Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 01-042

Automated Clustering and Extraction of Distinctive Words in Legal Documents

Neal Vaughn and Daniel Boley

December 03, 2001

Automated Clustering and Extraction of Distinctive Words in Legal Documents*

Neal Vaughn and Daniel Boley

December 3, 2001

Abstract

An agent is described to automatically organize and annotate a collection of text documents data set. This agent clusters the data collection and generates a set of distinctive word labels for each cluster of documents, all entirely autonomously. The agent is based on the use of Principal Direction Divisive Partitioning and k-means, applied to both the documents and the words, using a bag of words model. The agent is capable of extracting the words that are most useful in distinguishing among the documents. All this processing by the agent occurs without input from a human user, except to specify the original document set. The agent is illustrated with a collection of alcohol laws enacted in the 50 states of the U.S.

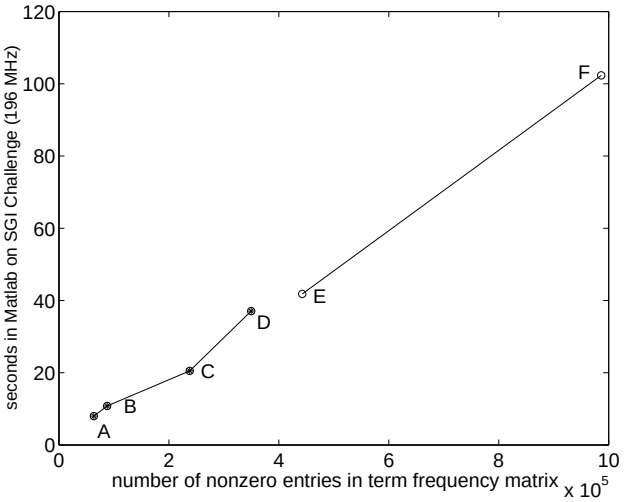
Keywords: PDDP, k-means, unsupervised clustering

1 Introduction

With the explosion of data available in electronic form, some automated assistance is needed to effectively explore large data collections. In many cases, users may not know exactly what they are looking for within the data collections, but if the individual documents could be grouped by some weakly relevant criteria, a user is quickly able to eliminate many documents as being much less relevant, and then focus in on the most promising documents. Unlike the usual information retrieval scenario, in which the user selects documents by providing a collection of keywords, often the user needs to browse the documents looking for patterns that may transcend keywords. This could occur if, for example, different documents have used different keywords for the same topics, or if the user is not intimately familiar with the variety of vocabulary used in the document set. In many cases, document collections may have been organized to serve one purpose, but the user needs a separate organization. In the case of the legal collection examined in this paper, the documents are organized by state, and within the state by the legislative chamber that originated the law, whereas the users need to organize the collection by the topics covered by the laws. The users might not want to prejudice their own reorganization of the document collection by predicting the relevant keywords in advance, partly to avoid imposing their own bias on the document collection and partly to avoid inadvertently losing some documents.

For these reasons, an unsupervised clustering algorithm is essential as an initial first step in imposing an organization on a new document collection. Beyond computing the clusters, it is useful to automatically generate some significant annotations to identify each cluster as a group. Carrying out these tasks with no human intervention allows one to incorporate these methods into an agent that could autonomously maintain the data collection, retrieve new documents, update the locally computed clusters, and maintain an annotated table of contents into the document collection. This paper discusses a scalable unsupervised algorithm that is capable of automatically generating annotated clusters automatically, while identifying the attributes or words that are the most important in distinguishing between the variety of documents in a given collection.

*This research supported in part by NSF grant IDM-9811229



key	A	B	C	D	E	F
docs	2340	2340	2340	2340	10794	21577
words	7358	8104	1458	21839	21190	32513

Figure 1: Relative times to compute PDDP clusters on data sets of various sizes.

The Principal Direction Divisive Partitioning Algorithm is an unsupervised, top-down clustering algorithm that has been shown to be useful for exploring large datasets such as web-based text documents [3, 2]. This algorithm was chosen because of its scalability to large datasets [3, 2]. One of the advantages of this algorithm is the automatic generation of distinctive word labels that can be used to describe the content of a cluster. For this paper, we will examine the application of this algorithm to large collections of legal documents and examine some refinements to the algorithm including pruning the word list used to cluster the documents and using k-means algorithm to recluster the documents. The clusters and the labels produced by PDDP are displayed by a document organizing agent and are helpful in understanding the differences between and similarities within groups of documents.

