Detecting Outlier Sections in US Congressional Legislation

Elif Aktolga
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts Amherst
Amherst, Massachusetts
elif@cs.umass.edu

Irene Ros, Yannick Assogba
IBM Watson Research Center
Visual Communication Lab
Center for Social Software
Cambridge, Massachusetts
{iros, yannick}@us.ibm.com

ABSTRACT

Reading congressional legislation, also known as bills, is often tedious because bills tend to be long and written in complex language. In IBM Many Bills, an interactive web-based visualization of legislation, users of different backgrounds can browse bills and quickly explore parts that are of interest to them. One task users have is to be able to locate sections that don't seem to fit with the overall topic of the bill. In this paper, we present novel techniques to determine which sections within a bill are likely to be outliers by employing approaches from information retrieval. The most promising techniques first detect the most topically relevant parts of a bill by ranking its sections, followed by a comparison between these topically relevant parts and the remaining sections in the bill. To compare sections we use various dissimilarity metrics based on Kullback-Leibler Divergence. The results indicate that these techniques are more successful than a classification based approach. Finally, we analyze how the dissimilarity metrics succeed in discriminating between sections that are strong outliers versus those that are 'milder' outliers.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Selection process

General Terms: Experimentation, Measurement, Algorithms

Keywords: Outlier Detection, Language Modeling, Dissimilarity

1. INTRODUCTION

Reading congressional legislation is often tedious and can be demanding to both the expert and average citizen. Bills are often written in complex legalese, which makes reading them a challenging task. Also, the political process of drafting legislation results in new content being added and old content being modified. This sometimes makes the overall gist and structure of a bill difficult to follow. We observe in the legislation data that bills that had content added to them over time are more likely to cover a wider range of topics. Such bills also discuss more varied aspects of the matter and contain sections whose contents represent slight, but still somewhat related shifts from the main topic.

IBM Many Bills [3, 13] is a web based visualization of congressional legislation aiming to make reading bills easier for lay users by providing a visual interface to the various components of a bill. The problem of understanding the structure of a bill is particularly acute in long bills, in which case it would be very helpful for the reader to know exactly where to look for parts of interest within a bill. One way IBM Many Bills supports this is by labeling individual sections with their main topic and color coding these topics to make them easy to identify within a bill. Another region of interest for users such as journalists, watch dog groups and concerned citizens are sections whose content differs from the main topic of the bill. This may indicate that the section is worth paying more attention to.

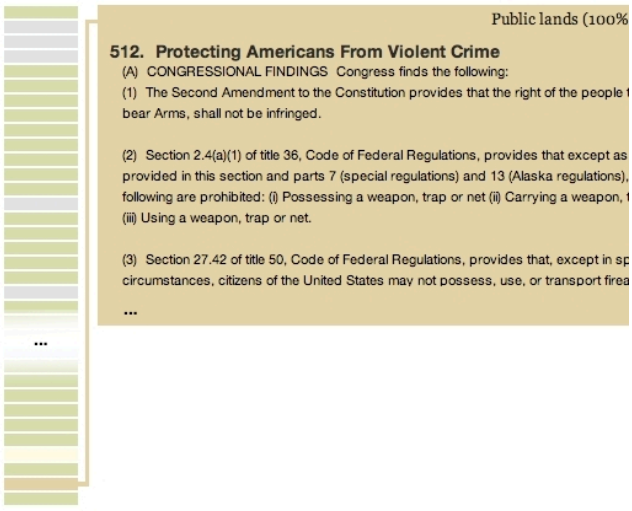

Figure 1: Example of a bill about credit cards with an outlier section about gun control.

To support the latter use case with an example, Figure 1 shows H.R. 627, Credit Card Accountability Responsibility and Disclosure Act of 2009, a bill about credit cards and consumer protection in the IBM Many Bills visualization. Each rectangular bar represents a section, the smallest unit of a bill. Towards the end of this bill there is a section about ensuring the right to carry guns in national parks en-

SIGIR'11, July 24–28, 2011, Beijing, China.

titled 'Protecting Americans from Violent Crime.' This is an unusual section within the context of credit cards, finance and consumer protection and is made particularly interesting because of its existence within this specific bill. We refer to such interesting and unusual sections as *outlier sections*. Our aim in this work is to automatically find such outlier sections in bills.

Unlike the example shown in Figure 1, most bills are consistent in their content and may contain minor topic shifts, or what we identify as *mild outliers*. Whether a section is an outlier or not depends on how important (and how dramatic) its topic shifts in relation to the rest of that bill, as determined by the reader. To illustrate this using the credit card bill described above, a section within it titled 'Privacy Protection for College Students' could be considered a mild outlier or a topic shift, because it still discusses matters related to credit cards and finance, but focuses on a specific target group – college students. This section would not be nearly as interesting to a user looking for anomalies as the one about gun control, but still may be worth noting to some readers. We observe through examples such as this one, that determining whether a section is an outlier is a challenging task because the decision is somewhat subjective, and depends on the context the section appears in, i.e. the content of the bill.

Another issue is that we want to control the number of falsely identified outlier sections shown to the user. The IBM Many Bills application that motivated this work is geared towards members of the general public. Because we do not want them to lose trust in the predictions of our algorithms, we aim at keeping the number of false positives low even at the cost of missing a true positive. We are in effect willing to trade recall for increased precision to a limited extent in order to maintain user expectations in pursuance of our goals.

Given that there is no pre-existing truth data for outliers in US Congressional Legislation, and the fact that outliers are contextual to the bills they appear in, we take a generative approach to detecting outliers within bills. Language models are created from different units of a bill, such as its individual sections, the entire bill text, or even from all the bills within a category, such as all 'Finance' bills. We then apply different dissimilarity measures to compare the section language models of a bill to other language models built from the different units just discussed, from which we obtain a ranking of the sections indicating the probability of each section being an outlier. We apply a threshold to this outlier score for the final decision of whether the section is marked as an outlier or not.

The most promising techniques use a 2-step approach: the main topical sections within a bill are first detected by ranking the top most relevant sections with respect to the bill's title keywords. We then apply dissimilarity measures between section language models of the main sections and remaining sections to obtain the final ranking for outlier sections. We use various dissimilarity metrics based on Kullback-Leibler Divergence [8, 16]. One of them is Kullback-Leibler Divergence Contribution (KLC), which we introduce based on inspiration from Lawrie and Croft's work [19]. The measures aid in distinguishing between outlier and non-outlier sections because they capture the following: (1) terms occurring in potential outlier sections that do not occur in main sections contribute to the outlier score with a positive KL Divergence value; (2) important topical terms contained in main sections that are less frequent in outlier sections contribute to the outlier score with a negative KL Divergence value. By utilizing this information in different ways, some of the dissimilarity metrics are more successful in catching strong outlier sections efficiently, whereas others do better on 'milder' outliers.

