \frac{1}{M^2} \sum_{b \in buckets} (m_b - M)^2 = \sum_{b \in buckets} \left(\frac{m_b}{M} - 1 \right)^2$$

The coefficient of variation effectively normalizes the mean frequencies of the probability distributions. The squared values of the coefficient of variations of the mean frequencies of the words in the Wikipedia.org data set are shown in the chart in Figure 28. The values from common words such as "AND" and "THE" is expectedly low so that they will be eliminated as non-keywords. Alternatively, we have used the value of variance over frequency as a threshold in order to accomplish the similar effect of reducing the influence of the mean frequencies over the variances. The results from this alternative approach are presented in Appendix E.

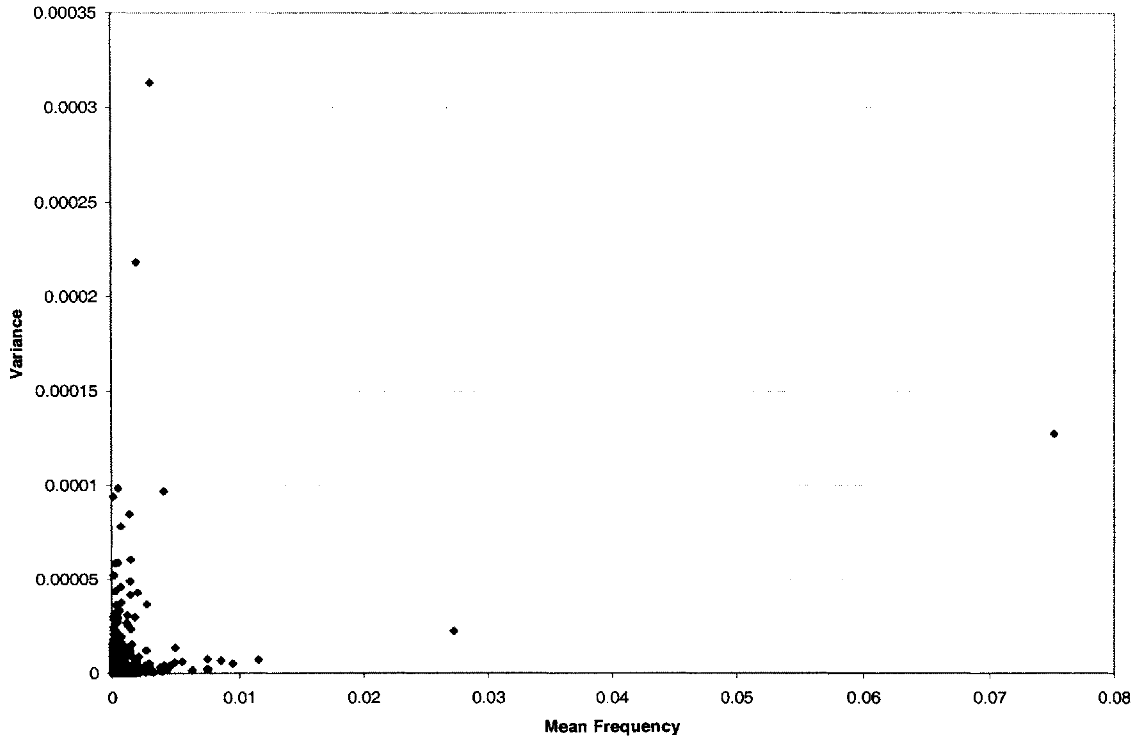


Figure 27: Variances and frequencies of the words in the Wikipedia.org data set

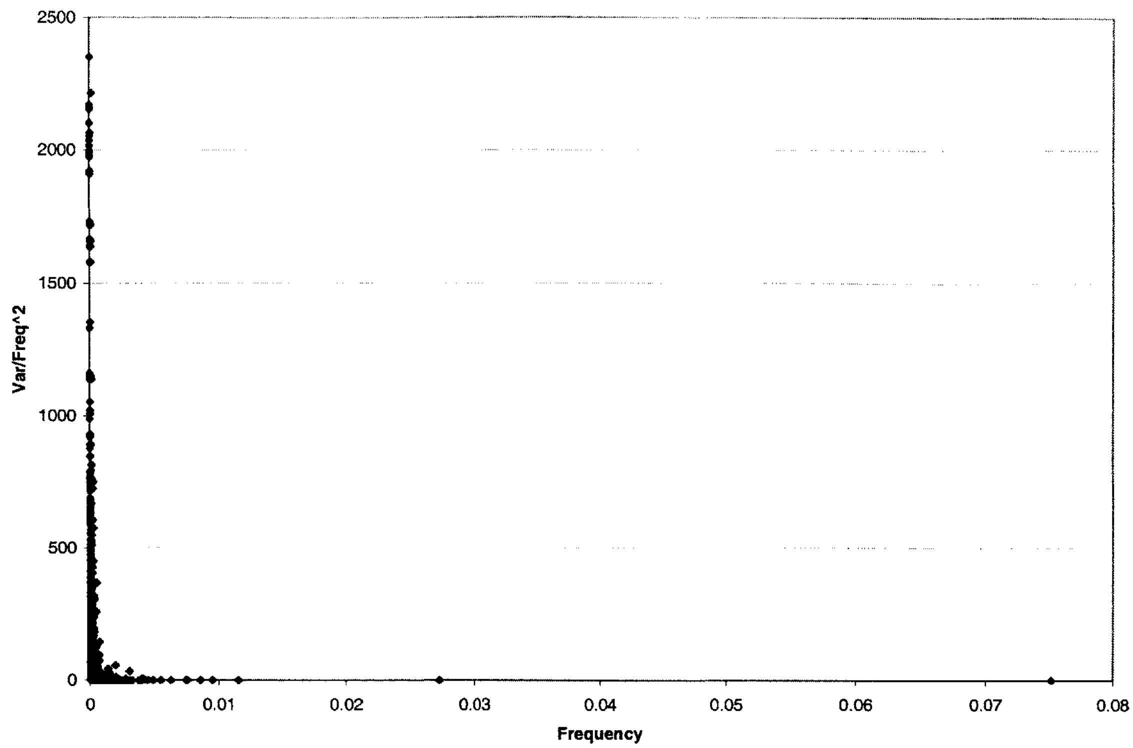


Figure 28: The squared values of the coefficients of variation of the words in Wikipedia.org data set

A downside of the coefficient of variation is that its value is very sensitive to words with low frequencies, which are the majority of the words in our data set. If we are to select words with high coefficients of variation as keywords, the resulting keywords would consist of many words with very low frequencies, which would affect the data representation negatively, as shown in Figure 29. Such rare words usually identify very few documents that contain one or a few instances of the words. Thus they are not good representatives of the main content of documents.

Alternatively, we identify keywords by excluding words whose coefficients of variation are lower than a certain threshold. In practice, we decide to keep a certain number of words with high Dinter. The number of remaining words is called the "threshold number." Then, out of the remaining words, we select an equal number of words with high frequencies from each category to compose the set of keywords. The number of words per categories is picked so that the number of the resulting keywords reaches a targeted amount: 400 in the Wikipedia.org data set.

Word	Frequency	Dinter
bielefeld	0.0000039	15980.97
cornelsen	0.0000079	15980.97
debe	0.0000039	15980.97
grosser	0.0000079	15980.97
historischer	0.0000079	15980.97
jungk	0.0000039	15980.97
kocher	0.0000039	15980.97
leipzig	0.0000039	15980.97
mielisch	0.0000039	15980.97
peip	0.0000039	15980.97

Figure 29: Frequencies and Dinter values of words sorted by descending Dinter values in the Wikipedia.org data set

In order to pick an appropriate threshold number, we use many sets of keywords resulted from different threshold numbers to construct feature vectors and compute the Adhesion and InvCohesion of the categories from the feature vectors. The result is shown in Figure 11. The sharp increase in Adhesion at the low threshold (high threshold number) reflects the exclusion of the two most frequent common words: "THE" and "AND". Although InvCohesion also increases, we consider the larger increase in Adhesion to be a positive effect. After the sharp increase, both Adhesion and InvCohesion increase at a diminishing rate. We hypothesize that the selection of thresholds in the rage of threshold with no major changes in Adhesion and InvCohesion would not have a large effect on the diversity ranking. In order to demonstrate this hypothesis, we pick three different threshold numbers: 13000, 14000, and 14500. The resulting diversity scores in the triplet test are shown in Figure 30. The correlations of the scores as shown in Figure 12 are high, confirming our hypothesis that, within an appropriated range of thresholds, the threshold does not have a large effect on diversity ranking.

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	6491	0.3275	0.3618	0.3437	0.7784	0.3106
1	36259	0.4094	0.4496	0.5608	0.8599	0.5113
2	11272	0.4595	0.5031	0.6777	0.9045	0.6201
3	109634	0.4390	0.4820	0.6806	0.9035	0.6370
4	203212	0.4838	0.5290	0.7796	0.9410	0.7287
5	102496	0.5004	0.5473	0.8465	0.9643	0.7974
6	323333	0.4364	0.4799	0.6889	0.9044	0.6451
7	589822	0.4828	0.5289	0.7941	0.9431	0.7417
8	1794402	0.5011	0.5486	0.8625	0.9677	0.8125
9	887864	0.5017	0.5499	0.8660	0.9685	0.8163

(a) Threshold number 13000

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	6491	0.3076	0.3404	0.3314	0.7741	0.2758
1	36259	0.3843	0.4249	0.5236	0.8531	0.4488
2	11272	0.4319	0.4767	0.6237	0.8939	0.5373
3	109634	0.4219	0.4650	0.6955	0.9067	0.6246
4	203212	0.4643	0.5105	0.7836	0.9428	0.7037
5	102496	0.4859	0.5333	0.8762	0.9730	0.7983
6	323333	0.4229	0.4670	0.6872	0.9031	0.6175
7	589822	0.4675	0.5148	0.7816	0.9421	0.7009
8	1794402	0.4902	0.5390	0.8759	0.9714	0.7970
9	887864	0.4924	0.5419	0.8694	0.9695	0.7917

(b) Threshold number 14000

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	6491	0.2704	0.3004	0.2955	0.7781	0.2294
1	36259	0.3640	0.4027	0.4862	0.8527	0.3945
2	11272	0.4184	0.4618	0.5973	0.8986	0.4906
3	109634	0.3942	0.4356	0.6372	0.8925	0.5314
4	203212	0.4444	0.4898	0.7335	0.9301	0.6149
5	102496	0.4642	0.5113	0.8103	0.9524	0.6872
6	323333	0.4021	0.4457	0.6729	0.9049	0.5767
7	589822	0.4552	0.5026	0.7645	0.9423	0.6545

8	1794402	0.4757	0.5248	0.8596	0.9680	0.7424
9	887864	0.4809	0.5310	0.8739	0.9737	0.7613

(c) Threshold number 14500

Figure 30: Results of the triplet test using keywords generated from different thresholds

We have used Dinter to measure the variations of the uses of words across categories in order to select words with high variations to be keywords because they are likely able to distinguish the contents of the documents between categories. Similarly, we define Dintra to represent the variations of the uses of words within the same categories. In order to be able to distinguish between the content of one category from the content of another category, a keyword needs to have not only high variation of frequencies across categories but also low variation of frequencies within categories. We find that most words with high Dintra are words with low frequencies as shown in Figure 31.