2 The Data

The dataset we used is a pilot collection of laws relating to alcohol legislation in all fifty states. This project arises out of a project at the School of Public Health of the University of Minnesota which tracks changes in alcohol law throughout the nation. The laws themselves vary greatly in size from a couple of thousand characters to over half a megabyte of text. They sometimes come with summaries that provide useful, though not always complete or accurate, information that can be used for an initial categorization. The goal is the make this data set browsable by the researchers at the School of Public Health. These researchers are not necessarily trained in the Law but are generally interested in knowing which topics have been addressed by politicians in a given year, in what way each topic has been addressed, and contrast those in one year with another year. In a typical year thousands of laws relating to alcohol are introduced in the state legislatures and several hundreds of them may be enacted into law. Over time, one could expect the total number of laws that need to be reviewed to be upwards of 100,000. For the purpose of developing the automated tools discussed in this paper, we have focused on on a pilot collection of just the laws enacted in 1998 (756 laws with nearly 8000 unique words, not including stop words). The tools, however, are intended to be applied the laws in other years, and to the many bills introduced each year that are never passed.

The method we use to achieve the goal of automatically organizing this information is to use an unsupervised method to organize the laws, help define the overall contours of alcohol law and aid in identifying laws that apply to specific policies (e.g., blood alcohol content limits, keg registration, alcohol excise tax rates, etc.) and discover patterns in the laws for further research.

One of the advantages of using PDDP to cluster this data is that it is unsupervised and therefore does not need to go through a training step. The PDDP method is also scalable to large datasets, as previous

experience has shown [3, 2]. For example, Fig. 1 illustrates how the running time of PDDP is approximately linear in the number of nonzeroes in the term frequency matrix. It is well known that k-means is also very scalable.

Another advantage is that the words which were most important in determining which partition each document fell into can be saved and provide insight into and a hierarchy for the structuring of the data. These labels are an automated way to describe what content the documents in a cluster have in common. Also, the clusters produced can be used to classify the data or provide a check to the existing classification. Once the clusters are produced the algorithm can be used to classify new data according the partitions already generated [2, 1].

3 Description of the Algorithms

In this section we describe the principal pieces that make up our proposed agent. This agent consists of several pieces:

1. Accept documents, count words and compute term frequency matrix.
2. Cluster documents and generate labels.
3. Produce a browsable output.

Step 1 is accomplished using standard data management tools. The counting of the words and generation of the term frequency matrix is based on off-the-shelf methods, including Porter's stemming algorithm [5] and standard sorting methods. Step 3 is carried out using the same techniques used to generate dendrograms (trees) for hierarchical clustering algorithms found in many standard statistical software packages. To make this usable in this domain, the generated trees must be browsable, and since we intend this browsing will be carried out using a standard off-the-shelf web browser, the generated trees have been linked to internally generated HTML pages. A sample of one of these HTML pages is shown in Figure 7. The process is unsupervised and automated, determined by prechosen parameters that that control the computational kernel and specify the features of the navigational tool. The agent is still under development, especially a user interface wrapped around the entire package that will allow a user to control the preprocessing, clustering, and final display.

In this paper, we discuss mainly step 2, specifically how to generate good labels automatically as a byproduct of the clustering already being carried out.

For all of the clusterings, step 1 is to process all the documents into an array of word count vectors. Stop words, common words such as conjunctions that provide no significant information about a document, are removed, common prefixes and suffixes deleted, and the remaining words are counted. The word lists are combined so that all the words present in all the documents make up one dimension of a matrix and all the documents make up another. Generally, the words are treated as variables and the documents as cases. The resulting matrix is very sparse as most laws contain only a small percentage of the total number of words found in all the laws. The word counts are then typically normalized to account for variations in document lengths. Clustering algorithms can now be applied to the data treating each word as a dimension in Euclidean space and the normalized word as a distance measure for that axis.

Step 2 was accomplished by combining the use of three different algorithms.

1. Principal Direction Divisive Partitioning (PDDP) was used to (a) create an initial clustering of the documents, (b) extract the set of "most distinguishing" words, those most useful in distinguishing between the clusters of documents, and (c) to cluster the set of most distinguishing words into groups that act as initial centers for a k-means iteration.

2. *(word pruning)* A filter was applied to the word vectors produced as a natural byproduct of the PDDP method, explained below. These word vectors, called principal direction vectors, give weights to each word, where the weight is proportional to the importance each word has in distinguishing each cluster from its neighboring cluster in the hierarchy. The filter can be set to select all words whose weight is above a certain threshold, or to simply select the top `nwords` from each vector. In the experiments shown here, we chose the latter, with `nwords` = 20.

