World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:5, 2008

Advanced Information Extraction with n-gram based LSI

Ahmet Güven, Ö. Özgür Bozkurt, and Oya Kalıpsız

***Abstract*—**Number of documents being created increases at an increasing pace while most of them being in already known topics and little of them introducing new concepts. This fact has started a new era in information retrieval discipline where the requirements have their own specialties. That is digging into topics and concepts and finding out subtopics or relations between topics. Up to now IR researches were interested in retrieving documents about a general topic or clustering documents under generic subjects. However these conventional approaches can't go deep into content of documents which makes it difficult for people to reach to right documents they were searching. So we need new ways of mining document sets where the critic point is to know much about the contents of the documents. As a solution we are proposing to enhance LSI, one of the proven IR techniques by supporting its vector space with n-gram forms of words. Positive results we have obtained are shown in two different application area of IR domain; querying a document database, clustering documents in the document database.

***Keywords*—**Document clustering, Information Extraction, Information Retrieval, LSI,n-gram.

I. Introduction

CONVENTIONAL information retrieval techniques work well in finding relevant documents for searched keywords in large document collections. For example, Google being the best search engine will bring the right documents for a main topic like "customer relationship management". However, as the number of documents being searched increases, the number of documents being retrieved for a topic also increases and most of the time there are thousands of documents returned. Nobody is interested in so many documents. What is important when there are lots of documents is to dig into the general topic that is being mined. The above query is expected to take a more specific form like "the effect of customer relationship management on company effectiveness" since lots of documents on hand about customer relationship management won't make sense.

In order to be able to explore the subtopics of a main topic or relations between topics and subtopics, a more comprehensive method of representing the contents of documents is needed. IR systems like Google represent the documents with the words in them, with the relationships among those words and with the statistical cooccurence patterns of these words. However sole usage of words has limits.

Manuscript received September 15, 2006.

Authors are with Yıldız Technical University, Department of Computer Science, Istanbul, Turkey (e-mails: guven@ges.net.tr, ozgurb@yildiz.edu.tr, kalipsiz@yildiz.edu.tr).

- Authors use different words in writing almost similar documents. Besides the words used by people to search documents also differ although they all intent to explain the same subject. This constraint is well known in IR field as polysemy and synonymy nature of words. [5]
- Word couples have different meanings than their separate usage. The sentences "Wolf killed man" and "man killed wolf" although consist of same words, have very different meanings.
- Position of words within the sentences has different power in forming the content of documents. For example, words that are close to object have more importance in determining the meaning.

We have tested our approach in two research areas of IR domain; document clustering and querying.

The organization of this paper is as follows. Section 2 describes the enhancement to original LSI. Section 3 explains the methods and measures used in the study. Section 4 gives basic information about data sets used. Section 5 and Section 6 have the detailed results for clustering and querying studies. Finally, Section 7 concludes the paper.

II. Combining LSI and n-gram

n-gram technique was first used to automatically determine the language of the documents [4]. This technique uses the statistical patterns of letters in the words of a specific language. For example the word 'computer' has following *n*-grams:

2-gram : c co om mp pu ut te er r
3-gram : c co com omp mpu put ute ter r

Every language has different *n*-gram patterns. By the help of this statistical knowledge, the language of a document can easily be determined [8].

To enhance LSI, we extend the word space of LSI by including *n*-gram form of words. This brings two advantages over original LSI. First one is to take the word couples into account. For instance, the English phrase to "kick the bucket" means to die. A listener knowing only the meaning of kick and bucket would not be able to deduce what the expression actually means, since phrase 'kick the bucket' can literally refer to the act of giving a kick to a bucket. Second one is about the power of words in the meaning of a sentence with respect to their position. Usually words closer to the object have more weight in the meaning of the sentence. For example the sentence "Today John has broken the glass" focuses on

"glass". Likewise the sentence "John has broken the glass today" focuses on time "today". If we use the same words to form the sentence "Today the glass was broken by John", meaning now focuses on subject, "John".

Instead of using just unigram words to form the term space, we use both 2-gram and 3-gram words with unigrams. The right meaning of a document collection can only be understood by considering single words, 2-gram words and 3-gram words all together since they all have some power on the meaning of the documents. Combined all together they can give the right meaning.