Word	Frequency	Dintra
cascade	0.0000158	2242089.87
impuritie	0.0000118	2242089.87
autocorrelation	0.0000039	1779680.01
biophysic	0.0000039	1689438.00
guideline	0.0000039	1689438.00
cutscene	0.0000039	1653901.33
indeo	0.0000197	1653901.33
ligo	0.0000079	1653901.33
owned	0.0000039	1653901.33
xanim	0.0000039	1653901.33

Figure 31: Frequencies and Dintra values of words sorted by descending Dintra values in the Wikipedia.org data set

Hypothetically, it is likely that rare words such as names only appear in a few documents. Although the rare words are good for representing the documents in which they appear, they are not suitable for representing the contents of the categories of the documents. Fortunately, our selection for words with high frequencies eliminates many words with high Dintra at the same time. Figure 32 shows the resulting keywords before the elimination of the words with high Dintra. The word "LFSR" (Linear Feedback Shift Register) possesses a significantly higher Dintra than the other keywords. Since the majority of the words with high Dintra are eliminated by the selection for high frequencies, we decide to keep eliminating the keyword with the highest Dintra value as long as the value is higher than the next highest Dintra value by more than 20 percents. In the case of Figure 32, we only eliminate LFSR because the Dintra value of ROBOCUP is higher than the value of BEAGLE by less than 20 percents.

Word	Dinter	Dintra
lfsr	1149.18	70772.83
robocup	288.53	22555.97
beagle	1139.48	21830.38
oracle	5236.23	20969.44
beacon	319.78	14664.64
lighthouse	319.78	13192.02
agp	451.36	12470.73
sin	153.90	12172.95
grand	155.95	11709.76
kerbero	362.69	10295.89
pound	35.16	9675.14
divx	750.16	9613.71
cracking	425.34	9117.03
reactor	426.31	8063.97
gaussian	53.05	6503.95
shadow	24.90	5536.69
neuron	53.00	5291.75
registration	54.61	5078.63
biomedical	238.78	5072.51
pilot	31.12	4892.55

Figure 32: The subset of keywords sorted by descending Dintra before the elimination of the words with high Dintra

The extended triplet test is designed to study the effect of the number of documents in the document sets toward the diversity scores of the document sets. We compute the diversity scores for three different sizes of document sets as shown in Figure 33. Figure 16 shows high correlations between the diversity scores. Figure 17 shows that the diversity scores increase at a diminishing rate as the size of the document set increases as expected.

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	6491	0.3076	0.3404	0.3314	0.7741	0.2758
1	36259	0.3843	0.4249	0.5236	0.8531	0.4488
2	11272	0.4319	0.4767	0.6237	0.8939	0.5373
3	109634	0.4219	0.4650	0.6955	0.9067	0.6246
4	203212	0.4643	0.5105	0.7836	0.9428	0.7037
5	102496	0.4859	0.5333	0.8762	0.9730	0.7983
6	323333	0.4229	0.4670	0.6872	0.9031	0.6175
7	589822	0.4675	0.5148	0.7816	0.9421	0.7009
8	1794402	0.4902	0.5390	0.8759	0.9714	0.7970
9	887864	0.4924	0.5419	0.8694	0.9695	0.7917

(a) 3-document test sets

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	5051	0.3814	0.4305	0.3318	0.7735	0.2772
1	46062	0.4646	0.5261	0.4885	0.8385	0.4180
2	17285	0.5200	0.5851	0.5694	0.8716	0.4888
3	139017	0.5062	0.5712	0.6194	0.8789	0.5524
4	310714	0.5549	0.6225	0.6924	0.9090	0.6179
5	156737	0.5774	0.6469	0.7653	0.9328	0.6928
6	409229	0.5024	0.5684	0.6115	0.8751	0.5454
7	897422	0.5522	0.6209	0.6876	0.9068	0.6123
8	2735050	0.5778	0.6489	0.7620	0.9300	0.6885
9	1351476	0.5786	0.6509	0.7557	0.9279	0.6833

(b) 6-document test sets

Type	NumTest	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
0	1763	0.4083	0.4658	0.3507	0.7802	0.2939
1	27384	0.4907	0.5619	0.4820	0.8361	0.4127
2	12929	0.5471	0.6240	0.5554	0.8663	0.4771
3	82253	0.5350	0.6102	0.6006	0.8722	0.5344
4	231402	0.5847	0.6635	0.6692	0.9005	0.5966
5	116717	0.6100	0.6911	0.7368	0.9226	0.6659
6	241855	0.5282	0.6045	0.5930	0.8683	0.5278
7	664214	0.5787	0.6585	0.6649	0.8985	0.5910

8	2028950	0.6065	0.6893	0.7347	0.9201	0.6625
9	1000984	0.6056	0.6895	0.7288	0.9181	0.6575

(c) 9-document test sets

Figure 33: The diversity scores resulted from the extended triplet test

Appendix D: Keyword Selection and Results on Email Data Set

Unlike the articles in the Wikipedia.org data set, the contents of the emails in the email data set are hashed. Therefore, we are unable to determine the meanings of the words in the emails. It is impossible to address the problem of synonyms and word forms. Moreover, we find that the email data set includes more than a million unique hashed words. The massive number of unique words is unlikely, compared to the limited amount of words that are commonly used in any languages. Fortunately, most of the million words occur only a few times. We hypothesize that the unlikely number of words is caused by occasionally mis-spelt words, fractions of words, and non-content-bearing words. During the keyword selection, we exclude the words with low frequencies in order to eliminate these undesirable words and also to reduce the number of words to a manageable amount.

We have mentioned in section 5.3 and Appendix B that we decide to use two sets of emails for keyword selection: BucSetAll and BucSetInt, and two sets of emails for feature vector creation and diversity score computation: EmSetAll and EmSetInt as shown in Figure 20. In BucSetAll, we exclude the words that occur fewer than 20 times. There remain 24,759 unique words that occur at least 20 times over the entire BucSetAll. Out of these words, we decide to select approximately 1,500 keywords to represent the contents of all emails. In BucSetInt, which includes fewer emails than BucSetAll, we exclude the words that occur fewer than 5 times. There remain 17,155 words that occur at least 5 times over BucSetInt. Out of these words, we also select approximately 1,500 keywords to represent the contents of all internal emails.

We select keywords on BucSetAll and BucSetInt using different threshold numbers and plot the values of Adhesion and InvCohesion of the buckets based on the keywords. The charts in Figure 18 show the effect of the threshold on Adhesion and InvCohesion for both sets of emails. Similar to the keyword selection in the Wikipedia.org data set, we select the threshold at the point that Adhesion and InvCohesion change slowly. The charts also show that the number of feature vectors created from the keywords decrease as the threshold increases, confirming our hypothesis that the words with high Dinter are likely to have low frequencies. Selecting words with high Dinter results in keywords with low frequencies, which are not likely to be able to represent the contents of all emails. This issue reminds us that it is not appropriated to select extremely high thresholds.

We decide to use the threshold number 24,000 for BucSetAll and the threshold number 15,000 for BucSetInt. We compute the diversity scores on EmSetAll and EmSetInt using the two sets of keywords. The results are shown in Figure 34, Figure 35, and Figure 36. Figure 19 shows the correlations of diversity scores on EmSetInt using the two sets of keywords. As suggested in section 5.1.3, the moderate correlations shows that the inclusion of external emails affect the quality of the keywords. Figure 22 shows the correlations of diversity scores on EmSetAll and EmSetInt using the same set of keywords generated on BucSetAll. Again, the moderate correlations confirm that the inclusion of external emails affects the diversity ranking.

In order to compare the effect of using incoming emails and outgoing emails to compute diversity scores of the employees, we compare the diversity scores based on both incoming and outgoing emails (IO), only incoming emails (INC), and only outgoing emails (OUT). Figure 23(a) shows the correlations of diversity scores on EmSetInt using keywords generated on BucSetAll based on IO, INC, and OUT. The high correlations in IO-INC and IO-OUT are likely due to the high number of overlapping emails between IO and INC, and between IO and OUT. The moderate correlations between INC and OUT indicate that there are differences between incoming emails and outgoing emails as discussed in section 5.3. Figure 23(b) shows the correlations of diversity scores on EmSetAll using keywords generated on BucSetAll based on IO, INC, and OUT. The correlations between IO and INC still remain high as most of the external emails are likely to be incoming emails, so the percentage of overlapping emails in IO and INC increases. The correlations between IO and OUT are only moderate due to the reduced percentage of overlapping emails. The most interesting result is that the diversity scores based on incoming emails show no correlation with the diversity scores based on outgoing emails. This result indicates that the inclusion of external emails eliminates the relationship between the contents of incoming emails and outgoing emails, which exists among internal emails.

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	1034	0.6848	0.8451	0.6665	0.9535	0.4554
2	980	0.6207	0.7276	0.6235	0.9522	0.4387
3	164	0.8050	0.9189	0.7419	0.9677	0.4902
4	704	0.8330	0.9421	0.7724	0.9763	0.5218
5	980	0.6207	0.7276	0.6235	0.9522	0.4387
6	704	0.8838	0.9608	0.7373	0.9727	0.4622
7	346	0.7266	0.8330	0.6956	0.9565	0.4445
8	435	0.3586	0.4391	0.4985	0.9030	0.3301
9	971	0.8333	0.9400	0.7084	0.9703	0.4699
10	370	0.7381	0.8746	0.6716	0.9525	0.4300
11	528	0.3895	0.4498	0.5260	0.9193	0.3793
12	14	0.5992	0.6854	0.8182	0.9776	0.5884
13	405	0.6207	0.8017	0.7340	0.9615	0.4870
14	632	0.4378	0.5358	0.4556	0.9129	0.3384
15	1337	0.7663	0.9006	0.7167	0.9548	0.4781
16	632	0.4378	0.5358	0.4556	0.9129	0.3384
17	1085	0.5402	0.7390	0.6325	0.9077	0.4337
18	749	0.7740	0.9121	0.7178	0.9629	0.4780
19	945	0.7138	0.8567	0.7810	0.9619	0.5460
20	1081	0.5952	0.7474	0.6596	0.9316	0.4388
21	313	0.6759	0.8429	0.7658	0.9719	0.5430
22	1093	0.6748	0.8069	0.6876	0.9601	0.4924
23	1018	0.5628	0.7277	0.5389	0.9154	0.3539
24	454	0.8159	0.9347	0.7791	0.9732	0.5366
25	1557	0.7814	0.9111	0.7366	0.9655	0.4961
26	81	0.5745	0.7458	0.5948	0.9243	0.3840
27	881	0.5831	0.7035	0.6393	0.9333	0.4217
28	375	0.5992	0.7278	0.6278	0.9366	0.4200
29	156	0.8820	0.9590	0.7588	0.9731	0.5099
30	115	0.8377	0.9333	0.7398	0.9694	0.5100
31	876	0.6434	0.7732	0.7084	0.9509	0.5205
32	194	0.6392	0.7518	0.7484	0.9502	0.5138
33	445	0.8086	0.9125	0.7743	0.9738	0.5437
34	2354	0.5194	0.7307	0.5075	0.9134	0.3308
35	488	0.6675	0.7963	0.7275	0.9631	0.5178
36	1280	0.7781	0.9133	0.7638	0.9689	0.5233
37	599	0.5472	0.7091	0.7274	0.9507	0.4800
38	1499	0.8093	0.9127	0.7737	0.9671	0.5565
39	251	0.6034	0.7588	0.7299	0.9506	0.5057
40	511	0.7043	0.8141	0.7236	0.9617	0.5502
41	548	0.5821	0.7321	0.5686	0.9137	0.3716
42	352	0.6949	0.8241	0.7721	0.9727	0.5458
43	450	0.8406	0.9095	0.7965	0.9657	0.5598
44	708	0.6624	0.7678	0.7341	0.9558	0.4956
45	611	0.8508	0.9047	0.7406	0.9690	0.5127
46	528	0.3895	0.4498	0.5260	0.9193	0.3793
47	138	0.4188	0.4899	0.5825	0.9320	0.3863