3. The k-means algorithm was applied to the documents to form the final clusters that the user will see. This method was iterated until convergence. This method depends critically on the way it is initialized, and in order to make the result deterministic and repeatable, we chose to use the output from PDDP to initialize the k-means method. In previous experiments [7], it was found that one can initialize k-means with the centers of the document clusters from PDDP (produced as a natural byproduct of PDDP), but it was also found that k-means in this case often results in only a minor refinement of the PDDP clusters. Our experience with this dataset was consistent with these previous results. Hence we decided to seed the k-means algorithm in a different way. The "most distinguishing" words previously extracted were separately clustered by means of PDDP (the most handy method available) and an artificial center was constructed from each resulting word cluster. Each artificial center is an "indicator vector," having an entry equal to 1 for each word in the word cluster and a zero entry for each remaining word. Hence each artificial center is a vector of the same dimension as a vector of word counts for a document. The artificial centers were then scaled to unit length and used to seed the k-means method.

We briefly describe the PDDP method here – details can be found in [3]. The PDDP algorithm takes a top-down approach by partitioning the data into two clusters and further subdividing clusters into smaller partitions recursively until some stopping value (e.g., a specific number of clusters or a specific density or ratio of densities of clusters) is reached. Successive splits result in a hierarchical structure arranged into a binary tree. A variety of measures can be employed to determine which cluster is to be partitioned next. For this project, the cluster with the largest total scatter value is selected as the next to be partitioned. The total scatter value is a measure of the cohesiveness of the cluster and reflects the distance between each document in the cluster and the overall mean of the cluster [3, 4]. Using this value tends to result in clusters with similar numbers of documents.

Each partition itself is computed by calculating the direction of maximal variance (the principal component or principal direction vector) of the matrix of the cluster and splitting the cluster at the centroid or mean. The centroid for a given collection of documents $\mathbf{d}_1, \ldots, \mathbf{d}_m$, is the hyperplane anchored at $\mathbf{w}$ where $\mathbf{w} = (\mathbf{d}_1 + \cdots + \mathbf{d}_m)/m$. The split is made at the hyperplane anchored at $\mathbf{w}$ and perpendicular to the single leading eigenvector $\mathbf{u}$. The algorithm uses a Lanczos-based solver to take advantage of the fact that the matrix is sparse to calculate the leading eigenvector of the sample covariance matrix which is the principal direction. The largest entries in the centroid vector correspond to the most common words throughout the cluster. The largest entries in the principal direction vector correspond to those words that are most significant in placing a document in one partition or the other. These words are used to form the annotations for each cluster. They can be helpful in describing the documents in a cluster as well as the overall structure of the dataset.

As mentioned above, two other techniques were used in conjunction with the PDDP, word pruning and k-means clustering. The pruned word list is useful for two purposes. First, it is the list from which the automated cluster annotations are drawn. The culled words are not so useful in distinguishing between clusters, and the purpose of the annotations are to indicate to the user how a given cluster differs from the neighboring clusters. Second, the document dataset can be represented by the pruned word list as effectively as by the original word list, but with correspondingly reduced space. By identifying the small set of words that are distinctive for the document domain, this small set can be used to cluster future documents in the same domain instead of having to keep all the words. To illustrate that the pruned word list leads to a good representation of the original data, but with less space, we have compared the clustering using all the words with that obtained using only the pruned words in Figure 3.

For purposes of comparison, we also used PDDP as well as k-means clustering on the documents with the pruned word list. Figure 3 are confusion matrices showing how similar the clusterings, whether from PDDP with all the words, PDDP with the pruned word list, or k-means with the pruned word list.

We summarize the steps in the overall agent, showing how the various methods just described are assembled into an overall agent, in Figure 2. We will compare the document clusters that result from steps 3, 4a, and 5.

1. Document words are stemmed, stop words are removed, and then counted.
2. Word counts for documents are compiled into a term frequency matrix.
3. PDDP is run to cluster documents and generate pruned word list.
4. Clustering is rerun with the pruned word list using PDDP either:
   (a) to cluster documents, or
   (b) to cluster words that are used as initial centers for step 5.
5. k-means is used to cluster documents using labels from 4b as centers.

Figure 2: The procedure followed by the Agent.