We also experiment with an approach that is not based on language modeling: since the category or classification information indicating the main topic for an entire bill (e.g. 'finance' or 'health') is the only truth data we have for the whole dataset, we also compare a classification-based technique to detecting outliers to the language modeling based approaches.

The remainder of this paper is organized as follows: Section 2 details previous and related approaches in the area; Section 3 describes the techniques that we compare to each other for the outlier detection task. In Section 4, we then evaluate the approaches on the 2009 US Congressional Legislation dataset. Section 5 contains an analysis of the KL Divergence Contribution dissimilarity metric that we use and its effect on outlier detection, after which we conclude the work in Section 6.

2. RELATED WORK

Outlier detection first emerged in statistics [4], where the aim is to find data points standing out from a certain distribution — these are typically numeric outliers in data. However, to be able to mine outliers with statistical techniques, characteristics about the distribution of the data must be known in advance, which may be a disadvantage if such information is not available.

Clustering techniques are unsupervised approaches that do not require prior knowledge about the distribution of the data [15, 25]. The aim is to group data into smaller sets that exhibit some structure, for which often a distance measure is required. For example, in flat clustering distance is measured in terms of closeness to the k-nearest neighbor [25]. In such an analysis, data points standing out from the grouped subsets of data would be identified as outliers. The challenge in our task would be to determine k in advance, since it varies depending on the content of the bill itself. Therefore, flat clustering techniques are rather unsuited to our setting. Hierarchical clustering methods are more appropriate in that k is not required [28]. A common technique is to use dendogram cutting for obtaining clusters at a certain similarity level. Still, determining the right similarity level depending on the content of the bill, and identifying which of the clusters constitute outlier sections constitute challenges.

Recently, there has been some research on mining non-numeric textual outliers from web data [1, 2], in which outliers are mined with more conventional text-mining and n-gram based methods combined with ideas from clustering. These approaches aim at identifying whole documents as outliers within a category both in the presence and absence of a domain dictionary. Our work differs in that: (1) to make conclusions about outlier sections, we do not use any technique requiring prior assumptions about the data such as a domain vocabulary, or any source other than the legislative texts; (2) we detect outliers in the form of sections within a bill, rather than the whole document. The background information we use for making our decisions about outliers

come from bill text in different units, but the outliers we detect are at the section level.

In the data mining community, outlier detection research has been done within the context of 'financial fraud detection' [26]. These techniques are mostly machine learning based and target only one type of outliers in text – fraud. Again, such techniques are more suitable when particular outliers are being observed that exhibit a clear pattern which is 'learnable' by such algorithms.

Another related group of work is that on text segmentation [5, 11, 12]. The aim in this research is to identify topic shifts to then break the text down to smaller sentences at such topic boundaries. We do not modify the given section structure of the US Congressional Legislation, but it would be a worthwhile extension of our work to segment long sections further down into smaller subsections to find more fine-grained outliers.

Kullback-Leibler Divergence [8, 16] has widely been applied in information retrieval in the context of language modeling both as a (dis)similarity metric [9, 19] to distinguish between models, as well as a ranking function [17, 22, 30] to rank documents. Inspired by this, we explore KL divergence based dissimilarity metrics for measuring the entropy between different units of language models. One of the metrics we arrive at is called 'KL Divergence Contribution' (Section 3.4.3), which – unlike traditional KL Divergence – measures the undirected weights of the contributions of terms towards the overall KL divergence score. We analyze the measure by means of examples in Section 5.

IBM Many Bills is an example of open government in that it promotes the transparency of bills and the interactivity between citizens and the government. Research in the areas of e-government and e-transparency has shown that this is an important issue [14, 29]. There has been a lot of research in the AI & Law area for tools to assist people and specialists such as lawyers with searching, viewing, and working with government data [18, 24]. We hope that with the addition of this outlier detection feature to the Many Bills system we can further expand the accessibility of these documents to more people outside the legal sphere.

3. OUTLIER DETECTION METHODS

In this section we detail various techniques of outlier detection. The first one described is based on the classification of sections and bills into topics, whereas the remaining methods are based on language modeling [23].

3.1 Classification approach (CL)

For this technique, which serves as our baseline, we utilize categories manually assigned to bills by the Congressional Research Service (CRS) as well as categories automatically assigned to sections by a trained classifier. The classifier is trained as follows: every bill in the original dataset has been assigned a human-determined *top category* describing the main topic of the bill. There is also a given set of categories for other subjects within the bill. Typical examples of categories are 'health', 'finance', 'science & technology' etc. Unfortunately, these categories do not point to specific sections within the bill, so this data strictly refers to the entire bill text. In order to obtain topic categories for sections, we train a multi-class maximum entropy document classifier for 83 of the classes that the CRS assigns as top categories to bills. The classifier is trained over 59552 bills from the past 9 years of congress with 10-fold cross validation. The Mallet [21] toolkit is used to train the classifier and then applied to the sections of bills from the 111th congress (2009-2010) to generate a topic for each section of a bill.

Given a bill D, we determine the probability of a section $s \in D$ being an outlier under the classification approach (CL) as follows: Let s_{class} be the category that was assigned by the classifier to s, further let $D_{cats} = \{d_{\text{c}_1}, d_{\text{c}_2}, \cdots, d_{\text{c}_n}\}$ be the set of categories manually assigned by CRS to D (this set includes the 'top category'). Then:

$$P_{\text{outlier}}(s|D_{cats}, s_{\text{class}}) = 1 - \max_i \text{Popularity}(s_{\text{class}}, d_{\text{c}_i}) \quad (1)$$

where i iterates over the elements of D_{cats} and the Popularity measure describes how discriminative this section-bill category pair is with respect to the most common section category seen with this bill's categories in the corpus:

$$\text{Popularity}(s_{\text{class}}, d_{\text{c}}) = \frac{\text{cnt}(s_{\text{class}}, d_{\text{c}})}{\max_j \text{cnt}(t_{\text{class}_j}, d_{\text{c}})} \quad (2)$$

where $\text{cnt}(s_{\text{class}}, d_{\text{c}})$ is the number of times s_{class} occurred in a bill with category d_{c} with respect to the whole corpus and t_{class} is the category assigned by the classifier to a section t that d_c is seen with. Equation (2) is normalized with $\max_j \text{cnt}(t_{\text{class}_j}, d_{\text{c}})$, which is the most frequently encountered section-bill category pair with respect to d_{c}.