48	1160	0.8522	0.9491	0.7681	0.9740	0.5290
49	566	0.8485	0.9407	0.7771	0.9749	0.5392
50	545	0.7034	0.8329	0.7353	0.9634	0.5137
51	675	0.5436	0.6710	0.5913	0.9333	0.4208
52	449	0.5510	0.7349	0.6839	0.9541	0.4584
53	754	0.7521	0.8741	0.7064	0.9569	0.4748
54	1266	0.6285	0.8229	0.6820	0.9441	0.4683
55	1807	0.6121	0.7269	0.6616	0.9472	0.5104
56	526	0.5457	0.6427	0.6874	0.9473	0.4686
57	395	0.7954	0.9247	0.7777	0.9738	0.5279
58	772	0.4222	0.5369	0.5706	0.9174	0.4347
59	105	0.3940	0.4814	0.5709	0.9147	0.4038
60	409	0.7382	0.8824	0.7528	0.9740	0.4971
61	555	0.8524	0.9303	0.7639	0.9710	0.5065
62	1261	0.7659	0.9097	0.7270	0.9648	0.4895
63	926	0.7239	0.8771	0.7522	0.9655	0.5197
64	1684	0.7903	0.9187	0.7446	0.9650	0.5055
65	1667	0.8247	0.9239	0.7500	0.9638	0.5080
66	486	0.4611	0.5193	0.6680	0.9352	0.4745
67	483	0.5992	0.7398	0.7470	0.9592	0.5096

(a) Diversity scores on EmSetInt using keywords generated on BucSetInt

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	2554	0.5567	0.6742	0.6699	0.9529	0.4421
2	22	0.5947	0.7066	0.7166	0.9672	0.4572
3	491	0.6988	0.8404	0.7429	0.9721	0.4591
4	1444	0.5460	0.6509	0.6543	0.9577	0.4264
5	2139	0.5417	0.6748	0.6636	0.9517	0.4511
6	1730	0.6625	0.8000	0.7340	0.9704	0.4496
7	616	0.6288	0.7639	0.6663	0.9552	0.4174
8	440	0.4284	0.5295	0.5444	0.9052	0.3396
9	1882	0.6026	0.7195	0.7077	0.9671	0.4534
10	760	0.6378	0.7795	0.6550	0.9549	0.4060
11	1885	0.4619	0.5577	0.6392	0.9459	0.4158
12	20	0.6779	0.7857	0.8100	0.9801	0.5399
13	413	0.7267	0.7952	0.7400	0.9686	0.4635
14	726	0.5755	0.6635	0.6629	0.9430	0.4251
15	2705	0.5098	0.6165	0.5697	0.9221	0.3539
16	2657	0.3993	0.4745	0.3799	0.9079	0.2369
17	1859	0.5639	0.7481	0.7045	0.9429	0.4753
18	1727	0.5706	0.6969	0.7133	0.9632	0.4378
19	975	0.6051	0.7305	0.7222	0.9649	0.4681
20	2127	0.5423	0.6650	0.6902	0.9572	0.4334
21	611	0.7143	0.8521	0.7620	0.9761	0.5072
22	2386	0.5713	0.7104	0.6808	0.9547	0.4360
23	2353	0.6179	0.7439	0.7274	0.9655	0.4516
24	932	0.6546	0.7997	0.7616	0.9714	0.4729
25	3851	0.5610	0.6990	0.6663	0.9483	0.4297
26	123	0.6234	0.7804	0.6836	0.9594	0.4246

27	1650	0.6276	0.7551	0.6812	0.9558	0.4134
28	694	0.5886	0.7182	0.6638	0.9567	0.4307
29	371	0.6563	0.7957	0.7354	0.9714	0.4659
30	182	0.6496	0.7646	0.7350	0.9670	0.4755
31	1637	0.6022	0.7207	0.7317	0.9630	0.5027
32	205	0.5845	0.7015	0.7526	0.9658	0.4883
33	822	0.7067	0.8447	0.7578	0.9747	0.4875
34	3196	0.6651	0.8193	0.6987	0.9583	0.4372
35	1034	0.6671	0.8165	0.7257	0.9663	0.4883
36	2242	0.5463	0.6669	0.6999	0.9623	0.4451
37	1348	0.5500	0.7097	0.7051	0.9649	0.4598
38	3124	0.4475	0.5313	0.4669	0.9156	0.3340
39	437	0.6546	0.7774	0.7622	0.9757	0.4959
40	1222	0.6362	0.7771	0.7470	0.9715	0.5205
41	1036	0.6752	0.7926	0.6920	0.9581	0.4188
42	792	0.7279	0.8595	0.7629	0.9743	0.5019
43	545	0.5478	0.6336	0.6608	0.9537	0.4292
44	1513	0.5891	0.7164	0.6977	0.9601	0.4182
45	1562	0.6507	0.7919	0.6936	0.9583	0.4650
46	719	0.4642	0.5601	0.5725	0.9303	0.3971
47	140	0.4654	0.5510	0.5822	0.9425	0.3865
48	2083	0.6173	0.7668	0.6986	0.9603	0.4490
49	1352	0.5451	0.6632	0.6292	0.9454	0.3908
50	963	0.6310	0.7792	0.7096	0.9627	0.4655
51	1465	0.5332	0.6491	0.6287	0.9363	0.4022
52	899	0.6771	0.8150	0.7423	0.9727	0.4833
53	1640	0.6525	0.7816	0.6975	0.9630	0.4159
54	1332	0.6728	0.8121	0.7534	0.9719	0.4950
55	3428	0.5680	0.6947	0.6755	0.9540	0.4975
56	1047	0.6444	0.7791	0.7176	0.9611	0.4608
57	848	0.6072	0.7347	0.7557	0.9743	0.4847
58	1398	0.5400	0.6683	0.6703	0.9593	0.4910
59	148	0.4656	0.5329	0.6155	0.9444	0.3976
60	1052	0.6714	0.8094	0.7430	0.9733	0.4557
61	603	0.5724	0.6907	0.7181	0.9688	0.4472
62	2286	0.5747	0.7190	0.6949	0.9577	0.4486
63	968	0.6898	0.8065	0.7387	0.9666	0.4726
64	3312	0.6077	0.7555	0.7312	0.9679	0.4679
65	3297	0.6073	0.7737	0.7239	0.9645	0.4710
66	489	0.6742	0.8076	0.7426	0.9694	0.4943
67	524	0.6370	0.7586	0.7253	0.9654	0.4699

(b) Diversity scores on EmSetInt using keywords generated on BucSetAll

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	6167	0.5815	0.6993	0.6767	0.9440	0.4573
2	25	0.6087	0.7238	0.7085	0.9656	0.4506
3	3999	0.7954	0.8937	0.8016	0.9716	0.5363
4	24307	0.7286	0.8614	0.6647	0.9425	0.4487
5	14951	0.7432	0.8690	0.7863	0.9658	0.5687

6	4197	0.6949	0.8389	0.7594	0.9727	0.4761
7	4736	0.6087	0.7551	0.6040	0.9378	0.3697
8	1428	0.6727	0.8012	0.7688	0.9572	0.5307
9	7152	0.6659	0.8097	0.7918	0.9748	0.5249
10	11935	0.7958	0.8989	0.7607	0.9614	0.5187
11	4194	0.7897	0.8419	0.7860	0.9655	0.5639
12	73	0.7966	0.8690	0.7454	0.9591	0.4924
13	1454	0.8038	0.8769	0.7905	0.9763	0.5107
14	2654	0.5475	0.6676	0.6967	0.9446	0.4543
15	7722	0.5414	0.6549	0.6209	0.9312	0.4054
16	16408	0.5777	0.6688	0.6714	0.9507	0.4884
17	14638	0.7168	0.8378	0.6887	0.9467	0.4680
18	2912	0.6800	0.7972	0.7873	0.9728	0.5160
19	2501	0.8325	0.8516	0.7630	0.9677	0.5176
20	11241	0.7183	0.8386	0.8012	0.9694	0.5704
21	4992	0.7356	0.8749	0.7872	0.9763	0.5046
22	7068	0.6195	0.7791	0.7228	0.9591	0.4688
23	7869	0.7287	0.8628	0.8123	0.9732	0.5577
24	6355	0.7642	0.8961	0.8068	0.9718	0.5404
25	8378	0.6040	0.7528	0.7249	0.9560	0.4799
26	139	0.6575	0.8140	0.7268	0.9648	0.4649
27	6467	0.6163	0.7504	0.6634	0.9497	0.4011
28	10037	0.7081	0.8381	0.7769	0.9673	0.5257
29	551	0.6806	0.8142	0.7391	0.9722	0.4711
30	2904	0.7780	0.8550	0.8020	0.9695	0.5472
31	7550	0.6278	0.7472	0.7676	0.9667	0.5282
32	1048	0.6978	0.8163	0.6995	0.9479	0.4752
33	2646	0.8081	0.8944	0.7908	0.9775	0.5191
34	5982	0.7066	0.8599	0.7473	0.9643	0.4769
35	8840	0.7832	0.9051	0.8108	0.9717	0.5793
36	12590	0.7554	0.8611	0.8058	0.9709	0.5653
37	1657	0.5701	0.7199	0.7072	0.9649	0.4612
38	6619	0.5031	0.6008	0.5492	0.9291	0.3923
39	6825	0.7555	0.8729	0.8102	0.9733	0.5504
40	11926	0.7963	0.8776	0.6972	0.9493	0.4839
41	6706	0.6797	0.8216	0.6949	0.9572	0.4252
42	3178	0.7531	0.8856	0.7800	0.9752	0.5084
43	1010	0.6627	0.7838	0.7851	0.9715	0.5402
44	5922	0.6483	0.7799	0.7356	0.9658	0.4555
45	5497	0.6492	0.8027	0.6992	0.9544	0.4697
46	15466	0.7350	0.8607	0.6906	0.9478	0.4874
47	1650	0.7441	0.8694	0.8023	0.9699	0.5645
48	7276	0.6478	0.7996	0.7158	0.9599	0.4588
49	7824	0.6939	0.8077	0.7546	0.9601	0.5107
50	6677	0.7294	0.8644	0.7765	0.9685	0.5233
51	16030	0.7969	0.8779	0.7761	0.9623	0.5316
52	1247	0.6946	0.8318	0.7564	0.9749	0.4931
53	5355	0.6602	0.8013	0.7070	0.9654	0.4277