4 Algorithmic Issues

Some of the issues that had to be addressed in order to apply the methods chosen to the data were: scaling the data, selection of stop words to be removed from the data, the stopping test to be used, and the word pruning criterion.

One issue that is peculiar to this dataset is the presence of laws that have many sections on different topics, so-called "garbage bills." For simplicity, we chose to keep all laws together, since we found it is far from a trivial task to determine the boundaries between the different topics, and even more fundamentally whether a given topic is really a new topic or a continuation of the same topic. Chopping the laws into different sections would have introduced additional overhead to the analysis, adding complication to the whole system. However, in an "industrial strength" agent, this issue would certainly have to be addressed.

The data could be unscaled, scaled by document length, or scaled with a TFIDF scaling [6]. Past experience with PDDP has shown that TFIDF scaling does not add much more accuracy than the simpler document length scaling [2], hence we chose the latter. Using totally unscaled data leads to unbalanced clusters using PDDP with single, large documents occupying a single cluster and large numbers of smaller documents in bigger clusters. We used a simple norm scaling because this has been shown to be sufficient for this method in the past.

The stop words removed from the data were those used in previous experiments reflecting the most common words in the English language with no initial effort made to identify stop words specific to the legal domain. However, after reviewing the most distinctive word labels, some words, such as the names of states, or words that did not provide useful distinctions between documents (e.g, bill, date, code, act, section, whereas, etc.) were added to the stop word list before running the algorithms. In general for PDDP, words that are common to all documents are less significant in dividing documents between clusters.

5 Computational Results

The success of the proposed automated agent depends on the quality of the clusters produced, the distinctiveness and descriptiveness of the automated labels generated for each cluster, and the distinctiveness of the words selected by the pruning process. It is hard to quantify any of these aspects except perhaps the first one, but even in this case only indirect measures are possible.

One way to verify the quality of the clusters is to see how robust they are as the algorithms used to produce them are changed. In order to facilitate comparisons between the different algorithms, the stopping test used was a specified number of clusters. After some experimentation the target of 40 clusters was selected as it minimizes the difference between the largest and smallest clusters without generating many small clusters with narrow common interests. The autonomous agent will use an automatic stopping test which compares the "within-cluster" scatter value to the "between-cluster" scatter value [3, 4].

The final clusters were fairly similar across the different techniques, as can be seen in the confusion matrices (Figure 3). It is seen that pruning the words has relatively little effect on the clusters, and refining the PDDP clusters with k-means had almost no effect at all.

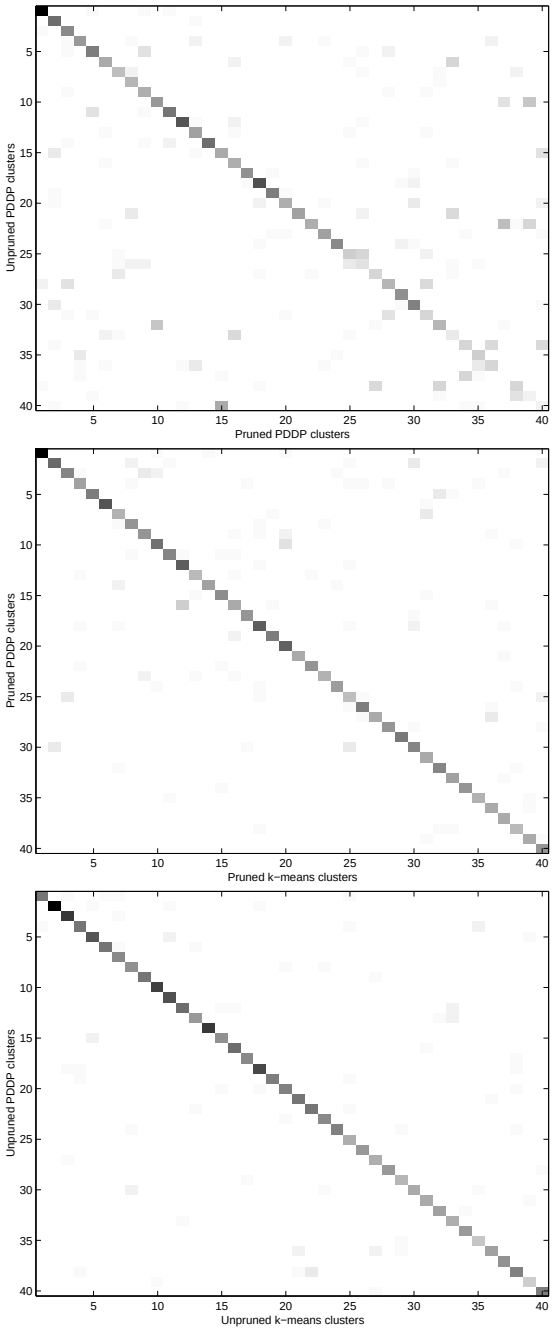


Figure 3: Confusion Matrix between Pruned and Unpruned PDDP and k-means using PDDP document clusters as starting point.