The intuition for s being an outlier depends on the degree of inverse popularity of s_{class}: $P_{\text{outlier}}(s|D_{cats}, s_{\text{class}})$ is higher the less popular the co-occurrence of s_{class} is with any element of D_{cats}. We maximize over all the popularity scores of D_{cats} because we want to give each s the greatest chance of not becoming an outlier. This helps in reducing the number of falsely identified outliers found in a bill.

3.2 Language Modeling

The remaining outlier detection techniques described below are based on language modeling [23], i.e. we create a unigram bag-of-words based representation of a unit of text in which the probability of each term in that model is determined by means of maximum likelihood estimation (MLE) as follows:

$$P_{\text{MLE}}(t|\Theta_U) = \frac{\text{cnt}_U(t)}{|U|} \quad (3)$$

where Θ_U is the language model of the unit of text U that token t is sampled from and $\text{cnt}_U(t)$ is the frequency count of t in U. Hence, probabilities for tokens occurring in certain units of text are determined based on their frequency counts with respect to the total number of tokens in those texts. Since the dissimilarity metrics we use for comparing language models are all based on Kullback-Leibler Divergence (KLD) [8, 16], we smooth the language models in an appropriate manner as detailed in Section 3.4.1.

In this work, we infer language models from different units of texts, which are explained in the next section.

3.3 Units of Language Models

We consider three different units of language models for outlier detection: category, document and section. The language models are always built in a unigram MLE fashion as described in Section 3.2. Note that these are independent of

the congressional legislation data we are using. They could be applied to any corpus consisting of documents that can be split into sections or paragraphs. The general topic of a document as determined by a classifier or manually is only required for the category model (Section 3.3.2). In all cases we used a 592 term stopword list containing common stopwords as well as a few legislation specific stop terms such as 'act', 'chapter', 'clause' etc. to filter the text of the sections.

3.3.1 Document Model

We infer the language model Θ_D given a document or a bill D. Then the likelihood of a section $s \in D$ being an outlier, given Θ_D, can be stated as:

$$P_{\text{outlier}}(s_{\text{LM}}|\Theta_D) = \text{dissim}_{\text{norm}}(s_{\text{LM}}||\Theta_D) \tag{4}$$

where dissim is any dissimilarity metric described in Section 3.4, and s_{LM} is the language model inferred from s. The dissimilarity scores across all sections s in D are normalized to probabilities between 0 and 1, which is indicated by *norm*.

3.3.2 Category Model

We infer the language model Θ_C given a category C of a document D whose sections are being ranked for outlier probability. This means that all documents $\mathcal{D}$ having category C are included in Θ_C. Then the likelihood of a section $s \in D$ being an outlier, given Θ_C, can be formulated as:

$$P_{\text{outlier}}(s_{\text{LM}}|\Theta_C) = \text{dissim}_{\text{norm}}(s_{\text{LM}}||\Theta_C) \tag{5}$$

where dissim is any dissimilarity metric described in Section 3.4, and s_{LM} is again the language model inferred from s. norm indicates that the scores are normalized between 0 and 1 across all sections s in D.

3.3.3 Section Model

For the section model approach we first infer language models Θ_{S_i} for each section $s_i \in D$. A single comparison between a section $s \in D$ and s_i is then achieved as follows:

$$P_{\text{dsm}}(s_{\text{LM}}|\Theta_{S_i}) = \text{dissim}_{\text{norm}}(s_{\text{LM}}||\Theta_{S_i}) \tag{6}$$

where s_{LM} is the language model inferred from s. More specifically, this denotes the probability of how dissimilar s_{LM} is from Θ_{S_i}. However, for determining the outlier probability of s, we need to know how s compares to each $s_i \in D$ *on average*:

$$P_{\text{outlier}}(s_{\text{LM}}|\Theta_S) = \frac{\sum_{i=1}^{n} P_{\text{dsm}}(s_{\text{LM}}|\Theta_{S_i})}{n} \tag{7}$$

where Θ_S denotes the average section model. Note that in total we perform n comparisons for each section $s \in D$, which altogether take $O(n^2)$. Before summing the comparisons, we normalize the dissimilarity scores for a section s across all section models Θ_{S_i} to probabilities between 0 and 1 as in the other models. $P_{\text{outlier}}(s_{\text{LM}}|\Theta_S)$ thus quantifies how probable s is to be an outlier, given that we have compared it to each other section s_i in D. Finally, we can rank all the sections $s \in D$ in order of their average dissimilarity scores to the section model Θ_S.

3.4 Dissimilarity Metrics

For determining which sections are outliers, we evaluate different dissimilarity metrics with the language models built using the modeling approaches described in Section 3.3. The dissimilarity metrics we choose here are all applicable to the language modeling framework. In fact they are all based on Kullback-Leibler Divergence [8, 16].

3.4.1 KL Divergence (KLD)

Kullback-Leibler Divergence (KLD) is a well-known non-symmetric measure for determining the difference between two probability distributions [8, 16]. In this context, it measures the difference between the two language models P and Q, where Q is the base model:

$$KLD(P||Q) = \sum_{i \in P \wedge Q} P(i) \cdot \log_2 \frac{P(i)}{Q(i)} \tag{8}$$

To be able to apply KLD to language models we smooth the models as follows: (1) If $i \in Q \wedge i \notin P$, then by definition of KLD, this would be $0 \log_2 0 = 0$ for token i, so we do not smooth in this case; (2) If $i \notin Q \wedge i \in P$, then by definition of KLD this would be ∞. For simplicity, we apply Add-1 Smoothing [6] in this case so that i is assumed to occur only once in Q, and the probabilities for other tokens in Q are adjusted accordingly. This prevents the overall KLD score from becoming ∞. With ∞ scores, distinguishing between sections and ranking them would become impossible.

Intuitively, these choices have the following effects on outlier detection: case (1) denotes that the base model Q might include additional tokens that are not contained in the candidate outlier section P that it is being compared to, which has no influence on P's KLD score. For example, for finding outliers in the credit card bill from Figure 1, a typical base model section Q, would discuss finance and credit card related issues with tokens such as 'credit', 'card', 'interest' etc., whereas P, say a section about credit card issues regarding college students, might not use all the credit card related tokens in Q (it might be missing 'interest' for instance). In this case, the mere absence of tokens in P that were present in Q do not make P a stronger outlier according to the choice of smoothing for (1).