III. Implementation

Our method employs several preprocessing steps such as stemming and removal of stop words on the document set. Each document is represented by a vector of frequencies of stemmed words and *n*-grams formed by word couples and triplets. As an extra preprocessing step, the actual term frequency of an item is replaced by the weighted frequency, i.e., *term frequency - inverse document frequency* (TF-IDF), in the document vector. Thereafter SVD is performed on the term-document matrix. On the factorized term-document matrix, we have done clustering and querying operations.

When the clusters are on hand, f-measure and entropy values are considered for determining the accuracy of the clustering.

A. Preprocessing

LSI requires the determination of the words which are important for the content of documents. For this purpose first step is preparing the term – document matrix. Each column of the matrix is a document vector consisting of the frequencies of each term within the document. Terms are stems of the words and *n*-grams formed by consecutive words of the documents, excluding the words which do not carry importance in the meaning, known as stop words. Stemming process is language dependent. The common stemmer for English, named Porter Stemmer [15], is a small algorithm; but, stemmers for most of the languages, like Turkish, can not be formed algorithmically. The complicated structure of Turkish language requires using a dictionary for this process. To stem Turkish words, dictionary based natural language processing tool called Zemberek [13] is used.

The resulting matrix is a term document matrix where each cell holds the frequency of each term within each document. Further processing requires replacing the frequency by tf-idf value, in order to evaluate the value of each term in the document set instead of any individual document. The idea is that if an item is too common across different documents, then it would have little discriminating power, and vice versa [10]. Tf-idf is a term weighting formula and is as follows:

$$w_{ij} = tf_{ij} * \log_2 \frac{N}{n}$$

w_{ij} : weight of word Tj in Document Di

tf_{ij} : frequency of word Tj in Document Di

N : number of documents in the collection

n : number of documents where word Tj occurs at least once

Next step is the key step that separates LSI from convetional "vector space model". It is called decomposition and the matrix is factorized by using Singular Value Decomposition (SVD) [2, 6]. After this process, words which are in the same context will still be closer to each other in another vector space but has less dimensions. Words with closer meaning will co-occur in the vector space and as a result, similar words will make similar documents fall into same clusters [2].

SVD operation decomposes a rectangular matrix, A, into product of three matrices, U, S and V. U is an orthonormal matrix and corresponds to rows of A. New matrix has fewer columns which are linearly independent of the others. S is a diagonal matrix with singular values only along one diagonal. These singular values scale the factors in the rows of the other matrices, such that when all three matrices are multiplied, the original matrix is formed. V is an orthonormal matrix and has columns corresponding to the original columns while rows are composed of derived singular vectors.

Following the decomposition by SVD, the *n* most important dimensions (those with the highest singular values in S) are selected. The amount of dimensionality reduction is critical and an open issue. Ideally, *n* should be large enough to fit the real structure in the data, but small enough such that noise, sampling errors or unimportant details are not modeled [6]. The reduced dimensionality solution then generates a vector of *n* real values to represent each document. The reduced matrix ideally represents the important and reliable patterns underlying the data in A.

B. Clustering Evaluation Techniques

After factorizing the matrix A, next step for clustering purposes comes and here we have to find the similarity of documents to each other. By definition, the document by document comparison matrix is obtained by multiplying the transpose of A by itself that is $DD = A^T . A$. When A is replaced by the reduced matrices obtained from SVD operation, the results is $V . S^2 . V^T$. The elements in the cell i, j of this matrix gives the similarity between document i and j. [16]

Entropy and F-measure are the two methods which are widely used for evaluating the accuracy of clustering algorithms. To use both techniques, the documents to be clustered need to be labeled before to be able to compare the resulting clusters and the labeled classes [17].

Entropy is a concept to show how homogenous a cluster is. Lower entropy levels are preferred since it shows that a resulting cluster is more homogeneous. We can calculate both the entropy of an individual cluster and the whole clustering process. Entropy for a cluster can be measured as below:

$$E_j = - \sum P(i,j) * \log P(i,j)$$

Here P(i,j) is the probability of documents labeled as *i* to be in cluster *j*. Total entropy for the whole clustering process is calculated by summing the entropies of all clusters. In order to balance the impact of each cluster on total entropy, normalization for each cluster is done by multiplying it with its weight with respect to its size

$$E = \sum n_j / n * E_j$$

Where n_j stands for the size of the cluster j and n stands for the total number of documents in the document set.