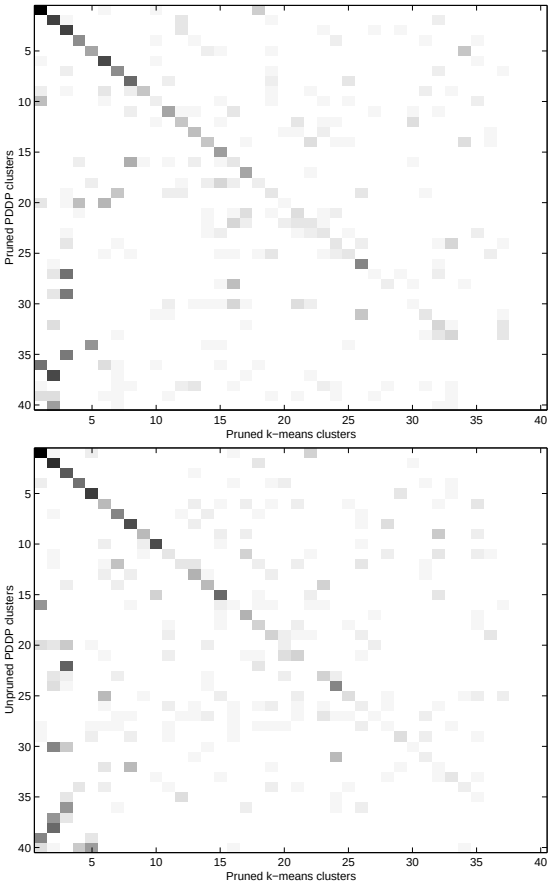


Figure 4: Confusion Matrices between Unpruned PDDP, Pruned PDDP and Pruned k-means using artificial centers as starting point

Words pruned from list		Words left after pruning	
abstract	accomplic	abus	annot
abstractor	accomplish	accident	applic
abut	accordingli	action	appoint
academ	account	accord	appropri
academi	accredit	addict	arrest
acceler	accru	addition	articl
accent	accrue	administr	assembli
accept	accumul	adopt	author
acceptor	accur	affair	awar
access	accuraci	age	beer
accessori	accus	agreem	benefit
accessto	accusatori	alcohol	beverag
accommod	achiev	amend	blood
accompani	...	appli	...

Figure 5: Sample words from pruned and unpruned word lists.

UNPRUNED:	test vehicl offic motor licens oper submit arrest chemic concentr blood alcohol
PRUNED:	test alcohol offic vehicl motor submit blood oper arrest concentr chemic refus
UNPRUNED:	applic board drug requir person licens license alcohol renew commiss liquor certifi
PRUNED:	applic board person requir drug licens license alcohol commiss profession certifi hear
UNPRUNED:	board counti read articl wicomico commission licens march annot replac salari appli
PRUNED:	board licens counti read articl wicomico commission march annot replac appli enact

Figure 6: Unpruned vs. pruned labels for similar clusters.

We investigated whether using a smaller set of only the most distinctive words, (the word pruning discussed in an earlier section) would produce equally good clusters and more meaningful labels. The final word list after pruning was only a fraction of the original word list in length (e.g., 265 words from an original 7762). The semantic content of the pruned word list is greater than the words that have been pruned as can be seen by comparing samples of the two (Figure 5). This difference is less noticeable when comparing the labels of similar clusters between pruned and unpruned PDDP clusters (Figure 6). This is because the less meaningful words, those which have less magnitude in the principal direction vector, are less important in determining which partition a document is assigned to.

The labels that the PDDP algorithm provides for each of the clusters are generally quite informative about the content of the specific laws. For example, one cluster with 12 laws in it has a label with words one associates with drug and alcohol treatment programs. An examination of the contents of the laws in that cluster shows that they all deal with drug and alcohol abuse treatment programs or related issues (Figure 7). The twelve most distinctive words for the cluster are listed at the top of the example. The context on the right is an excerpt from the document showing the context where the distinctive word is used.