Case (2) on the other hand refers to P including a new token previously unseen in Q, resulting in larger relative entropy: this may indicate that the candidate outlier section P discusses other issues than the base model Q, which contributes to its likelihood of being an outlier, resulting in a higher score. To refer back to the credit card example: since section P is about credit card issues regarding college students, it mentions 'college', 'students' and other tokens that do not occur in the base model section Q. These make P distinct from Q, which results in a higher score.

If P includes a token common to Q, the more its usage differs from that in Q, the larger is the impact on the KLD score. This impact is either in a positive direction if $P(token) > Q(token)$, and negative otherwise. In summary, this metric does not 'penalize' a section for missing tokens, but it is penalized for new (previously unseen) tokens and varying usage of previously seen tokens.

3.4.2 JS Divergence (JS)

The symmetric version of KLD, Jensen-Shannon divergence is defined as follows [20]:

$$JS(P||Q) = \frac{1}{2}KLD(P||M) + \frac{1}{2}KLD(Q||M) \quad (9)$$

where $M = \frac{1}{2}(P + Q)$. We apply this measure to the language models P and Q; M is built as a new language model from P and Q accordingly.

3.4.3 KL Divergence Contribution (KLC)

The contribution of a token i towards the KLD score was first characterized in Lawrie and Croft's work [19]: it is 0 if $P(i) = Q(i)$, positive if $P(i) > Q(i)$, and negative if $P(i) < Q(i)$. In previous work it was discovered that tokens with high contributions towards KLD are topical terms, whereas those with low contributions are less likely to be about the topic [10].

tokens	sum of |contributions|
arms	0.099
bear	0.099
firearms	0.099
fish	0.099
land	0.099
protecting	0.099
...	...
payment	8.3
account	8.7
fees	9.3
consumer	19.9
card	24.8
credit	30.3

Table 1: Analysis of token contributions towards the KL Divergence score in all the sections of the credit card bill from Figure 1.

We analyzed the validity of this statement for H.R. 627, the credit card bill from Figure 1. To do so we chose the 10 most representative credit card related sections of the bill as base models Q and compared them to the remaining sections P in the bill. By ignoring the direction of the contributions of tokens (positive or negative), it became evident that topical terms such as 'credit', 'card', 'fees' etc. were present in most of the sections with varying usage, whereas more off topic terms such as 'firearms', 'fish' etc. only occurred in very few sections. By considering the usage of terms across all the sections as a sum of the absolute value contributions as shown in Table 1, we can see this difference.

As emphasized in Section 3.4.1, a section P's outlier probability increases the more the use of language differs from the base model Q. Note that in traditional KL Divergence, positive and negative contributions of different tokens cancel each other out and weaken the overall score of a section. However, the use of language in an outlier section may differ from the base section both in terms of *fewer topical tokens*, such as the sparse usage of tokens like 'credit', 'card', 'account' etc., and in terms of *more off-topic tokens*, such as 'firearms', 'weapons' etc., which is the difference in usage of tokens that we might like to measure. Due to the attenuating effect of KLD with negative scores, this difference is evened out to some regard. We therefore introduce the measure KL Divergence Contribution (KLC) as follows:

$$KLC(P||Q) = \sum_{i \in P \wedge Q} |P(i) \cdot \log_2 \frac{P(i)}{Q(i)}| \quad (10)$$

Note that the only difference to KLD is the absolute value inside the formula. This allows us to keep the weight of each token's contribution regardless of its direction. This measure is one of our contributions in this paper, since we extend the notion of single term contributions towards the KLD score [19] to a measure that captures the sum of absolute value term contributions with respect to two distributions.

3.4.4 Symmetric KL Divergence Contribution (KLCSYM)

Analogous to Jensen-Shannon divergence (Section 3.4.2), which is a symmetric version of KL Divergence, we can also use a symmetric version of KL Divergence Contribution:

$$KLCSYM(P||Q) = \frac{1}{2}KLC(P||M) + \frac{1}{2}KLC(Q||M) \quad (11)$$

where $M = \frac{1}{2}(P+Q)$. Again, we apply this to the language models P and Q. M is built as a new language model from P and Q accordingly.

3.5 2-step approach (2S)

Among the three models we introduced in Section 3.3, the Section Model is the slowest to compute, since it requires a comparison between all sections in a bill in a quadratic manner. This becomes a problem when the section model is applied to long bills. We propose an alternative 2-step approach that reduces the number of comparisons: instead of applying dissimilarity measures directly to language models, we first locate the main sections in a bill D that are topically representative. For this, we score and rank each section S using the bill's title terms T by means of Okapi BM25:

$$\text{score}(s, T) = \sum_{t \in T} \text{IDF(t)} \cdot \frac{\text{tf}(t, S) \cdot (k_1 + 1)}{\text{tf}(t, S) + k_1 \cdot (1 - b + b \cdot \frac{|S|}{\text{avgSLength}})} \quad (12)$$

where $\text{tf}(t, S)$ is the frequency of title term t occurring in section S, avgSLength is the average section length in the corpus, and $k_1 = 2, b = 0.75$ are constants. T is usually a fairly long title query despite stopping with the 592 terms stopword list described previously. By scoring sections in this way, we obtain a ranking from which we choose the top m sections as the main sections. The choice of m depends on $|D|$, i.e. #sections in D, which we tuned during training to include between 9% and 55% of sections in a bill. For shorter bills, we include a larger number of sections relative to the length of the bill to reliably capture the main content, whereas for longer bills this percentage is smaller.

The second step then consists of comparing the top m sections in D to the remaining $k = n - m$ sections by employing a dissimilarity measure (Section 3.4), where $n = |D|$ in terms of sections. This is just like using the Section Model (Section 3.3.3) on the m and k sections:

$$P_{\text{outlier}}(s_{\text{LM}}|\Theta_{S^*}) = \frac{\sum_{i=1}^{m} P_{\text{dsm}}(s_{\text{LM}}|\Theta_{S_i})}{m} \quad (13)$$

where s_{LM} is one of the k remaining sections and Θ_{S^*} denotes the average section model built from the main m sections only. $P_{\text{dsm}}(s_{\text{LM}}|\Theta_{S_i})$ is estimated as in Equation (6).