F-measure is based on mostly used two metrics in IR field: recall and precision. Precision shows how many of the documents in a cluster are in the right cluster with respect to the cluster size. Recall shows how many of the total documents to be put in a specific cluster is really in that cluster. To calculate f-measure first of all the f-measure of clusters and the classes have to be calculated as follows:

$$F(i,j) = \frac{2 * Recall(i,j) * Precision(i,j)}{Recall(i,j) + Precision(i,j)}$$

The total F-measure for the clustering process is then

$$F = \sum n_i / n * \max F(i,j)$$

Where n_j stands for the number of documents in class *i* and *n* is the total number of documents is the document set. Higher values of f-measure indicate higher quality of clustering.

IV. Characteristics of the Document Sets

Our study basically focuses on Turkish documents. Since Turkish is not a much studied language in the field of information retrieval, to prove the results we have obtained in Turkish, we tested our approach on English documents as well. Results obtained on both document sets are provided in the next section.

Turkish Document Set: There is not yet a fully studied and overall accepted Turkish document set. Therefore, an experimental document set of 615 documents which are collected from managerial magazines belonging to past 5 years are collected. The size of the documents ranges from 4 to 100 and can be classified in 13 sub categories.

Reuters 21578 English Document Set [14]: This is the collection of documents that appeared on Reuters newswire in 1987. The documents were assembled and indexed under categories. Each document can belong to multiple categories and multiple classes within the categories. There are six categories; people, places, orgs, topics, companies and exchanges.

V. Experimental Results

As mentioned before, our approach to produce a better clustering algorithm than the current ones by improving LSI technique by enhancing it with *n*-gram words. This technique is still an unsupervised method and as the number of documents being created increases it is vital to have a successful unsupervised method in information retrieval field of study.

A. Clustering

To make our approach comparable, we have clustered both the Turkish document set and English one with both normal LSI and n-gram based LSI. Clustering also shows the the precision and recall comparison of both techniques since f-measure criteria is composed of recall and precision calculations. The algorithm details are listed in Table I with their corresponding acronyms used in graphs.

TABLE I
Algorithms used for Comparison

Algorithm	Explanation
Tfidf_LSI_HAC	LSI based HAC (Tfidf based vector space)
Tfidf_ngram_LSI_ HAC	Ngram based LSI based HAC (Tfidf based vector space)

TABLE II
Entropy Measurements for Clustering of Turkish Documents

# of clusters	Tfidf_LSI_HAC	Tfidf_ngram_LSI_HAC
5	0,926381306	0,90608821
10	0,834725048	0,73937549
15	0,795165679	0,65872017
20	0,700502409	0,59382918
25	0,660731649	0,54026323
30	0,594019001	0,48636027
35	0,571697287	0,46587803
40	0,534692964	0,42687611

F-measure and entropy calculations for 615 Turkish documents which were preclassified into 13 classes are given in Table II and Table III. Their graphics are shown in Fig. 1 and Fig. 2.

TABLE III
F-Measure Calculations for Clustering of Turkish Documents

# of clusters	Tfidf_LSI_HAC	Tfidf_ngram_LSI_HAC
5	0,246469381	0,26671552
10	0,261534557	0,38382652
15	0,269808758	0,41365287
20	0,316816249	0,39988699
25	0,325498299	0,4177481
30	0,331417705	0,43376812
35	0,331566884	0,39960739
40	0,349225421	0,42172488

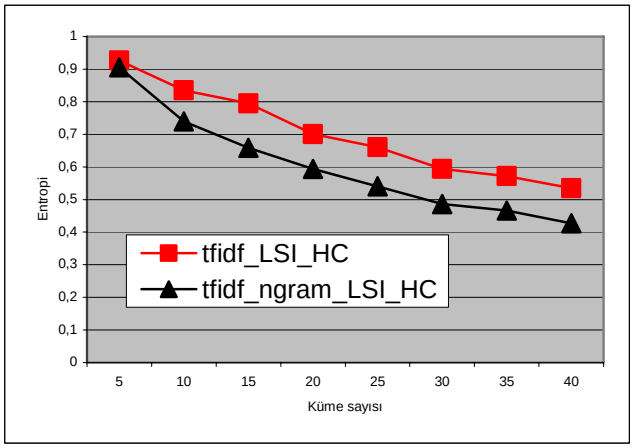

Fig. 1 Change of entropy relative to the number of clusters for Turkish document set