An alternate way to initialize k-means was also used. In this method, which is detailed in the Description of Algorithms section of this paper, we transposed the term frequency matrix and used PDDP to cluster the words – after pruning them first. The results of this method show some differences with the original clustering, but overall this method gave similar results (Figure 4). In a few cases, some PDDP document clusters ended up being coalesced into a single k-means cluster (Figure 8), and in these cases, the coalesced clusters were actually cleaner, more cohesive, clusters.

The net effect of this then was to improve the quality of the clusters in terms of grouping like laws together which also resulted in less balanced clusters (more larger and smaller cluster sizes). There were 3 fewer clusters for a total of 37 after running k-means because three of the initial starting centers had no documents closer to them than any other center. The maximum and minimum cluster size for k-means was 82 documents and 2 documents. For the pruned PDDP clusters the maximum and minimum cluster sizes

Words: drug alcohol treatment fund court offend addict cost assembli abus driv committe

document	context (with distinctive word highlighted)
1997_CA_A_1784.htm	AB 1784, Baca. Alcohol and **drug** treatment for adolescents
1997_CA_SB_2015.htm	Existing law provides for the Medi-Cal **Drug** Treatment Program, under which
1997_FL_H_553.htm	convictions involving **drug** or alcohol shall be
1997_HI_H_2843.htm	RELATING TO **DRUG** DEMAND REDUCTION ASSESSMENTS
1997_OK_S_1361.htm	An Act relating to **drug** courts; amending Section 7, Chapter 359, O.S.L. 1997
1997_OK_S_645.htm	An Act relating to criminal procedure; creating the Oklahoma **Drug** Court Act;
1997_PA_H_679.htm	**Drug** and Alcohol Abuse; imposing duties on the Department of Health to
1997_TN_HJR_70.htm	alcohol and **drug** abuse treatment services, including the cost effectiveness
1997_TN_S_3097.htm	special fund to be known as the "Alcohol and **Drug** Addiction Treatment Fund"
1998_CO_S_25.htm	THE ADMINISTRATION OF THE ALCOHOL AND **DRUG** DRIVING SAFETY PROGRAM,
1998_LA_S_152.htm	(7) To a local public or private nonprofit agency involved in **drug** abuse prevention and treatment
1998_MD_H_8.htm	By: Delegate Menes (Chairman, Special House Committee on **Drug** and Alcohol

Figure 7: Sample illustration of browser output, showing cluster #77 from the tree of Fig. 9.

Cluster 57 (16):	establish retail town sale sundai hour holidai engag princip busi council pursuant
Cluster 58 (29):	applic board person requir drug licens license alcohol commiss profession certifi hear
Cluster 69 (19):	wine special class beer counti board fee articl licens baltimor liquor light
Cluster 70 (24):	beverag manufactur person amend wholesal vendor applic commiss sale rule local requir
Cluster 2 (82):	licens beverag alcohol wine applic author person sale fee license contain liquor

Figure 8: Pruned PDDP cluster (top) are consolidated into one k-means cluster (bottom).

were 39 and 12.

One example of this improvement can been seen in the way the 82 laws in the largest k-means cluster were mostly split between 4 PDDP clusters. All but 10 of the 82 laws were also present in the four most similar PDDP clusters and these four clusters shared most of their documents with the k-means clusters (16 out of 16, 23 out of 29, 16 out of 19, and 17 out of 24). Looking at the labels and the text of the laws we can see that the larger k-means cluster combined laws related to licensing from the 4 smaller clusters (Figure 8).

Another example of a consolidation of PDDP clusters into a more thematically coherent k-means is cluster 6 which pulls together laws related to motor vehicle license violations from PDDP clusters #50 (22 out of 25), #60 (9 out of 22), a few laws out of clusters #39 (4 out of 25), #49 (3 out of 20) and single laws scattered among several other clusters. Also, k-means cluster #15 contains all the documents in PDDP cluster #77 but also includes additional alcohol and drug treatment program related laws scattered throughout the other PDDP clusters. (Figure 10).

In summary, although all methods examined produced good results, significantly better results were achieved by using PDDP to prune the word list and generate a set of artificial centers that were then used to initialize the k-means step. Because we did not see similar improvements when using the PDDP document clusters to initialize the k-means centers, we conclude that the improvement was due to pruning process, the special method we used to seed the centers, or some interaction of these elements, and not simply to k-means algorithm itself.