In this comparison, the top m sections are by definition excluded from the candidate outlier sections k and they can therefore never become outlier sections. Intuitively, sections that were determined to be topically representative of a bill should never become outlier sections. So while this approach reduces the number of comparisons between sections, it also protects the main sections from becoming outliers by excluding them.

4. EXPERIMENTS

4.1 Data and Implementation Details

We evaluate the outlier detection methods on several subsets of the 111^{th} Session (2009-2010) of US Congressional Legislation (data obtained from GovTrack [27]), which is a real-world dataset. This includes 7940 bills from 37 categories both from the House of Representatives and from the Senate. These bills have 110,688 sections in total, with the average number of sections per bill approximately equal to 14. The average section length is 2227 characters. Each bill comes with metadata from which we use: the official title (e.g. 'To amend the Internal Revenue Code of 1986 to provide for an extension of the employer wage credit for employees who are active duty members of the Uniformed Services.'), short title (e.g. 'Small Business Supporting our Troops Act of 2009'), the top category assigned to a bill (e.g. 'taxation'), further categories (e.g. 'small business', 'income tax credits'), and the section texts. For the 2-step approach described in Section 3.5, we merge the official and short titles of a bill to generate the query used for ranking the sections in the first step of the process. The titles are stopped using the same 592 term stop list for the language model and contain important key phrases for a bill.

The human-labeled training and test sets were created by randomly choosing bills from the corpus across a number of different categories. The training set consists of 13 bills with a total of 683 sections while the test set consists of 11 bills with 823 sections. We have complete outlier judgments for these sets.

4.2 Evaluation Methods

To evaluate the outlier detection methods we compare against human judgments of bills' sections. The judgments come from three annotators that are not experts in the legislation area, but have extensively dealt with bills in addition to having sufficient familiarity with the outlier detection task. Each section is assigned one of the following labels: **1** for not being an outlier, **2** for a mild outlier, or **3** for being a strong outlier. There are reasons for preferring this scale over a binary one: we observed that annotators tend to be uncertain about some sections since the decision is often based on subjective impressions. Thus, the 'mild outlier' label represents the sections that show topic shifts not deemed completely off-topic. We observed that many bills have mild outliers (around 5%), whereas the annotators' use of the label **3** proved more conservative: only 1% of all ratings were a **3**. To obtain the final judgment for a section we average the judgments from the annotators and round the averages accordingly, so that a section with {1, 1, 3} judgments is assigned a **2**; a section with {1, 1, 2} is assigned a **1**.

We measured inter-annotator agreement for each judgment from the three annotators with Cohen's Kappa (weighted) [7]. Cohen's Kappa is 0 or smaller if there is no agreement between the annotators, and it approaches 1 as agreement becomes perfect. For our judgments, the mean Kappa for 3 pairwise evaluations over the full set is 0.626 with a p-value of < 0.01 using the z-test. This signifies a good agreement between the annotators.

For all results reported on the test set in Section 4.3 we map the truth judgments to outlier scores for sections as follows: a section is counted as a *correct outlier* (true positive) if it was marked as a mild outlier (**2**) or a strong outlier (**3**) in the truth judgments **and** the algorithm being evaluated returns an outlier score of at least 70% for this section. This threshold was determined to be optimal for the algorithms in the training data. If a section has an outlier score of 69% or less and it was judged with a **1** by the annotators, it is interpreted as a correct non-outlier section (true negative).

We evaluate the outlier detection methods using well-known measures: (1) *Precision* measures the percentage of correctly detected outliers among all found by the algorithm; (2) *Recall* measures the percentage of correctly identified outliers among all actual outliers as determined in the truth data; (3) *Accuracy* measures the percentage of correctly marked sections (both true positives and true negatives); (4) *% Sections Predicted to be Outliers* measures the fraction of sections that were detected as outliers versus those that were not. This is with respect to all the bills that a technique examines. Hence, it indicates the percentage of outlier sections found in 'the corpus' by a particular approach. Note that % sections predicted as outliers is inversely proportional to accuracy. This measure is important because precision and recall on their own are not indicative enough of how much noise an approach introduces. We observe that the smaller the % sections predicted as outlier sections by a technique – together with a high precision and accuracy, the more reliably the method finds correct outliers (Table 2). Therefore, in the following section where we look at the results, we will favor techniques that exhibit these characteristics.

4.3 Results

We evaluate each of the techniques considered in Section 3 on the test set. The results are shown in Table 2.

First, we note that the classification approach performs worse than all but the worst performing language modeling based approach, marking 13.1% of sections as outliers, and having low precision (0.368) and accuracy (0.795).

Additionally, we observe that the numbers for precision, accuracy, and % sections predicted to be outliers improve as the unit of the language model used becomes smaller (Category > Document > Section = 2S). This indicates that comparing section language models to larger models is noisy for the outlier detection task. When comparing the two smallest units of language models – Section and 2S – the 2-step approaches perform better than direct application of dissimilarity measures to the section models. 2S-JS achieves the best results in terms of precision (0.542) and accuracy (0.914) *while* maintaining a lower % sections predicted to be outliers in the corpus (2.8%).

When using any of the 2-step approaches the differences between dissimilarity measures are not significant except when using symmetric KLC (KLCSYM), which performs the worst no matter which granularity of language model is used. The section scores are greatly boosted with this

Table 2: Evaluation of Outlier Detection Methods. All results are averages run on the test dataset. CL=Classification approach, Doc=Document Model, Cat= Category Model, Sec=Section Model, 2S= 2-step approach. Significant results (*) between 2S-JS and other methods with p-value < 0.05 using the paired two-sided t-test are marked.