54	4229	0.7745	0.8792	0.8118	0.9727	0.5623
55	20406	0.6975	0.8245	0.7858	0.9658	0.5844
56	12191	0.7571	0.8878	0.7981	0.9689	0.5456
57	7886	0.7226	0.8497	0.8099	0.9745	0.5442
58	19389	0.7458	0.8758	0.7651	0.9617	0.5451
59	150	0.4722	0.5397	0.6232	0.9458	0.4026
60	3345	0.7842	0.8613	0.7880	0.9767	0.5048
61	2715	0.5779	0.6966	0.7513	0.9696	0.4873
62	6709	0.7064	0.8129	0.7287	0.9602	0.4949
63	2521	0.7487	0.8528	0.8034	0.9725	0.5343
64	8699	0.7343	0.8375	0.7704	0.9721	0.5044
65	6743	0.7490	0.8876	0.8020	0.9715	0.5559
66	3023	0.7518	0.8482	0.7803	0.9731	0.4998
67	2783	0.6604	0.8009	0.7602	0.9616	0.5275

(c) Diversity scores on EmSetAll using keywords generated on BucSetAll

Figure 34: Diversity scores based on both incoming and outgoing emails (IO)

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	1457	0.6425	0.7647	0.7033	0.9573	0.4646
2	11	0.4741	0.5907	0.5851	0.9307	0.3964
3	359	0.7200	0.8589	0.7111	0.9649	0.4354
4	590	0.8008	0.9117	0.7810	0.9765	0.5100
5	1132	0.6999	0.8308	0.7071	0.9659	0.4845
6	1113	0.8120	0.9238	0.7618	0.9754	0.4657
7	357	0.6542	0.7910	0.6807	0.9485	0.4143
8	188	0.4645	0.5586	0.5527	0.9242	0.3566
9	1162	0.8206	0.9342	0.7474	0.9745	0.4627
10	409	0.6683	0.8111	0.6896	0.9571	0.4382
11	1037	0.6091	0.7408	0.6665	0.9516	0.4368
12	13	0.6100	0.6907	0.7867	0.9737	0.5243
13	294	0.6403	0.6778	0.7097	0.9593	0.4326
14	457	0.6837	0.7577	0.6641	0.9573	0.4491
15	1426	0.7433	0.8760	0.7116	0.9565	0.4455
16	1133	0.8216	0.9357	0.7560	0.9750	0.4794
17	590	0.6528	0.8240	0.7197	0.9502	0.4747
18	941	0.7732	0.8916	0.7292	0.9617	0.4553
19	548	0.7228	0.8581	0.7532	0.9641	0.4893
20	1260	0.6192	0.7644	0.6602	0.9390	0.4133
21	378	0.7628	0.8944	0.7831	0.9775	0.5272
22	1141	0.6823	0.8284	0.7027	0.9645	0.4632
23	1453	0.6690	0.8251	0.6970	0.9538	0.4280
24	512	0.6613	0.8022	0.7461	0.9633	0.4741
25	2151	0.7007	0.8606	0.6986	0.9603	0.4511
26	76	0.5848	0.7388	0.6496	0.9263	0.4274
27	1057	0.6741	0.8111	0.6992	0.9527	0.4332
28	375	0.7091	0.8411	0.7430	0.9676	0.4874
29	344	0.8142	0.9315	0.7522	0.9699	0.4619

30	100	0.8138	0.9200	0.7665	0.9729	0.4969
31	967	0.6232	0.7958	0.7258	0.9453	0.4987
32	134	0.6420	0.7561	0.7602	0.9570	0.4974
33	478	0.8120	0.9228	0.7764	0.9761	0.5172
34	1762	0.6867	0.8636	0.6318	0.9448	0.3973
35	542	0.6890	0.8207	0.7574	0.9671	0.5107
36	1229	0.6917	0.8354	0.7490	0.9681	0.4786
37	645	0.6026	0.7445	0.7520	0.9670	0.4862
38	1672	0.7984	0.9143	0.7657	0.9717	0.5199
39	277	0.7037	0.8465	0.7673	0.9676	0.5061
40	850	0.7676	0.8929	0.7669	0.9716	0.5366
41	637	0.6865	0.8235	0.6860	0.9439	0.4220
42	504	0.7404	0.8701	0.7603	0.9742	0.5077
43	318	0.8346	0.9269	0.7755	0.9663	0.4955
44	756	0.6389	0.7683	0.6860	0.9526	0.4058
45	776	0.7562	0.8844	0.7376	0.9669	0.4972
46	365	0.6283	0.7421	0.7073	0.9581	0.4890
47	85	0.6459	0.7553	0.7406	0.9672	0.4789
48	1079	0.8380	0.9462	0.7776	0.9761	0.5053
49	729	0.6272	0.7526	0.6603	0.9546	0.4200
50	542	0.7057	0.8451	0.7152	0.9585	0.4882
51	738	0.6840	0.8223	0.7141	0.9603	0.4681
52	536	0.6750	0.8419	0.7423	0.9700	0.4704
53	933	0.7512	0.8761	0.7225	0.9606	0.4405
54	622	0.7104	0.8613	0.7385	0.9614	0.4806
55	1831	0.7192	0.8450	0.7479	0.9634	0.5370
56	671	0.7055	0.8297	0.7447	0.9654	0.4917
57	479	0.6142	0.7269	0.7652	0.9709	0.4974
58	587	0.6198	0.7299	0.7015	0.9578	0.4994
59	80	0.3810	0.4693	0.5654	0.9244	0.3344
60	735	0.8151	0.9282	0.7602	0.9738	0.4724
61	293	0.7030	0.8178	0.6901	0.9607	0.4196
62	1168	0.8044	0.9224	0.7533	0.9718	0.4799
63	544	0.8081	0.9084	0.7634	0.9710	0.4897
64	1646	0.7043	0.8558	0.7291	0.9673	0.4594
65	1743	0.6833	0.8410	0.6669	0.9556	0.4274
66	258	0.6898	0.7894	0.7408	0.9633	0.5062
67	347	0.7256	0.8434	0.7712	0.9686	0.4920

(a) Diversity scores on EmSetInt using keywords generated on BucSetInt

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	1491	0.5932	0.7073	0.6868	0.9605	0.4530
2	11	0.5418	0.6361	0.6694	0.9560	0.4403
3	362	0.7182	0.8461	0.7517	0.9728	0.4652
4	598	0.6432	0.7537	0.7318	0.9712	0.4867
5	1147	0.5976	0.7262	0.7056	0.9610	0.4861
6	1121	0.6691	0.8034	0.7389	0.9717	0.4548
7	356	0.6538	0.7797	0.7031	0.9634	0.4395
8	190	0.4731	0.5644	0.5767	0.9208	0.3663

9	1177	0.6054	0.7199	0.6977	0.9661	0.4519
10	410	0.6787	0.8147	0.6952	0.9616	0.4438
11	1045	0.5157	0.6148	0.6700	0.9549	0.4347
12	14	0.6550	0.7642	0.7732	0.9792	0.4987
13	295	0.6489	0.7065	0.7201	0.9618	0.4473
14	457	0.6433	0.7344	0.7074	0.9550	0.4561
15	1433	0.6134	0.7261	0.6880	0.9528	0.4320
16	1143	0.5271	0.6136	0.5739	0.9430	0.3756
17	590	0.6352	0.7817	0.7181	0.9552	0.4786
18	946	0.6560	0.7715	0.7320	0.9674	0.4553
19	550	0.6704	0.7942	0.7508	0.9712	0.4867
20	1268	0.5777	0.6996	0.6993	0.9570	0.4432
21	378	0.7272	0.8592	0.7594	0.9758	0.5062
22	1150	0.6207	0.7568	0.7036	0.9617	0.4565
23	1462	0.6124	0.7357	0.7149	0.9616	0.4460
24	516	0.6776	0.8130	0.7630	0.9730	0.4868
25	2172	0.6203	0.7510	0.7092	0.9601	0.4596
26	76	0.5848	0.7394	0.6091	0.9442	0.3718
27	1059	0.6636	0.7808	0.7008	0.9614	0.4260
28	377	0.6461	0.7738	0.7139	0.9680	0.4642
29	345	0.6585	0.7964	0.7350	0.9713	0.4665
30	101	0.6894	0.7948	0.7534	0.9705	0.4891
31	982	0.6223	0.7839	0.7287	0.9593	0.5030
32	134	0.6012	0.7273	0.7439	0.9655	0.4847
33	492	0.7241	0.8591	0.7619	0.9753	0.4951
34	1783	0.6874	0.8285	0.7149	0.9616	0.4555
35	547	0.6960	0.8342	0.7441	0.9709	0.4970
36	1239	0.5900	0.7100	0.7109	0.9658	0.4600
37	648	0.5851	0.7192	0.7228	0.9692	0.4711
38	1713	0.5922	0.6888	0.6311	0.9492	0.4463
39	278	0.6586	0.7758	0.7580	0.9747	0.4981
40	858	0.6830	0.8193	0.7579	0.9737	0.5258
41	640	0.6644	0.7747	0.6896	0.9565	0.4203
42	507	0.7122	0.8481	0.7551	0.9728	0.4984
43	338	0.5651	0.6464	0.6787	0.9562	0.4395
44	760	0.6339	0.7477	0.6962	0.9614	0.4205
45	780	0.6673	0.8081	0.7116	0.9628	0.4823
46	366	0.6071	0.7246	0.7129	0.9613	0.4907
47	85	0.5953	0.7004	0.7039	0.9652	0.4734
48	1088	0.6378	0.7764	0.7112	0.9647	0.4649
49	730	0.6083	0.7287	0.6811	0.9586	0.4320
50	549	0.6575	0.8068	0.7249	0.9644	0.4843
51	740	0.5798	0.6906	0.6703	0.9486	0.4375
52	537	0.6779	0.8204	0.7455	0.9729	0.4871
53	935	0.6831	0.8028	0.7093	0.9635	0.4282
54	634	0.6724	0.8076	0.7473	0.9700	0.4880
55	1873	0.6315	0.7550	0.7184	0.9654	0.5236
56	672	0.6985	0.8095	0.7312	0.9649	0.4721

57	485	0.6619	0.7888	0.7618	0.9755	0.4983
58	606	0.6016	0.7267	0.6812	0.9609	0.4927
59	81	0.3953	0.4575	0.5329	0.9314	0.3403
60	739	0.7109	0.8436	0.7496	0.9742	0.4635
61	299	0.5989	0.7056	0.7097	0.9674	0.4412
62	1186	0.6303	0.7631	0.7291	0.9662	0.4698
63	553	0.7129	0.8318	0.7608	0.9720	0.4870
64	1669	0.6457	0.7798	0.7383	0.9692	0.4737
65	1773	0.6306	0.7855	0.7144	0.9633	0.4640
66	258	0.6733	0.8125	0.7268	0.9668	0.4958
67	361	0.6618	0.7785	0.7391	0.9685	0.4750