6 The Agent

The agent we propose in this paper is formed by putting together the algorithms discussed in this paper as described in Figure 2. This agent is designed to be only semi-autonomous, in the sense that a user will invoke this agent when desiring to process a new or updated collection of bills. We have intentionally omitted an

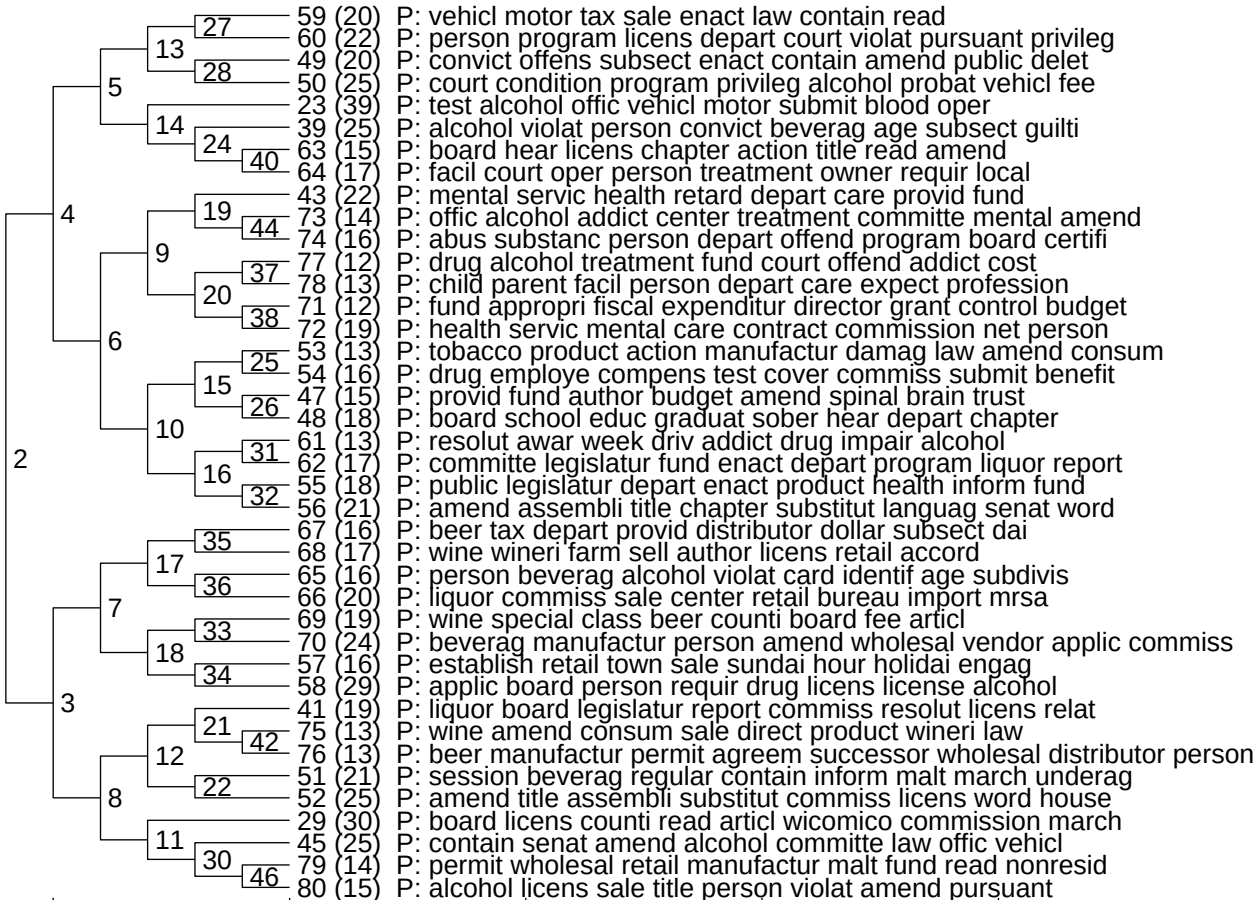


Figure 9: Tree structure for document collection, pointing the user to individual nodes in the tree.

initial step of collecting the documents, since each data source has a very different interface. In the example in the present paper, the documents were already collected by the School of Public Health. However, a separate process not described in this paper has been implemented to automatically invoke an off-the-shelf search engine to retrieve up to 1,000 documents containing certain keywords. The present agent is then invoked automatically to organize and label these documents.