Technique	*% Sections Predicted to be Outliers*	*Precision*	*Recall*	*Accuracy*
CL	13.1	0.368*	0.434	0.795*
Cat-KLD	13.9	0.311*	0.50	0.743*
Cat-JS	18.2	0.294*	0.503	0.657*
Cat-KLC	18.0	0.294*	0.54	0.66*
Cat-KLCSYM	50.0	0.273*	0.661	0.429*
Doc-KLD	11.9	0.271*	0.273	0.808*
Doc-JS	12.5	0.325*	0.436	0.78*
Doc-KLC	16.3	0.316*	0.486	0.743*
Doc-KLCSYM	39.6	0.313*	0.649	0.633*
Sec-KLD	4.2	0.343*	0.271*	0.872*
Sec-JS	3.7	0.35*	0.371*	0.873*
Sec-KLC	8.5	0.34*	0.471	0.842*
Sec-KLCSYM	11.8	0.327*	0.496	0.798*
2S-KLD	3.8	0.528	0.429	0.895*
2S-JS	**2.8**	**0.542**	0.446	**0.914**
2S-KLC	6.1	0.445	0.471	0.858*
2S-KLCSYM	5.2	0.442*	0.471	0.877*

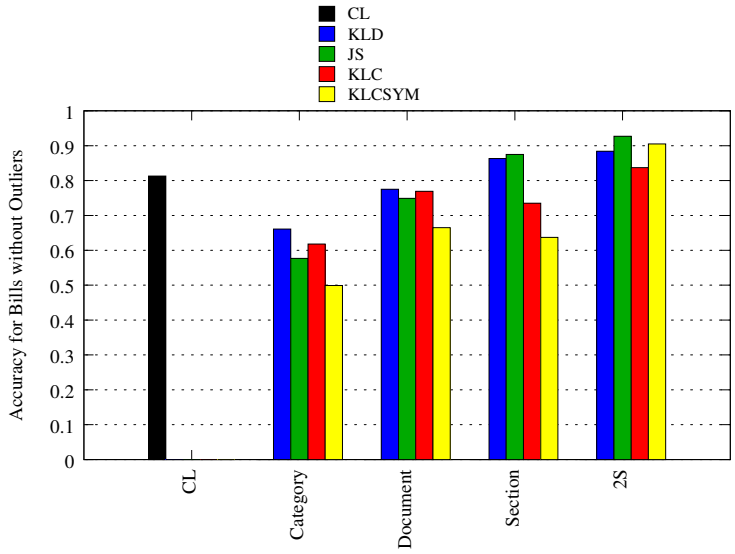

Figure 2: Accuracy for 7 Bills without Outliers. The 2S approaches do the smallest amount of mislabeling (high accuracy scores). 2S-JS is most accurate and statistically significant over all other techniques with p-value < 0.04 using the paired two-sided t-test.

method since it emphasizes differences among the compared models very strongly (half KL contribution to the average model from both sides, see Section 3.4.4), which proves to be too sensitive towards the prediction of outlier sections as it is evident in high recall and low precision and accuracy. This is poor performance according to our aims. We further analyze the differences between the other dissimilarity metrics in Section 5.

We also wanted to know how well the techniques perform when there are *no outliers in a bill*. Our annotators judged 7 bills as not containing outliers. In this case, we want the outlier detection algorithms to produce the fewest false positives. Figure 2 shows the accuracies of the techniques. A high accuracy in labeling sections correctly indicates that the techniques make fewer mistakes. We again observe in the language modeling approaches that the methods become more accurate as the unit of the language model used becomes smaller. The 2S approaches perform best here, with 2S-JS significantly doing the smallest amount of mislabeling of sections (0.92 accuracy) compared to all other techniques. The classification approach performs relatively well with 0.81 accuracy.

Apart from the train and test sets for which we have the judgments, we were also curious to understand how the techniques perform on the larger corpus. Figure 4 gives an overview of the scores assigned to sections by 2S-KLC over the full set of bills from different categories Finance, Health and Economics. The Finance category has 319 bills, Health has 766, and Economics has 123. The red line shows the distribution of the scores across all 37 categories. We were

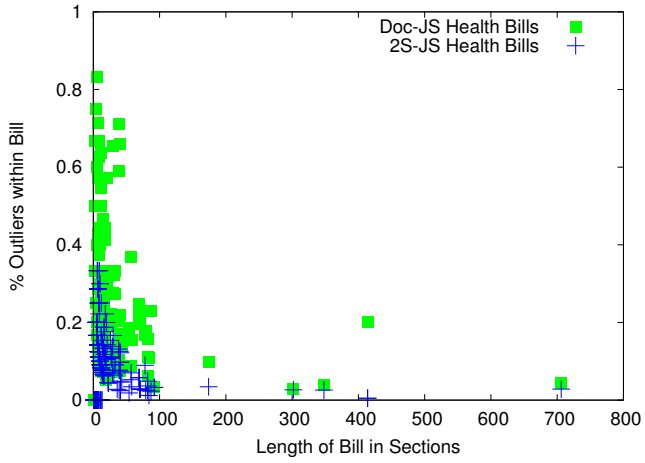

Figure 3: Fraction of sections that are predicted outliers in a bill for 2S-JS and Doc-JS for all Health bills. 2S-JS reduces the % outliers predicted within bills drastically.

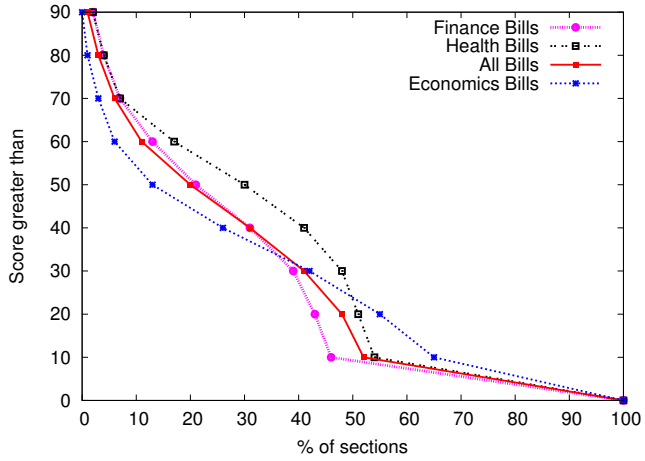

Figure 4: Overview of section scores for all bills in the categories Finance, Health, Economics, and all categories combined. Less than 10% of sections are marked as outliers.

interested in seeing how the outlier detection approaches behave across a wider set of bills without needing to judge them all. One measure that allows us to do that is '% sections predicted to be outliers' with respect to the whole dataset. In Table 2 we observed that approaches with smaller % sections predicted as outliers tend to perform better. We observe that the % sections predicted to be outliers at the outlier threshold ($y \geq 70\%$) is similar to the % sections predicted to be outliers for 2S-KLC reported in Table 2 (6.1%), since not more than about 8% of the sections are marked as outliers across these different categories.

The threshold for marking a section as an outlier is an important parameter. Intuitively, a higher threshold increases precision but hurts recall, while a lower threshold decreases precision and achieves better recall. A lower threshold also results in the % sections predicted as outliers to be increased for most of the techniques. We observed in our training set that increasing the threshold to more than around 70% lead to a large number of false negatives for all the techniques evaluated. Hence, we set the threshold to 70% for all approaches. If our aim was high recall (detecting as many outliers as possible) at the cost of precision, then using a lower threshold would be more appropriate.