(b) Diversity scores on EmSetInt using keywords generated on BucSetAll

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	3128	0.6381	0.7533	0.7122	0.9560	0.4892
2	13	0.5577	0.6587	0.6550	0.9516	0.4276
3	3259	0.8104	0.8932	0.7838	0.9664	0.5275
4	22077	0.7176	0.8504	0.6071	0.9319	0.3914
5	11187	0.8004	0.9132	0.7351	0.9565	0.5159
6	2785	0.7393	0.8661	0.7786	0.9760	0.4963
7	2634	0.6619	0.8091	0.6730	0.9523	0.4248
8	969	0.7378	0.8423	0.7586	0.9595	0.5179
9	5320	0.6979	0.8330	0.8059	0.9765	0.5456
10	9310	0.8217	0.9099	0.6975	0.9498	0.4672
11	3334	0.8613	0.8997	0.7611	0.9604	0.5392
12	65	0.7938	0.8646	0.7219	0.9550	0.4753
13	1094	0.8130	0.8824	0.7916	0.9756	0.5092
14	1644	0.6865	0.8033	0.7711	0.9660	0.5072
15	4459	0.6609	0.7871	0.7428	0.9611	0.4929
16	10201	0.7313	0.8361	0.7843	0.9675	0.5647
17	12527	0.6917	0.8150	0.6165	0.9337	0.3988
18	2100	0.7520	0.8607	0.8064	0.9750	0.5390
19	1486	0.8558	0.8707	0.8001	0.9736	0.5479
20	8388	0.7736	0.8941	0.7672	0.9618	0.5376
21	3183	0.7820	0.8954	0.8037	0.9767	0.5221
22	3896	0.7169	0.8526	0.7696	0.9686	0.5108
23	6006	0.7506	0.8813	0.7949	0.9685	0.5509
24	5043	0.7919	0.9140	0.7906	0.9677	0.5354
25	4779	0.6980	0.8237	0.7672	0.9664	0.5143
26	91	0.6390	0.7922	0.6929	0.9566	0.4463
27	4075	0.6664	0.7989	0.6928	0.9580	0.4233
28	6521	0.7837	0.9007	0.7840	0.9669	0.5352
29	508	0.6815	0.8129	0.7386	0.9720	0.4705
30	2183	0.7547	0.8288	0.7703	0.9614	0.5193
31	4224	0.7598	0.8741	0.8055	0.9749	0.5631
32	952	0.6911	0.8071	0.6609	0.9404	0.4411
33	1807	0.8761	0.9262	0.8111	0.9788	0.5429
34	3409	0.7485	0.8811	0.7799	0.9701	0.5069
35	6445	0.8001	0.9145	0.7754	0.9644	0.5479

36	9616	0.8222	0.9024	0.7820	0.9658	0.5497
37	947	0.5971	0.7184	0.7172	0.9673	0.4662
38	3660	0.6772	0.7897	0.7277	0.9642	0.5116
39	5113	0.7868	0.9018	0.7954	0.9688	0.5437
40	11181	0.7964	0.8755	0.6671	0.9438	0.4545
41	4207	0.7204	0.8505	0.7208	0.9620	0.4505
42	1866	0.7587	0.8897	0.7925	0.9746	0.5245
43	700	0.6997	0.8153	0.8027	0.9732	0.5568
44	3571	0.7241	0.8466	0.7624	0.9702	0.4887
45	2893	0.6832	0.8344	0.7348	0.9614	0.5018
46	13278	0.7522	0.8774	0.6254	0.9369	0.4118
47	1463	0.7686	0.8902	0.7897	0.9672	0.5444
48	3958	0.7422	0.8500	0.7648	0.9694	0.5046
49	5080	0.8133	0.8908	0.7754	0.9646	0.5318
50	4541	0.7798	0.8827	0.7990	0.9709	0.5470
51	11826	0.8567	0.9117	0.7510	0.9591	0.5142
52	885	0.6974	0.8374	0.7619	0.9753	0.4982
53	3362	0.6986	0.8336	0.7233	0.9674	0.4430
54	3055	0.7754	0.8701	0.7721	0.9632	0.5320
55	13307	0.7929	0.9058	0.7752	0.9640	0.5586
56	9973	0.7880	0.9104	0.7723	0.9642	0.5251
57	5783	0.7883	0.9040	0.7947	0.9694	0.5408
58	14222	0.7528	0.8781	0.6584	0.9416	0.4406
59	83	0.4090	0.4717	0.5510	0.9344	0.3526
60	2494	0.8159	0.8726	0.8012	0.9773	0.5216
61	1783	0.6045	0.7131	0.7495	0.9673	0.4960
62	3900	0.8010	0.8600	0.7809	0.9697	0.5373
63	1835	0.7476	0.8449	0.8054	0.9708	0.5348
64	4893	0.8306	0.8854	0.7999	0.9747	0.5334
65	4490	0.7660	0.8883	0.7707	0.9638	0.5353
66	1736	0.7817	0.8693	0.8014	0.9740	0.5254
67	1914	0.6974	0.8086	0.7594	0.9604	0.5278

(c) Diversity scores on EmSetAll using keywords generated on BucSetAll

Figure 35: Diversity scores based only on incoming emails (INC)

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	1070	0.4236	0.5643	0.4973	0.9053	0.3086
2	10	0.6669	0.7760	0.7434	0.9613	0.4759
3	134	0.4967	0.6949	0.6433	0.9586	0.4134
4	870	0.6923	0.8299	0.7646	0.9725	0.4777
5	1027	0.5331	0.6728	0.5826	0.9414	0.3932
6	613	0.8015	0.9161	0.7332	0.9687	0.4461
7	267	0.5054	0.6559	0.5854	0.9164	0.3610
8	260	0.3547	0.4643	0.4678	0.8893	0.2896
9	744	0.8139	0.9256	0.7543	0.9726	0.4636
10	365	0.5373	0.6953	0.5838	0.9301	0.3413
11	840	0.4534	0.5840	0.5675	0.9231	0.3759

12	7	0.5073	0.6171	0.8182	0.9395	0.5898
13	237	0.5547	0.5847	0.6603	0.9480	0.3951
14	432	0.5824	0.6593	0.5209	0.9253	0.3647
15	1289	0.5878	0.7548	0.4463	0.8966	0.2650
16	1483	0.8302	0.9341	0.7137	0.9679	0.4392
17	1280	0.4104	0.5580	0.5482	0.8687	0.3760
18	780	0.6318	0.7965	0.7226	0.9614	0.4539
19	467	0.5858	0.7526	0.7161	0.9501	0.4679
20	1249	0.4541	0.5910	0.5316	0.8990	0.3228
21	256	0.6795	0.8479	0.7916	0.9754	0.5349
22	1228	0.5752	0.7280	0.6492	0.9510	0.4215
23	1025	0.5959	0.7610	0.6809	0.9514	0.4190
24	443	0.5393	0.7164	0.7176	0.9509	0.4332
25	1687	0.5964	0.8001	0.5688	0.9290	0.3584
26	55	0.6071	0.7768	0.6349	0.9385	0.4075
27	684	0.4987	0.6207	0.6368	0.9241	0.3852
28	317	0.5863	0.7286	0.6594	0.9447	0.4198
29	26	0.7248	0.8635	0.7684	0.9714	0.4721
30	84	0.7703	0.9001	0.7161	0.9618	0.4489
31	703	0.5125	0.6176	0.6318	0.9448	0.4371
32	71	0.4957	0.5998	0.6664	0.9219	0.4120
33	332	0.6968	0.8462	0.7459	0.9678	0.4832
34	1448	0.5820	0.7749	0.5587	0.9289	0.3233
35	497	0.5282	0.6808	0.6833	0.9518	0.4646
36	1307	0.5634	0.7229	0.6977	0.9536	0.4229
37	700	0.4716	0.6551	0.7118	0.9411	0.4478
38	1338	0.8262	0.9422	0.7347	0.9648	0.4810
39	208	0.6375	0.7977	0.7343	0.9595	0.4774
40	369	0.6450	0.7914	0.7065	0.9548	0.5094
41	400	0.6379	0.7602	0.6715	0.9403	0.4033
42	302	0.6836	0.8186	0.7756	0.9743	0.5065
43	201	0.8386	0.9166	0.8047	0.9663	0.5264
44	785	0.5023	0.6275	0.6758	0.9435	0.3991
45	768	0.7226	0.8554	0.6924	0.9546	0.4558
46	377	0.3122	0.3960	0.3641	0.8782	0.2420
47	59	0.2477	0.2991	0.5003	0.8967	0.3034
48	944	0.8567	0.9555	0.7658	0.9722	0.4968
49	651	0.4545	0.5692	0.4942	0.9192	0.3126
50	514	0.7107	0.8617	0.7428	0.9619	0.4960
51	737	0.5557	0.7193	0.6220	0.9369	0.3939
52	362	0.5962	0.7631	0.7280	0.9660	0.4525
53	740	0.7119	0.8453	0.6748	0.9532	0.4105
54	702	0.6532	0.8220	0.7136	0.9539	0.4731
55	1594	0.5247	0.6503	0.6105	0.9241	0.4471
56	378	0.4375	0.5470	0.6155	0.9281	0.3865
57	395	0.4197	0.5134	0.6920	0.9530	0.4268
58	820	0.3667	0.4648	0.5308	0.9081	0.3966
59	67	0.4130	0.5115	0.5969	0.9269	0.3694

60	315	0.7339	0.8800	0.7441	0.9725	0.4569
61	308	0.6639	0.7942	0.7113	0.9625	0.4356
62	1100	0.7939	0.9189	0.7471	0.9696	0.4709
63	415	0.7233	0.8457	0.7195	0.9498	0.4745
64	1620	0.6435	0.8108	0.7082	0.9627	0.4394
65	1529	0.5082	0.7288	0.6101	0.9458	0.3991
66	231	0.2759	0.3437	0.6766	0.9274	0.4417
67	169	0.5143	0.6828	0.7471	0.9563	0.5041

(a) Diversity scores on EmSetInt using keywords generated on BucSetInt

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	1069	0.4861	0.6077	0.6277	0.9362	0.4121
2	11	0.5920	0.7112	0.7223	0.9679	0.4460
3	134	0.5907	0.7655	0.7178	0.9691	0.4468
4	874	0.4782	0.5793	0.5904	0.9454	0.3805
5	1035	0.4666	0.5998	0.6005	0.9357	0.4027
6	615	0.6218	0.7662	0.7008	0.9635	0.4233
7	267	0.5877	0.7265	0.5921	0.9387	0.3700
8	260	0.3872	0.4933	0.5079	0.8897	0.3112
9	757	0.5920	0.7139	0.7285	0.9673	0.4621
10	365	0.5877	0.7324	0.6095	0.9459	0.3659
11	841	0.3866	0.4728	0.5877	0.9308	0.3827
12	7	0.5731	0.6751	0.8312	0.9736	0.5759
13	237	0.5691	0.6194	0.7027	0.9536	0.4352
14	433	0.5334	0.6219	0.6307	0.9292	0.3982
15	1290	0.3764	0.4607	0.3442	0.8644	0.2039
16	1521	0.2989	0.3659	0.1929	0.8730	0.1043
17	1285	0.4232	0.6033	0.6269	0.9157	0.4222
18	781	0.4178	0.5268	0.6714	0.9530	0.4058
19	468	0.5575	0.6906	0.6992	0.9581	0.4572
20	1250	0.4920	0.6149	0.6932	0.9583	0.4333
21	256	0.6911	0.8373	0.7705	0.9769	0.5095
22	1241	0.5185	0.6562	0.6526	0.9459	0.4117
23	1037	0.6110	0.7402	0.7318	0.9681	0.4491
24	444	0.6055	0.7648	0.7475	0.9653	0.4494
25	1701	0.4825	0.6245	0.5967	0.9289	0.3805
26	56	0.6539	0.8064	0.7419	0.9668	0.4710
27	684	0.5374	0.6682	0.6233	0.9388	0.3748
28	317	0.5020	0.6308	0.5775	0.9357	0.3736
29	26	0.5424	0.6777	0.7385	0.9716	0.4603
30	84	0.5942	0.7118	0.7132	0.9627	0.4577
31	703	0.4693	0.5747	0.6701	0.9484	0.4543
32	71	0.5181	0.6207	0.7513	0.9602	0.4799
33	333	0.6458	0.7872	0.7272	0.9696	0.4584
34	1459	0.6340	0.7931	0.6725	0.9530	0.4098
35	500	0.6208	0.7829	0.6975	0.9585	0.4733
36	1307	0.4820	0.5980	0.6678	0.9528	0.4119
37	701	0.5016	0.6791	0.6778	0.9574	0.4436
38	1442	0.2672	0.3252	0.2148	0.8621	0.1571