Once invoked, the agent is entirely autonomous, requiring no human intervention. The input consists of set of text or HTML documents. The output of this agent will be a web browsable document of the form illustrated in Figure 9 (if PDDP output is used) or Figure 10 (if k-means is used). The user can click on this document to examine the detailed contents of each individual node, as illustrated in Figure 7.

We have illustrated the core algorithms that form the agent to autonomously process a text collection, in this case a specialized collection of legal documents. The purpose of this agent would be to apply these techniques to an unindexed collection of bills, such as those introduced but not yet passed. We have shown how the unsupervised top-down algorithm PDDP produces clusters of documents with labels that are automatically generated. We have seen these labels can provide useful information about the content of the documents in any cluster. Pruning the word list before clustering eliminates less meaningful and distinctive words and results in equally good or better clusters than the full list. We have also explored two different algorithms to process the documents after pruning: PDDP and k-means. Performing k-means on pruned word clusters as starting points produces clusters that tend to be less balanced in terms of number of documents in each cluster, but the larger clusters from k-means tend to be reasonably cohesive anyway, as illustrated by Figure 8.

Cluster 1 (67):	person alcohol violat test subsect provid oper licens vehicl amend court motor
Cluster 2 (82):	licens beverag alcohol wine applic author person sale fee license contain liquor
Cluster 3 (72):	alcohol beverag licens contain read amend person sale public enact law chapter
Cluster 4 (25):	vehicl motor person provid licens oper violat contain inform court amend owner
Cluster 5 (14):	depart servic enact rule legislatur contain program health inform public establish report
Cluster 6 (42):	person court licens vehicl violat convict depart alcohol motor program pursuant requir
Cluster 7 (28):	liquor licens alcohol person license sale amend provid retail contain commiss enact
Cluster 8 (34):	amend assembli contain enact public title delet ad house inform chapter alcohol
Cluster 9 (2):	applic licens alcohol beverag counti convict commiss read amend requir law contain
Cluster 10 (8):	oper person facil contain requir hear privileg subsect amend enact owner public
Cluster 11 (13):	program person amend chapter provid drug establish subdivis health servic contain requir
Cluster 12 (10):	test offic alcohol person law drug employe amend driv public read servic
Cluster 13 (6):	child court person unborn parent amend provid depart mother violat expect subsect
Cluster 14 (16):	servic depart provid report person health public abus mental contract inform counti
Cluster 15 (28):	treatment drug alcohol abus provid program court health servic depart person substanc
Cluster 16 (26):	substanc abus mental servic depart health provid treatment public drug retard contain
Cluster 17 (21):	fund appropri depart servic provid fiscal public author budget program director health
Cluster 18 (19):	board licens alcohol term counti hear provid person drug appoint read contain
Cluster 19 (10):	facil public amend oper author ad follow contain program mean depart assembli
Cluster 20 (12):	health mental servic care depart substanc abus provid program fund commission report
Cluster 21 (17):	public provid establish law enact contain amend assembli health ad addition person
Cluster 22 (9):	permit alcohol beverag sale holder liquor contain person provid wine read division
Cluster 23 (20):	contain enact ad read public sale beverag senat effect session report inform
Cluster 24 (13):	tax amend provid person law chapter sale subdivis licens author requir articl
Cluster 25 (15):	commiss licens assembli report resolut provid special contain amend relat applic public
Cluster 26 (27):	wine licens wineri sale amend retail contain alcohol person wholesal author beverag
Cluster 27 (4):	limit licens fee alcohol hold beverag sale dollar citi person pursuant public
Cluster 28 (16):	action amend law chapter tobacco effect read relat product contain person damag

Figure 10: Center labels for k-means clusters. There is no tree because k-means does not produce one.

References

[1] D. Boley and V. Borst. Unsupervised updating of a classification tree in a dynamic environment. In *Autonomous Agents'99 Conference*, pages 390–391, 1999.

[2] D. Boley, M. Gini, R. Gross, S. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. Web document categorization. *Decision Support Systems*, 27:329–341, 1999.

[3] D. L. Boley. Principal Direction Divisive Partitioning. *Data Mining and Knowledge Discovery*, 2(4):325–344, 1998.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification and Scene Analysis.* John Wiley, 2nd edition, 2001.

[5] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[6] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[7] S. Savaresi and D. Boley. On the performance of bisecting k-means and PDDP. In *First SIAM International Conference on Data Mining (SDM'2001)*, 2001. SIAM.