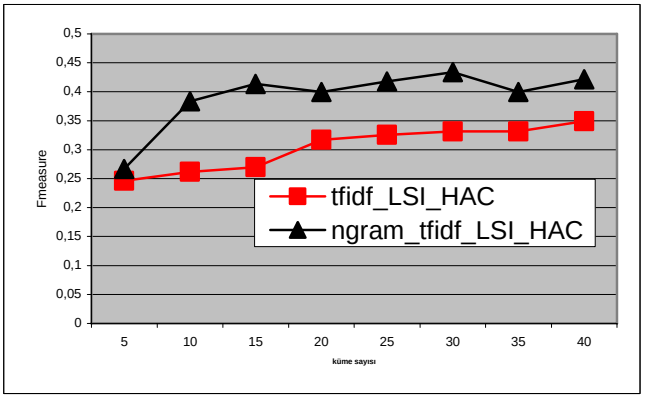

Fig. 2 Change of f-measure relative to the number of clusters for Turkish document set

It is obvious from the graphics that n-gram based LSI does the clustering much better than normal LSI with respect to f-measure and entropy measure. The clusters formed by n-gram based LSI are more homogeneous (lower entropy) and have much better quality (higher f-measure). Turkish document set was preclassified into 13 classes. N-gram based LSI was more successful than normal LSI in clustering documents into 13 clusters. Moreover as the number of clusters increases stil n-gram based LSI is better in clustering. This means clusters, though they are homogenous have inner clusters inside and when there is a need to further cluster a cluster, n-gram based LSI does it much better.

To test our approach on the English document set, we have randomly selected 1680 documents from 20.000 documents in Reuters21578 corpora which were preclassfied into 20 classes. Entropy and f-measure calculations for different number of clusters are shown in Table IV and Table V. Graphics for these tables are given in Fig. 3 and Fig. 4.

TABLE IV
ENTROPY MEASUREMENTS FOR CLUSTERING OF ENGLISH DOCUMENTS

# of clusters	Tfidf_LSI_HAC	Tfidf_ngram_LSI_HAC
5	0,926381306	0,90608821
10	0,834725048	0,73937549
15	0,795165679	0,65872017
20	0,700502409	0,59382918
25	0,660731649	0,54026323
30	0,594019001	0,48636027
35	0,571697287	0,46587803
40	0,534692964	0,42687611

TABLE V
F-MEASURE CALCULATIONS FOR CLUSTERING OF ENGLISH DOCUMENTS

# of clusters	Tfidf_LSI_HAC	Tfidf_ngram_LSI_HAC
5	0,246469381	0,26671552
10	0,261534557	0,38382652
15	0,269808758	0,41365287
20	0,316816249	0,39988699
25	0,325498299	0,4177481
30	0,331417705	0,43376812
35	0,331566884	0,39960739
40	0,349225421	0,42172488

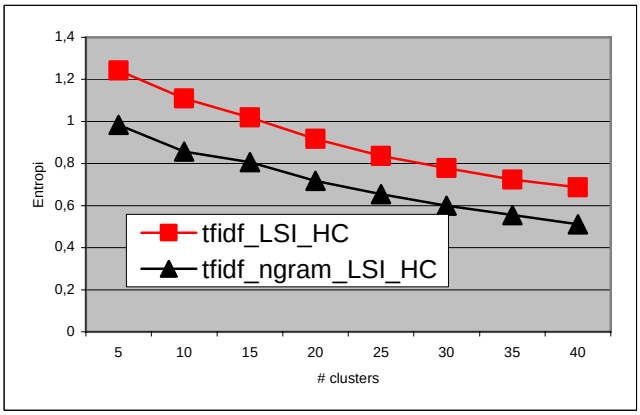

Fig. 3 Change of entropy relative to the number of clusters for English document set