39	208	0.6383	0.7758	0.7553	0.9731	0.4792
40	369	0.5068	0.6535	0.7130	0.9641	0.5021
41	408	0.6667	0.7863	0.6792	0.9568	0.4059
42	304	0.7394	0.8658	0.7655	0.9752	0.5001
43	221	0.5306	0.6239	0.6509	0.9527	0.4255
44	785	0.5249	0.6500	0.6918	0.9563	0.4110
45	785	0.6250	0.7672	0.6627	0.9515	0.4391
46	377	0.3229	0.4008	0.3904	0.8883	0.2677
47	59	0.2961	0.3467	0.3906	0.9004	0.2712
48	997	0.5800	0.7388	0.6680	0.9523	0.4187
49	651	0.4550	0.5643	0.5541	0.9249	0.3313
50	520	0.6051	0.7583	0.6723	0.9545	0.4300
51	738	0.4805	0.6007	0.5774	0.9210	0.3595
52	362	0.6575	0.7895	0.7339	0.9713	0.4746
53	746	0.6016	0.7301	0.6651	0.9594	0.3883
54	715	0.6628	0.8118	0.7459	0.9704	0.4923
55	1609	0.4668	0.5958	0.6010	0.9326	0.4431
56	380	0.4995	0.6382	0.6696	0.9454	0.4259
57	395	0.5155	0.6407	0.7419	0.9713	0.4608
58	820	0.4737	0.6037	0.6605	0.9553	0.4903
59	67	0.5387	0.6145	0.6753	0.9526	0.4381
60	316	0.5494	0.6909	0.7227	0.9699	0.4348
61	309	0.5410	0.6659	0.7254	0.9696	0.4511
62	1129	0.5104	0.6649	0.6470	0.9460	0.4184
63	415	0.6338	0.7449	0.6733	0.9502	0.4325
64	1666	0.5697	0.7254	0.7248	0.9663	0.4633
65	1529	0.5688	0.7489	0.7293	0.9649	0.4749
66	231	0.6447	0.7720	0.7482	0.9705	0.4853
67	170	0.5758	0.7037	0.6921	0.9565	0.4608

(b) Diversity scores on EmSetInt using keywords generated on BucSetAll

Employee#	NumVector	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
1	3045	0.5160	0.6322	0.6278	0.9279	0.4136
2	12	0.6072	0.7277	0.7121	0.9665	0.4399
3	745	0.5629	0.7301	0.7126	0.9695	0.4326
4	2258	0.4841	0.5949	0.6181	0.9488	0.3999
5	3807	0.4394	0.5629	0.5997	0.9316	0.3870
6	1418	0.6006	0.7512	0.7029	0.9617	0.4238
7	2109	0.5416	0.6710	0.4908	0.9134	0.2823
8	469	0.4240	0.5296	0.5923	0.9090	0.3697
9	1884	0.5518	0.6838	0.7058	0.9598	0.4279
10	2640	0.5446	0.6887	0.6304	0.9414	0.3614
11	861	0.3852	0.4711	0.5862	0.9303	0.3816
12	9	0.5515	0.6671	0.8333	0.9730	0.5545
13	479	0.7421	0.8026	0.7272	0.9654	0.4619
14	1174	0.4021	0.4898	0.5331	0.8992	0.3316
15	3281	0.3661	0.4465	0.3346	0.8621	0.1962
16	6214	0.2825	0.3456	0.1707	0.8685	0.0867
17	2127	0.5239	0.6729	0.6690	0.9303	0.4504

18	812	0.4145	0.5235	0.6698	0.9522	0.4051
19	1058	0.5113	0.6482	0.6766	0.9528	0.4425
20	3244	0.4629	0.5729	0.6439	0.9483	0.4062
21	1832	0.6199	0.7857	0.7352	0.9717	0.4555
22	3177	0.5031	0.6360	0.6444	0.9422	0.4006
23	2009	0.5876	0.7284	0.7277	0.9660	0.4515
24	1340	0.5777	0.7425	0.7173	0.9608	0.4245
25	3621	0.4920	0.6298	0.6502	0.9372	0.4190
26	57	0.6600	0.8127	0.7442	0.9667	0.4712
27	2485	0.5064	0.6326	0.5936	0.9292	0.3497
28	3516	0.5112	0.6546	0.5983	0.9376	0.3649
29	43	0.5519	0.7058	0.7237	0.9674	0.4648
30	724	0.5906	0.7291	0.7115	0.9596	0.4552
31	3374	0.4425	0.5346	0.6561	0.9403	0.4318
32	96	0.5060	0.5987	0.7526	0.9621	0.4834
33	842	0.5958	0.7438	0.7115	0.9691	0.4420
34	2619	0.6406	0.8045	0.6899	0.9535	0.4267
35	2408	0.6698	0.8175	0.7484	0.9654	0.5113
36	3278	0.4654	0.5855	0.6733	0.9504	0.4105
37	711	0.5060	0.6862	0.6829	0.9581	0.4482
38	2990	0.2770	0.3315	0.2073	0.8603	0.1518
39	1761	0.6070	0.7441	0.7221	0.9654	0.4489
40	750	0.5162	0.6669	0.7305	0.9664	0.5118
41	2511	0.5912	0.7232	0.6369	0.9451	0.3709
42	1331	0.7253	0.8656	0.7461	0.9726	0.4722
43	324	0.5736	0.6896	0.7189	0.9634	0.4743
44	2383	0.5049	0.6307	0.6671	0.9522	0.3833
45	2607	0.6052	0.7491	0.6454	0.9434	0.4249
46	2212	0.2862	0.3541	0.3045	0.8680	0.1921
47	191	0.2590	0.3282	0.2775	0.8717	0.1890
48	3320	0.5375	0.6893	0.6381	0.9442	0.3893
49	2773	0.4426	0.5501	0.5333	0.9150	0.3127
50	2242	0.5622	0.7114	0.6174	0.9422	0.3847
51	4217	0.4916	0.6193	0.5964	0.9228	0.3509
52	362	0.6575	0.7895	0.7339	0.9713	0.4746
53	2034	0.5922	0.7266	0.6712	0.9604	0.3974
54	1191	0.6685	0.8206	0.7588	0.9732	0.4982
55	7153	0.4110	0.5039	0.5257	0.9160	0.3533
56	2223	0.4560	0.5947	0.6521	0.9367	0.3968
57	2135	0.4733	0.5851	0.7153	0.9667	0.4304
58	5195	0.4924	0.6316	0.7029	0.9583	0.4986
59	67	0.5387	0.6145	0.6753	0.9526	0.4381
60	854	0.5242	0.6766	0.7155	0.9691	0.4274
61	937	0.5025	0.6362	0.7295	0.9699	0.4548
62	2838	0.4888	0.6479	0.6167	0.9387	0.4025
63	686	0.6476	0.7747	0.6971	0.9575	0.4452
64	3829	0.5302	0.6763	0.7024	0.9639	0.4398
65	2258	0.5932	0.7837	0.7528	0.9693	0.4915

66	1287	0.6863	0.7984	0.7293	0.9681	0.4483
67	876	0.5188	0.6629	0.6665	0.9459	0.4508

(c) Diversity scores on EmSetAll using keywords generated on BucSetAll

Figure 36: Diversity scores based only on outgoing emails (OUT)

We are also interested in the effect of the diversity of email contents on the productivity of project teams. For each set of diversity scores shown in Figure 34, Figure 35, and Figure 36, we compute team-based diversity scores by averaging the diversity scores of the employees working in the project teams based on their contributions to the projects.

In addition to the diversity scores computed by using all emails, this study provides the diversity scores based on emails divided into four-week periods. Specifically, there are 9 four-week periods:

1. 1 October 2002 – 28 October 2002
2. 29 October 2002 – 25 November 2002
3. 26 November 2002 – 23 December 2002
4. 24 December 2002 – 20 January 2003
5. 21 January 2003 – 17 February 2003
6. 1 October 2003 – 28 October 2003
7. 29 October 2003 – 25 November 2003
8. 26 November 2003 – 23 December 2003
9. 24 December 2003 – 20 January 2004

Both diversity scores for employees and team-based diversity scores are computed during the periods for future studies to evaluate the changes in diversity over time.

Appendix E: Alternative cutoff: Variance over Frequency

As mentioned in Appendix C, we initially use the variance of the word frequencies across categories or buckets over overall frequency (Var/Freq) as an alternative threshold before we decide to use the coefficient of variation as the final threshold. We define:

$$\text{Var/Freq} = \frac{1}{M} \sum_{b \in buckets} (m_b - M)^2$$

In Appendix C, we show that the variance of the word frequencies across categories does not provide an appropriated threshold for keyword selection due to the effect of the overall word frequencies on the variances. The reason that we use Var/Freq and Dintra is to balance the effect of the overall frequencies. In this Appendix, we provide the results of keyword selection and diversity scores based on using Var/Freq as a threshold for keyword selection. The results indicate that Var/Freq is also potentially useful as a threshold for keyword selection.

E.1 Wikipedia.org data set

Figure 37 shows the Var/Freq of the words in the Wikipedia.org data set. The values of Var/Freq of the frequent common words: "THE" and "AND", are expectedly low.

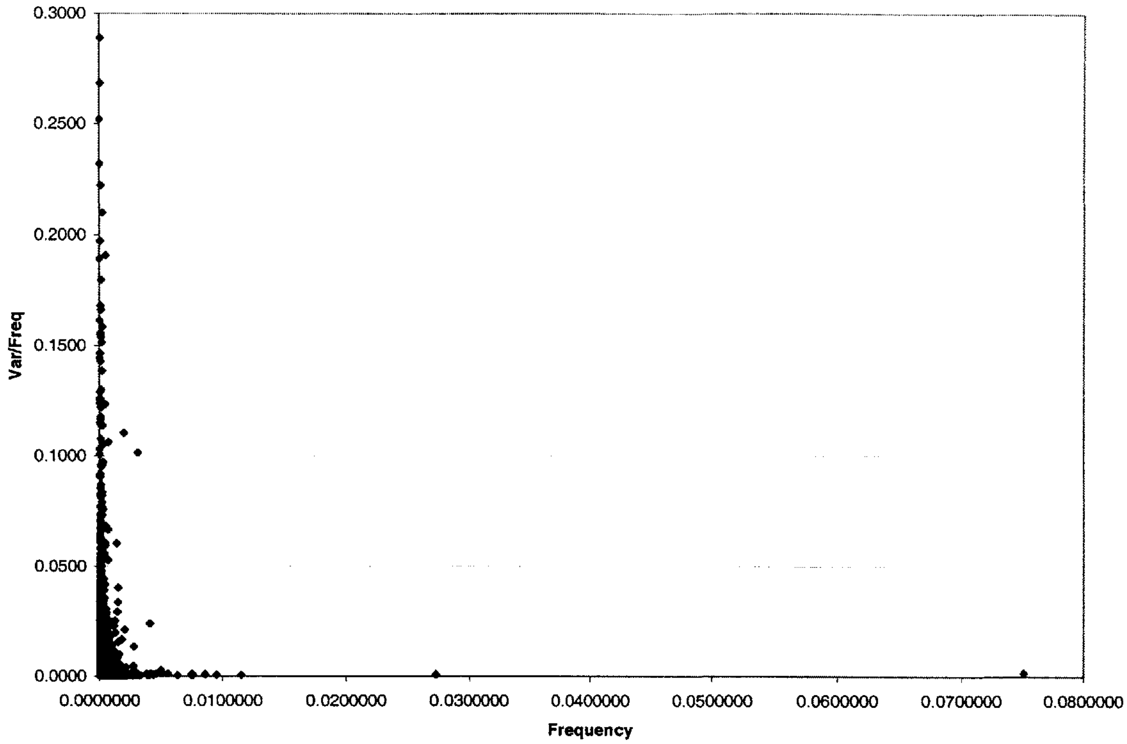


Figure 37: Var/Freq and frequencies of the words in the Wikipedia.org data set

In order to select the threshold for keyword selection, we select keywords from many thresholds and plot the Adhesion and InvCohesion of the categories based on the resulting feature vectors. The results are shown in Figure 38.