We also wanted to verify the superiority of the 2-step approaches over other methods in the larger dataset. Figure 3 shows the performance of 2S-JS versus Doc-JS on all the bills from the 'Health' category in terms of the percentage of sections predicted as outliers *within single bills*: Whereas 2S-Doc marks up to 80% of (shorter) bills as outliers, 2S-JS is more discriminative with up to 30% of bills marked as outliers. Note that '% outliers within a bill' is not the same measure as '% sections predicted as outliers' in the corpus as in Table 2: in Figure 3 we look at the fraction of a bill marked as outliers, whereas the measure '% sections predicted as outliers' was referring to the whole dataset. We obtain similar results with other categories.

5. ANALYSIS OF KL DIVERGENCE CONTRIBUTION MEASURE

The results in Table 2 indicate that for the 2-step approach methods 2S-JS achieves the best performance, although it is not significantly better than 2S-KLD and 2S-KLC. In this section, we want to shed some light on the behavior of our new dissimilarity measure, KL Divergence Contribution (KLC). We do not include KLCSYM in this analysis since the results from Section 4.3 indicated that it is a poor metric.

One way to understand the behavior of KLC is to observe the effect of each term towards the section scores in a bill. Terms that have a high contribution towards each section's score are usually main topical terms (Table 1, [19]), whereas off-topic terms have low contributions. By observing the sum of each term's contribution across all the sections in a bill, we can not only identify the role of the term, but also how differently the measures weight each term. For example, the KLD measure assigns negative scores to terms if they occur more frequently in a main section than in a candidate outlier section. If this is a consistent trend for a term across most of the sections, we can identify this by noting that the sum of that term's scores is negative. Using this we examine the behavior of KLC in two scenarios that we have identified.

5.1 Scenario 1: Mild outliers

Mild outliers are more likely to have a closer distribution of topical terms to non-outlier sections while containing a limited amount of off-topic terms. If we use H.R. 2035 (the Pregnant Women Support Act) as an example, we see a bill that has a single 'mild' outlier. This bill is about pregnancy, support for new parents, and other related issues. The mild outlier in this bill is about monetary aspects of adoption assistance programs and specifically relates to changes in tax credits for the new fiscal year. The outlier is 'mild' because it is still marginally related to the main topic of the bill, yet its topic shifts to the minutiae of appropriations and spending.

In this scenario, Table 4 reveals the following: the top half shows typical off-topic outlier terms such as 'amount', 'dollar', 'taxable', 'adoption', 'inflation', and 'credit'. For these terms, 2S-KLD and 2S-KLC perform similarly, not

Term	sum of JS contributions	sum of KLC |contributions|	sum of KLD contributions
disorders	0.0007	0.001	0.0009
human	0.0007	0.01	0.002
physical	0.04	0.14	0.12
disabilities	0.02	0.07	0.07
care	0.01	0.02	0.02
...	...	...	...
park	1.23	3.06	2.29
river	1.32	3.06	1.99
management	2.09	4.11	1.09
wilderness	5.02	6.39	-3.09
land	5.04	7.71	-2.14
national	5.82	9.75	-3.97

Table 3: Sum of scores of terms for JS, KLC, and KLD across the bill for H.R. 146. KLC is most discriminative, followed by KLD. JS is too weak for this bill.

Term	sum of KLC |contributions|	sum of KLD contributions	sum of JS contributions
inflation	0.013	0.013	0.005
taxable	0.032	0.032	0.012
dollar	0.06	0.06	0.023
credit	0.072	0.035	0.035
amount	0.18	0.18	0.061
adoption	0.26	0.20	0.136
...	...	...	...
pregnant	1.09	-0.62	0.815
women	0.54	-0.48	0.478
services	0.45	-0.27	0.29
support	0.44	-0.23	0.305
assistance	0.15	-0.07	0.131
program	0.11	-0.04	0.105
parents	0.14	-0.03	0.145

Table 4: Differences in term scores across the bill between 2S-KLC, 2S-KLD, and 2S-JS for H.R. 2035.

revealing any differences in handling the outlier. The lower part of Table 4 reveals the scores for main topical terms. In comparing to the other measures, we see that 2S-KLD assigns lower scores to topical terms, while 2S-KLC and 2S-JS assign higher scores. The negative scores for 2S-KLD indicate that in this bill, across all sections, the topical terms occurred slightly more frequently in main sections (Q) than in candidate outlier sections (P), because under KL Divergence a term x with $P(x) < Q(x)$ will have a negative contribution towards the total score. This causes the scores to sufficiently even out for mild outliers. Given that 2S-KLC assigns the highest scores to topical terms, it causes the scores of more non-outlier sections to be pushed towards the outlier threshold. As a result, under 2S-KLC more non-outlier sections are marked as outliers than with 2S-KLD.

5.2 Scenario 2: Strong Outliers

Strong outliers present term distributions that are quite different from topical sections. An example of a strong outlier was mentioned earlier – the gun control section within the credit card bill H.R. 627 from the introduction. Another bill with strong outliers is H.R. 146, The Omnibus Public Land Management Act of 2009, a bill focused on matters pertaining to public lands with several health-related outliers towards the end. These sections contain off-topic terms that do not appear elsewhere in the bill – terms like 'physical', 'disabilities' and 'paralysis'. Table 3 shows the scores for off-topic terms in the top half and topical terms in the bottom half for all three methods. We see that 2S-KLC assigns much higher scores to topical terms with fairly low scores to off-topic terms. Given that a strong outlier will have a high concentration of off-topic terms and very few topical terms, KLC will result in higher contribution scores for the topical terms than it would in a non-outlier section. This is because when such an outlier section is compared to a main section that primarily uses topical terms and very few off-topic terms, then KLC will note both differences in the distributions strongly by means of the absolute value. Combined with higher scores for off-topic terms, in comparison to the other measures, this results in an overall higher section score using 2S-KLC.

From this analysis we conclude that the dissimilarity measures are sensitive with respect to mild outliers versus strong outliers. We also note in the tables that 2S-JS stands in between the two measures 2S-KLC and 2S-KLD, since its scores do not appear as high as KLC, but they do not have the negative dampening effect of KLD. This is also apparent when comparing the formulae of the dissimilarity metrics. Therefore, we conclude that depending on the data and the kind of outliers attempted to be detected, it is worth employing and comparing all these different dissimilarity metrics before making a choice.