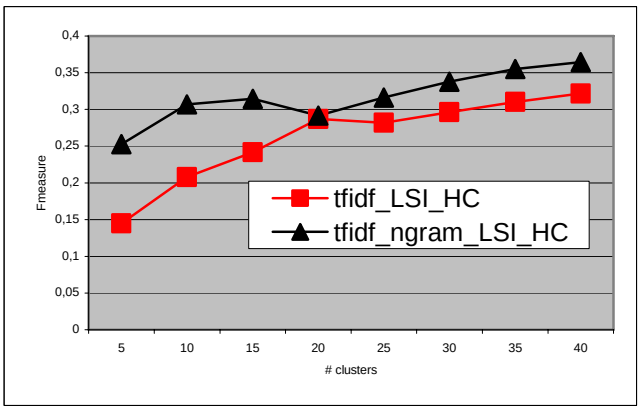

Fig. 4 Change of f-measure relative to the number of clusters for English document set

Likewise as in Turkish corpora, our proposed approach does better clustering for English document set. Surprisingly for 20 clusters normal LSI has reached n-gram based LSI but still at that point n-gram based LSI has formed more homogeneous clusters. For further number clusters, n-gram based LSI never loses.

B. Query Processing

Our tests on searches in our Turkish and English document sets had proved the success of n-gram based LSI over normal LSI. Some sample query results are given in Table VI, VII, and VIII. Relevance in the second column of the tables is the cosine value of the angle between the query vector and the document vector.

Since we believe a deeper searching is needed within topics, the classical precision and recall measures used to evaluate conventional querying mechanisms are not valid in our case. Because when dealing with subtopics and relation within topics, it is not easy beforehand to know which document belongs to which subtopic and for almost every corpora, preclassification process does not deal with this issue. For instance in Reuters 21578 collection, a topic name like "earn" has many different documents which have only the earning subject in common but all differ in content.

TABLE VI
QUERY RESULTS FOR TURKISH SEARCH TERM "BILGI YÖNETIMININ PAZARLANMASI" MEANING "MARKETING OF INFORMATION MANAGEMENT"

bilgi yönetiminin pazarlanması	
Normal LSI	Relevancy
Pazarlama Stratejileri	0,573
Crm - 12 Soruda Bire-Bir Pazarlama	0,565
Pazarlama - Başarinin Sirri Farklilaşmada	0,481
Pazarlama - Gerilla Usulu Pazarlamanin Onbeş	0,419
Pazarlama Performansini Ölçüyor Musunuz	0,388
Pazarlama Şirketlere Hocadan Not	0,363
Pazarlama Kotlerden Yarina Dersler	0,326
Pazarlama Yarim Einstein Yarim Picasso	0,322
Pazarlamada Yeni Trendler	0,297
LSI with *n*-gram terms	Relevancy
Bilgi Yönetimi Nedir	0,557
Bilgi Yönetimi	0,471
Bilgi Yönetimi Ve Uygulamaları	0,421
Bilgiyi Iyi Yöneten Karini Artiriyor	0,399
Bilgi Yön-Mitoslar Ve Gerçekler	0,39
Bilgi Yön-Etkin Bilgi Yönetimi	0,385
Pazarlama Stratejileri	0,322
Bilgi Yön-Şirketiniz Bilgi Yönetimine Hazır Mı	0,328
Crm - 12 Soruda Bire-Bir Pazarlama	0,327

For the Turkish query test, the sample query shown in Table VI is given. This query is searching documents which are related with "marketing of information management". Surprisingly all the documents returned from normal LSI are all related with marketing. However results returned from n-gram based LSI hits the target by bringing documents about information management and its marketing.

TABLE VII
QUERY RESULTS FOR SEARCH TERM "OIL COMPANY ACQUISITION"

oil company acquisition	
Normal LSI	Relevancy
acq,-19826	0,365
acq,crude,-4315	0,363
grain,oilseed,veg-oil,sorghum,sun-meal-15500	0,293
acq,crude,-9913	0,27
acq,-19837	0,253
acq,crude,nat-gas,-16007	0,252
gold,-16589	0,246
acq,crude,nat-gas,-8100	0,244
heat,-5706	0,237
LSI with *n*-gram terms	Relevancy
gas,fuel,-17441	0,479
gas,-19083	0,367
acq,crude,-4315	0,303
heat,-19223	0,292
gas,-17173	0,29
heat,gas,-11880	0,269
acq,crude,-5116	0,229
acq,-19984	0,216
gas,-18367	0,211