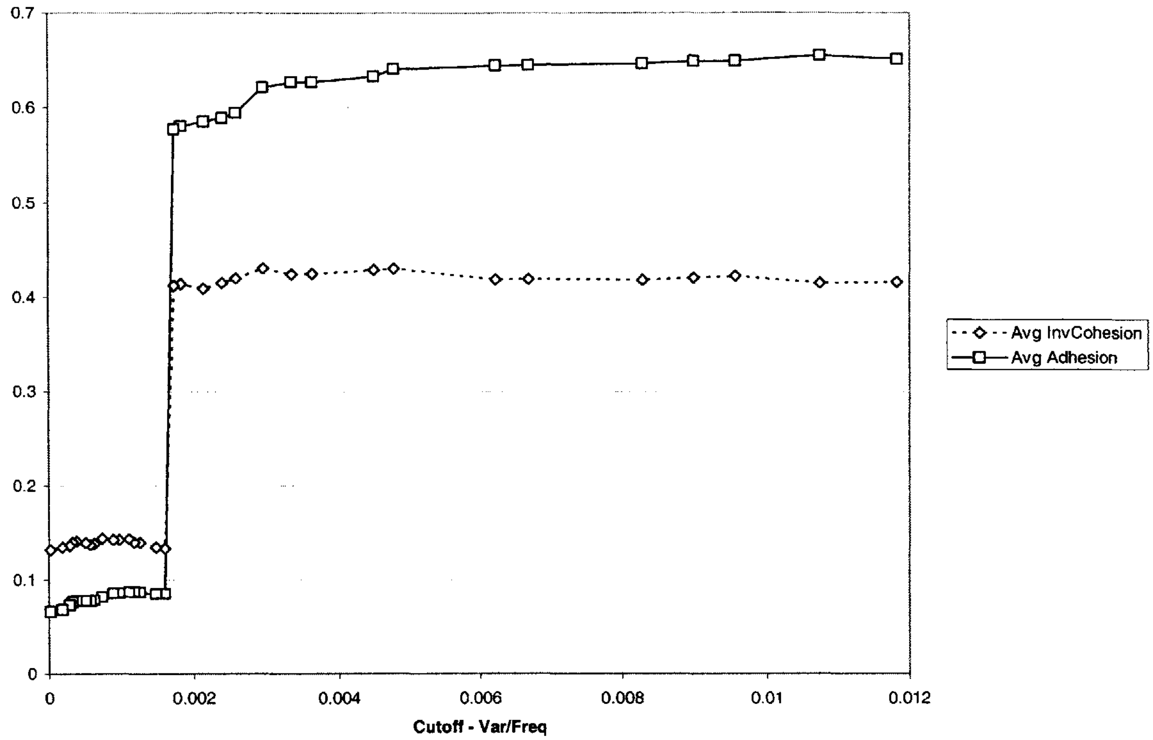


Figure 38: Adhesion and InvCohesion across multiple Var/Freq threshold in Wikipedia.org data set

We pick three threshold numbers after the sharp increase in Adhesion and InvCohesion: 1500, 5000, and 6500. The correlations shown in Figure 39 indicate that the Var/Freq threshold does not have a large effect on the diversity ranking, similar to our results from using the Dinter threshold.

Correlation - Cutoff	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
6500-5000	0.9999	0.9999	0.9927	0.9973	0.9940
6500-1500	0.9984	0.9980	0.9710	0.9813	0.9806
5000-1500	0.9989	0.9988	0.9911	0.9912	0.9951

Figure 39: Correlations of diversity scores from different set of keywords generated by three threshold numbers: 1500, 5000, and 6500.

The chart in Figure 40 shows that our diversity metrics behave similarly, and the result is confirmed by the correlations in Figure 41.

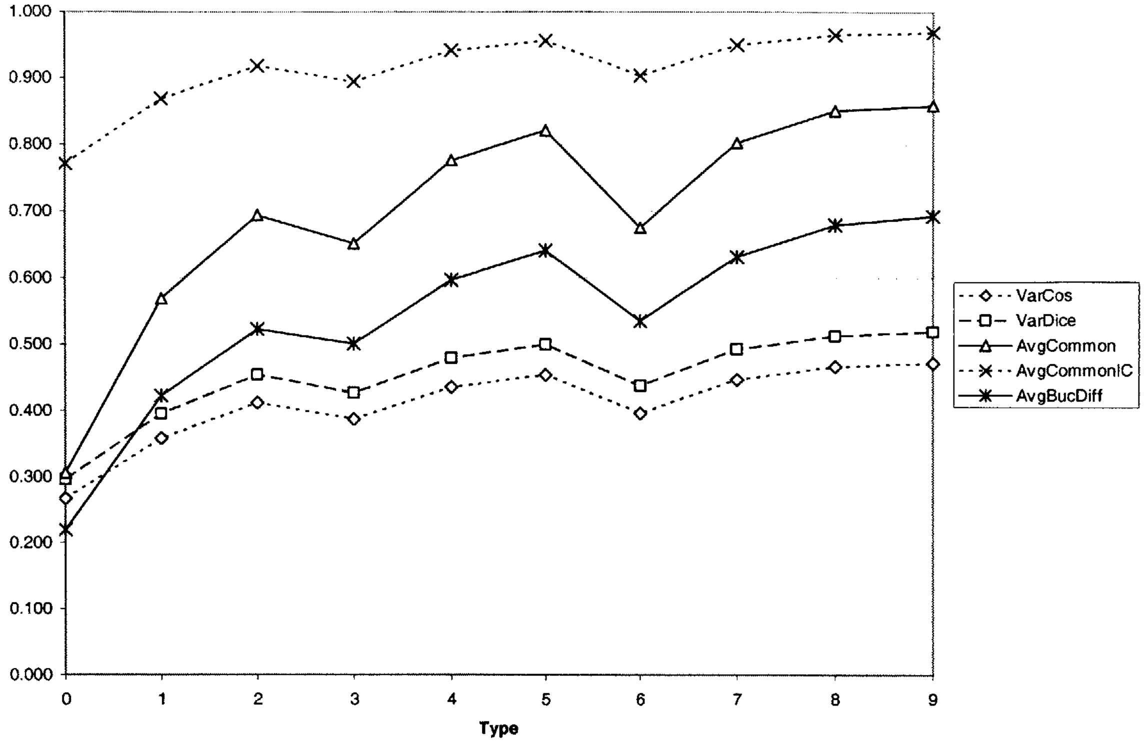


Figure 40: Averages of diversity scores grouped by configuration types

Correlations	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	1.0000	1.0000			
AvgCommon	0.9987	0.9987	1.0000		
AvgCommonIC	0.9987	0.9987	0.9991	1.0000	
AvgBucDiff	0.9960	0.9963	0.9971	0.9937	1.0000
Type	0.8314	0.8335	0.8307	0.8135	0.8696

Figure 41: Correlations of diversity scores across multiple metrics

The results of the extended triplet test in Figure 42 and Figure 43 also show high correlations between the diversity scores from document sets of different sizes. Figure 43 shows that the diversity scores increase as the size of the document set increases. The results are similar to the results from using the Dinter threshold.

Correlation - #docs	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
3docs-6docs	0.9994	0.9996	0.9995	0.9998	0.9997
3docs-9docs	0.9988	0.9992	0.9992	0.9996	0.9994
6docs-9docs	0.9998	0.9998	0.9999	0.9998	0.9999

Figure 42: Correlations of diversity scores across multiple sizes of document sets