6. CONCLUSIONS

We have presented various new ways of approaching outlier detection of sections within documents, which are particularly applicable in the absence of large amounts of truth data. We did this study within the domain of Congressional Legislation. One of the techniques we utilized is classification based, in which the algorithm detects outliers by observing the inverse popularity of a section's category together with its bill's category in the corpus. We also introduced language modeling approaches to outlier detection that differ in the unit of language models used for comparing against sections, and in the dissimilarity measures that they utilize. The dissimilarity measures are based on Kullback-Leibler Divergence. We saw that the language modeling based methods perform better when compared against classification both in terms of precision and accuracy with a smaller fraction of overall outlier sections found. The best performing methods use a 2-step approach, for which first the most topically relevant sections are detected in the bill, after which the remaining sections are compared against the

former ones. Sections with an outlier score of 70% or above are marked as outliers. We saw that these techniques perform better than the direct application of dissimilarity measures to the Category, Document, and Section models.

Finally, for the 2-step approach methods, we discovered that the dissimilarity metrics yield different results: the KL Divergence measure works better for bills that have 'mild outliers' which are marginally related to the main topic of the bill. For bills with stronger outliers however, which indicate a very unusual topic shift, KL Divergence Contribution – which we introduce in this paper – is more discriminative because it assigns greater scores to rare terms and less frequently used topical terms in outliers. Our analysis of the term scores does not give us strong indication as to why 2S-JS performs marginally better than the other measures in our evaluation. Given our analysis in Section 5 and the fact that most of the bills in our evaluation set do not contain very strong outliers, we hypothesize that 2S-JS, being a milder measure, does better in the presence of milder outliers.

As for future work, we would particularly be interested in incorporating synonyms into the model comparisons. The current language modeling techniques do a one-by-one comparison of terms, and we believe that the use of synonyms would result in more accurate discrimination between outlier sections and non-outlier sections.

One of the central challenges of this work was to achieve good outlier detection with limited data. We hope to increase the size of our truth-labeled data set in the future, which would enable us to draw further conclusions about the techniques.

7. REFERENCES

[1] M. Agyemang, K. Barker, and R. Alhajj. Framework for mining web content outliers. In *SAC '04: Proc. 2004 ACM symposium on Applied computing*, pages 590–594, New York, NY, USA, 2004. ACM.

[2] M. Agyemang, K. Barker, and R. Alhajj. Hybrid approach to web content outlier mining without query vector. In A. M. Tjoa and J. Trujillo, editors, *Data Warehousing and Knowledge Discovery*, volume 3589 of *Lecture Notes in Computer Science*, pages 285–294. Springer Berlin / Heidelberg, 2005.

[3] Y. Assogba, I. Ros, and M. McKeon. Docblocks: communication-minded visualization of topics in u.s. congressional bills. In *Proc. 28th of the international conference extended abstracts on Human factors in computing systems*, pages 4117–4122, New York, NY, USA, 2010. ACM.

[4] V. Barnett and T. Lewis. *Outliers in statistical data*. John Wiley, 1994.

[5] J. P. Callan. Passage-level evidence in document retrieval. In *Proc. 17th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '94, pages 302–310, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[6] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In *Proc. 34th annual meeting on Association for Computational Linguistics*, pages 310–318, Morristown, NJ, USA, 1996. Association for Computational Linguistics.

[7] J. Cohen. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220, 1968.

[8] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley series in Telecommunications. Wiley, New York, NY [u.a.], 1991.

[9] S. Cronen-Townsend and W. B. Croft. Quantifying query ambiguity. In *Proc. second international conference on Human Language Technology Research*, pages 104–109, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[10] S. Cronen-Townsend and W. B. Croft. Quantifying query ambiguity. In *Proc. second international conference on Human Language Technology Research*, pages 104–109, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[11] M. A. Hearst. Multi-paragraph segmentation of expository text. In *Proc. 32nd annual meeting on Association for Computational Linguistics*, ACL '94, pages 9–16, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics.

[12] M. A. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *Proc. 16th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '93, pages 59–68, New York, NY, USA, 1993. ACM.

[13] IBM Many Bills. A Visual Bill Explorer. `http://manybills.researchlabs.ibm.com/`.

[14] P. Jaeger. E-government around the world: lessons, challenges, and future directions. *Government Information Quarterly*, 20(4):389–394, 2003.

[15] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: algorithms and applications. *The VLDB Journal*, 8(3-4):237–253, 2000.

[16] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[17] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proc. 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 111–119, New York, NY, USA, 2001. ACM.

[18] G. T. Lau, S. Kerrigan, K. H. Law, and G. Wiederhold. An e-government information architecture for regulation analysis and compliance assistance. In *Proc. 6th international conference on Electronic commerce*, ICEC '04, pages 461–470, New York, NY, USA, 2004. ACM.

[19] D. J. Lawrie and W. B. Croft. Generating hierarchical summaries for web searches. In *Proc. 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 457–458, New York, NY, USA, 2003. ACM.

[20] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37:145–151, 1991.

[21] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[22] D. Metzler, S. Dumais, and C. Meek. Similarity measures for short segments of text. In G. Amati, C. Carpineto, and G. Romano, editors, *Advances in Information Retrieval*, volume 4425 of *Lecture Notes in Computer Science*, pages 16–27. Springer Berlin / Heidelberg, 2007.

[23] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proc. 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY, USA, 1998. ACM.

[24] C. Privault, J. O'Neill, V. Ciriza, and J.-M. Renders. A new tangible user interface for machine learning document review. *Artificial Intelligence and Law*, 18:459–479, 2010.

[25] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. *SIGMOD Rec.*, 29(2):427–438, 2000.

[26] N. T. Roman, C. D. Ferreira, L. A. A. Meira, R. Rezende, L. A. Digiampietri, and J. J. Filho. Attribute-value specification in customs fraud detection: a human-aided approach. In *Proc. 10th Annual International Conference on Digital Government Research*, pages 264–271. Digital Government Society of North America, 2009.

[27] 2009 US Congressional Legislation. Available at GovTrack:. `http://www.govtrack.us/data/us/111/`.

[28] J. Ward, Joe H. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):pp. 236–244, 1963.

[29] E. W. Welch and C. C. Hinnant. Internet use, transparency, and interactivity effects on trust in government. In *Proc. 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 5 - Volume 5*, pages 144.1–, Washington, DC, USA, 2003. IEEE Computer Society.

[30] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proc. tenth international conference on Information and knowledge management*, pages 403–410, New York, NY, USA, 2001. ACM.