TABLE VIII
QUERY RESULTS FOR SEARCH TERM "OIL INDUSTRY CONSUMER PRICE"

oil industry consumer price	
Normal LSI	Relevancy
acq,crude,-9913	0,347
grain,oilseed,veg-oil,sorghum 15500	0,312
acq,crude,-9947	0,269
grain,rice,-19059	0,258
cpi,-10260	0,256
acq,sugar,crude,-11213	0,236
heat,-5706	0,232
heat,-12670	0,23
heat,gas,-11880	0,227
LSI with n-gram terms	Relevancy
cpi,-10260	0,481
acq,crude,-9913	0,333
acq,crude,-9947	0,253
cpi,-6982	0,242
cpi,-6920	0,231
cpi,-10553	0,224
cpi,-20247	0,221
cpi,-8726	0,217
cpi,-16950	0,211

Table VII and VIII are showing two sample queries done in English documents set. As can be observed easily, n-gram based LSI approach bring almost very different documents than normal LSI and the documents returned from n-gram based LSI are the true documents. This is because n-gram based LSI is more powerful in extracting the right meaning from documents.

VI. Conclusions and Future Work

It is shown that, a systematic approach to consider the word couples as well as the words themselves provides more successful clustering of documents. We obtained these successful results by applying *n*-gram technique to words of the documents and using these terms in the word list which determine the content of documents.

One point to mention arises when *n*-gram method is used to form word couples. Since it is a systematic approach, meaningless word couples are also formed. However, the way LSI works in calculating the statistical patterns of word co-occurrences handles this problem. Since the meaningful word couples expected to be used frequently throughout the document set, the dummy word couples will be separated from the meaningful ones automatically.

There is still way to go in enhancing the *LSI with n-gram words* technique. Following are two points to consider in designing a search engines based on *LSI with n-gram words* technique:

- Objects and words closer to objects should have higher term values so a way to calculate it should be found.
- While determining the value of words and word couples to the documents, their linguistic nature has to be considered like being nouns, adjectives etc.

References

[1] Bellot, P. and El-Beze, M., *A Clustering Method for Information Retrieval,* Technical Report IR-0199, Laboratoire d'Informatique d'Avignon,France, 1999.
[2] Berry, M. W., Drmac, Z. and Jessup E. R.: Matrices, Vector Spaces, and Information Retrieval, SIAM Review, v.41 n.2, p.335-362, June 1999.
[3] Boley D., Principal direction divisive partitioning. Data Mining and Knowledge Discovery, 2(4), 1998.
[4] Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai "Class-based n-gram models of Natural Language", *Computational Linguistics*, vol. 18, pp. 467–479, 1992.
[5] Croft, W.B. and Xu, J.: Corpus-specific stemming using word form co-occurence. In Proceedings for the Fourth Annual Symposium on Document Analysis and Information Retrieval (pp. 147-159), Las Vegas, Nevada. 1995.
[6] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R.: (1990). Indexing by latent semantic analysis. Journal of the American Society for Information Science, 41(6), 391-407.
[7] Duda, R. O., Hart, P. E., and Stork, D. G., Pattern Classification. Wiley, New York.2001.
[8] Ekmekcioglu, F. C., Lynch, M. F. and Willett, P. (1996): Stemming and N-gram Matching for Term Conflation in Turkish Texts. Inf. Research, Vol. 2, No. 2.
[9] Kohonen, T., "The Self-Organizing Map," Proceedings of the IEEE, vol. 9, 1990, pp. 1464-1479.
[10] Lingpipe NLP Library http://www.aliasi.com/lingpipe
[11] Salton, G. and McGill, M. J.: Int. to modern information retrieval. McGraw-Hill.
[12] Willet, P., Recent trends in hierarchical document clustering: a critical review. Information Processing and Management, vol. 24(5), pages 577-- 597, 1988.
[13] Zemberek Turkish NLP Library: https://zemberek.dev.java.net/
[14] "Reuters21578collection", http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html
[15] Porterstemmer , http://www.tartarus.org/martin/PorterStemmer/
[16] Foundations of Statistical Natural Language Processing (Hardcover) by Christopher D. Manning, Hinrich Schütze.
[17] Unsupervised Machine Learning Techniques for Text Document Clustering, Arzucan Özgür, Ethem Alpaydın.