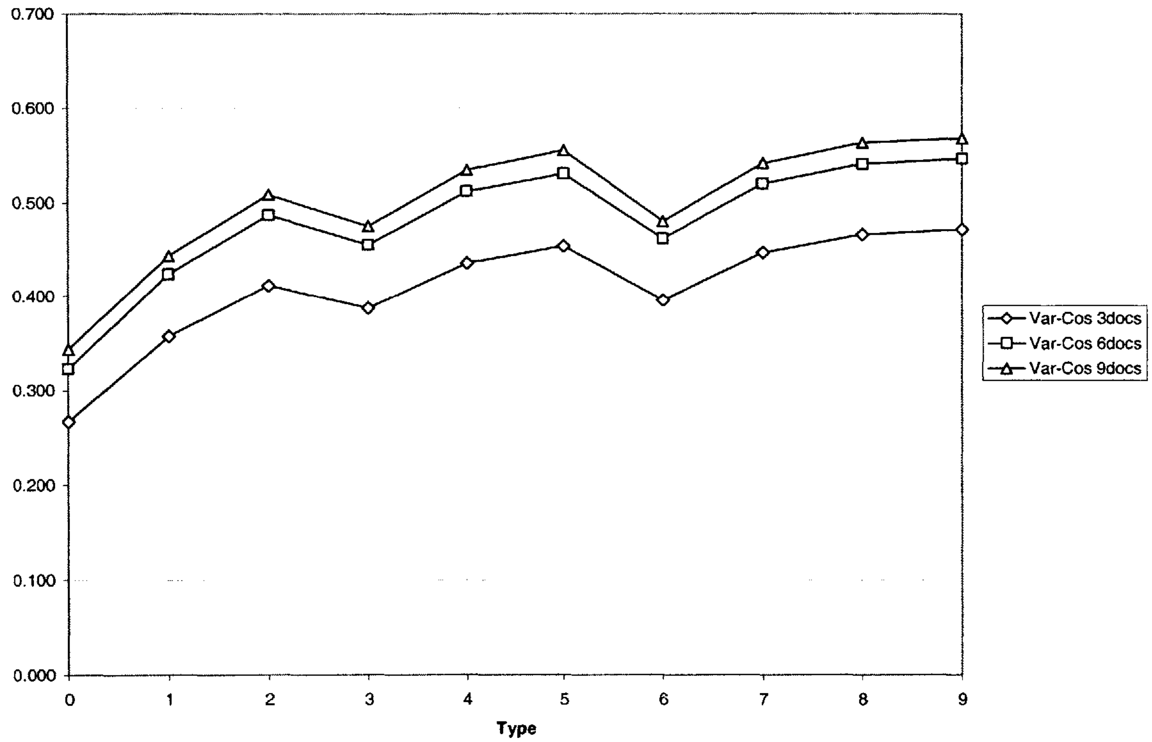
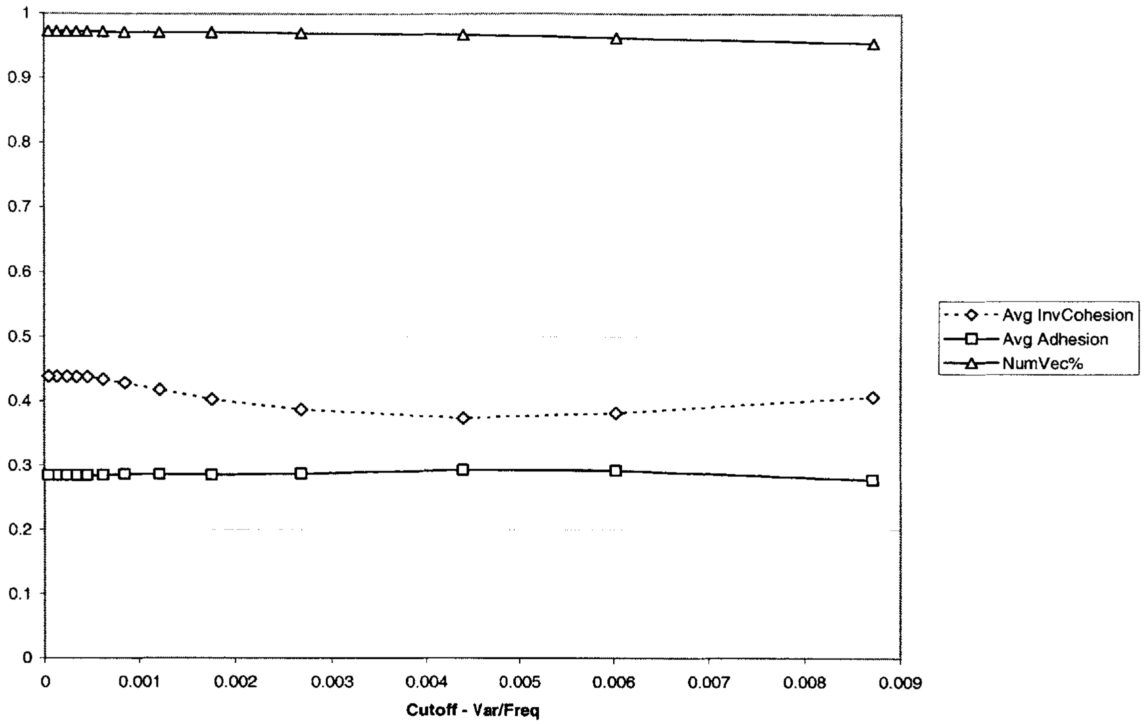


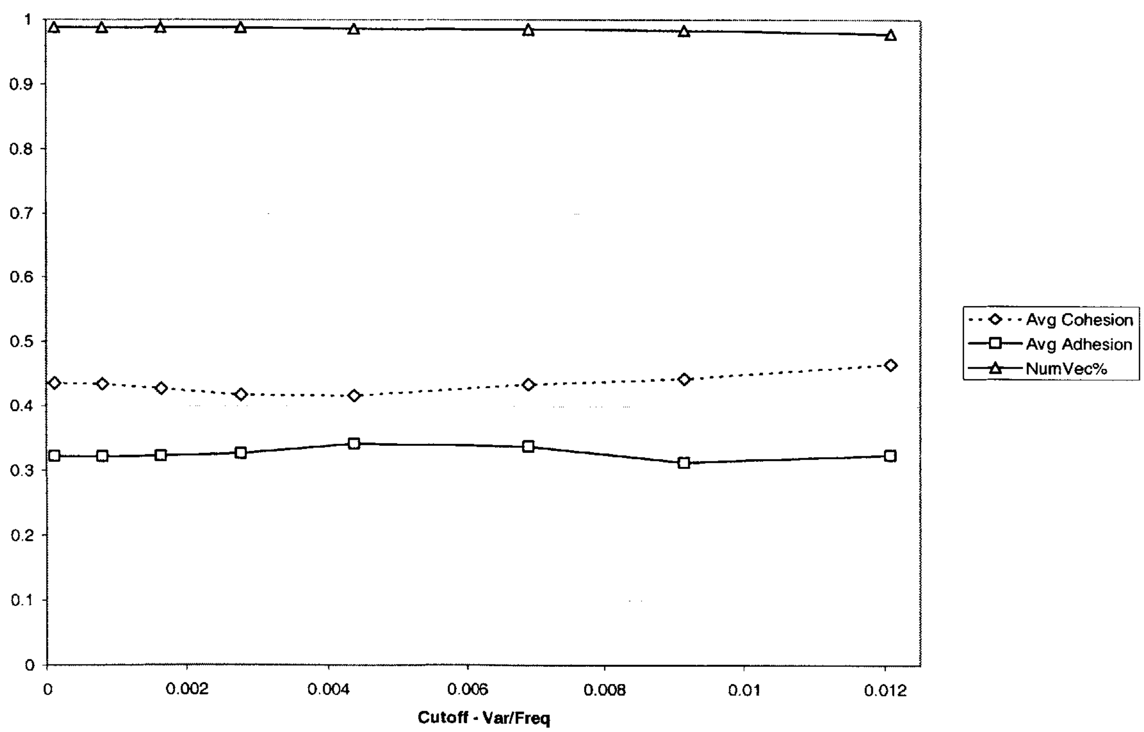
Figure 43: VarCos scores vs configuration types across multiple sizes of document sets

E.2 Email data set

We select two sets of keywords from BucSetAll and BucSetInt. In order to do so, we generate keywords using many Var/Freq threshold and plot the resulting Adhesion and InvCohesion as shown in Figure 44. We decide to use the threshold number 6000 for BucSetAll and the threshold number 8000 for BucSetInt. The correlations of the diversity scores on EmSetAll using the two sets of keywords generated from BucSetAll and BucSetInt are shown in Figure 45. The unexpectedly high correlations indicate that the inclusion of external emails does not affect the keyword selection process as much as it does when we use the Dinter threshold.



(a) Keyword selection on BucSetAll



(b) Keyword selection on BucSetInt

Figure 44: Adhesion and InvCohesion during the keyword selection on email data set

Correlation - Keyword	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
BucSetInt-All	0.8841	0.8562	0.8980	0.8441	0.8380

Figure 45: Correlations of diversity scores from different sets of keywords generated from BucSetAll and BucSetInt

Figure 46 shows the correlations across diversity metrics. The high correlations indicate that the diversity metrics behave similarly.

Correlation - MetricInt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	0.9635	1.0000			
AvgCommon	0.6760	0.6444	1.0000		
AvgCommonIC	0.7293	0.6700	0.7644	1.0000	
AvgBucDiff	0.5398	0.5304	0.8684	0.6006	1.0000

(a) Feature vectors generated from the keywords selected by BucSetInt

Correlation - MetricAll	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
VarCos	1.0000				
VarDice	0.9779	1.0000			
AvgCommon	0.7365	0.7570	1.0000		
AvgCommonIC	0.7448	0.7532	0.9424	1.0000	
AvgBucDiff	0.7496	0.7679	0.9301	0.8724	1.0000

(b) Feature vectors generated from the keywords selected by BucSetAll

Figure 46: Correlations of diversity scores across multiple diversity metrics

Figure 47 shows the correlations of diversity scores on EmSetInt and EmSetAll using the same set of keywords generated from BucSetAll. Again, the moderate correlations indicate that the inclusion of external emails affect diversity ranking.

Correlation - IntExt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
EmSetInt-All	0.5241	0.5615	0.4728	0.5945	0.3047

Figure 47: Correlations of diversity scores on EmSetInt and EmSetAll

Figure 48 shows the effect of incoming and outgoing emails. Similar to the previous results, incoming and outgoing emails exhibit differences in contents. The differences are clear with the inclusion of external emails as there is no correlation between the diversity scores based on INC and the scores based on OUT.

Correlation - InOutInt	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
IO-INC	0.7733	0.8027	0.7017	0.7421	0.6559
IO-OUT	0.8463	0.8834	0.9257	0.9251	0.8982
INC-OUT	0.3815	0.4876	0.4251	0.4787	0.3054

(a) Correlations of diversity scores on EmSetInt

Correlation - InOutAll	VarCos	VarDice	AvgCommon	AvgCommonIC	AvgBucDiff
IO-INC	0.7731	0.7692	0.6134	0.6394	0.7427
IO-OUT	0.3464	0.4353	0.4943	0.6318	0.2537
INC-OUT	-0.0215	0.0365	0.0607	0.1592	-0.0319

(b) Correlations of diversity scores on EmSetAll

Figure 48: Correlations of diversity scores computed from incoming, outgoing, and both incoming and outgoing emails.

In summary, the results from using the Var/Freq threshold are similar to the results from using Dinter threshold. Var/Freq can be an alternative measurement for the property of a keyword that distinguishes the contents of categories.

References

Allison, P. D. (1978). Measures of Inequality. American Sociological Review, 43, 865-880.

Ancona, D. G. & Caldwell, D. F. (1992). Demography and Design: Predictors of New Product Team Performance. Organization Science, 3(3), 321-341.

Aral, S., Brynjolfsson, E., & Van Alstyne, M. (2006). Information, Technology and Information Worker Productivity: Task Level Evidence. Proceedings of the 27th Annual International Conference on Information Systems, Milwaukee, Wisconson.

Berry, M. W., Dumais, S. T., & O'Brien, G. W. (1995). Using linear algebra for intelligent information retrieval. SIAM Review, 37(4), 573-595.

Duda, R.O., & Hart, P.E. (1973). Pattern Classification and Scene Analysis. New York: Wiley.

Li, M. & Vitányi, P. (1997). An Introduction to Kolmogorov Complexity and Its Applications. New York: Springer-Verlag.

Lin, D. (1998). An Information-Theoretic Definition of Similarity. In 15th International Conference on Machine Learning (pp. 296-304). San Francisco, CA: Morgan Kaufmann.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development, 2(2), 159-165.

Pfeffer, J. & O'Reilly, C. (1987). Hospital Demography and Turnover Among Nurses. Industrial Relations, 36, 158-173.

Rasmussen, E. (1992). Clustering algorithms. In Frakes, W. B., & Baeza-Yates, R. (Eds), Information Retrieval: Data Structures and Algorithms (pp. 419-442). Englewood Cliffs, NJ: Prentice Hall.

Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (pp. 448-453). Montreal, Canada.

Salton, G. (1988). Automatic Text Processing. Massachusetts: Addison-Wesley.

Salton, G. & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing and Management, 24(5), 513-523.

Salton, G., Wong, A., & Yang, C. S. (1975). A Vector Space Model for Automatic Indexing. Communications of the ACM, 18(11), 613-620.

Van Alstyne, M. & Brynjolfsson, E. (2005). Global Village or Cyberbalkans: Modeling and Measuring the Integration of Electronic Communities. Management Science, 51(6), 851-868.

Zipf, G. K. (1949). Human Behavior and the Principle of Least Effort. Cambridge, MA: Addison-Wesley.