

Multidimensional Text Analysis for eRulemaking

Namhee Kwon
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
+1-310-822-1511
nkwon@isi.edu

Stuart W. Shulman
University of Pittsburgh
121 University Place, Suite 600
Pittsburgh, PA 15260
+1-412-624-3776
shulman@pitt.edu

Eduard Hovy
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, 90292
+1-310-822-1511
hovy@isi.edu

ABSTRACT

To support rule-writers, we are developing techniques to automatically analyze large number of public comments on proposed regulations. A document is analyzed in various ways including argument structure, topics, and opinions. The individual results are integrated into a unified output. The experiments reported here were performed on comments submitted to the Environmental Protection Agency in response to their proposed rule for mercury regulation.

Categories and Subject Descriptors

I.2.7 [Natural Languages]: Text analysis

General Terms

Algorithms, Experimentation, Human Factors, Languages

Keywords

Regulations, Electronic Rulemaking, Federal Government, Text Annotation, Semantic Frame, Argument Structure

1. INTRODUCTION

Every year thousands of government personnel at over 150 federal agencies and sub-agencies collaborate with stakeholder interest groups, lobbyists, lawyers, and citizens to craft as many as 8,000 regulations. The various actors involved increasingly use information and communication technologies in what has become known as electronic rulemaking, or e-rulemaking [30].

One former Director of the Federal Register recently warned that using the Internet to increase public participation in rulemaking “has inspired a sort of rulemaking arms race” in which some groups using web-based services “are convinced their position is strengthened by taking ten salient points and masquerading them as thousands of unique thoughts from thousands of thoughtful taxpayers.”[13].

Neil Eisner noted in his dg.o2005 keynote address that electronic rulemaking is a research domain teeming with eager government partners facing mounting information management challenges. At NSF-funded workshops and in numerous focus groups and

interviews, personnel at diverse federal agencies with significant rule writing responsibilities have indicated there is a dearth of tools for sorting through the comments on which modifications to proposed rules are supposed to be based [31]. As a result, headway is starting to be made in the various information retrieval tasks related to the automated sorting of large-scale public comment datasets [23][38].

The stakes are significant. Federal agency-issued rules are arguably much more important than the many fewer laws passed each year by the U.S. Congress [9]. Officials in the Bush Administration’s Office of Management and Budget reported in 2005 that the 190 major rules promulgated during the last 24 years collectively add well over \$100 billion annually in regulatory costs to the U.S. economy [26].

In certain exceptionally controversial cases, such as the Environmental Protection Agency’s recent mercury rulemaking, organizations such as MoveOn.org (a virtual organization with no office boasting over 3 million e-mail savvy members) seized upon the open and easily accessible “notice and comment” process (mandated by the 1947 Administrative Procedure Act) to issue sometimes voluminous demands for more stringent regulations. The data used in the experiments reported here were samples drawn from a population of over 536,000 e-mail messages and an additional 4,264 comments (consisting of some “good” e-mails, web form, fax, and paper submissions) that were submitted to the EPA about the proposed mercury rule. As one legal scholar rethinking regulatory democracy noted, “though individual members of the public who write comments usually make unsophisticated statements, those messages tend to include, at their core, constructive insights relevant to agencies’ legal mandates.”[12].

Given the sheer volume of comments and the time pressure on rule-writers, we are building a system to perform various types of analysis on the comments in order to provide a rich and multidimensional preview of a body of comments, and in order to help the rule-writers manage their work efficiently. The final goal of this work is to provide collective analysis of all comments including the quantitative and distributional analysis for each opinion and subtopics, and help the rule-writers to select more informative documents by highlighting the main focus of each document.

This analysis system consists of several independent modules. Importantly, the first step is to find duplicated comments and reduce them to a single instance, and to find near-duplicate comments and extract their idiosyncratic contents. This work is being performed by our colleagues at CMU, Jamie Callan et al.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

The 7th Annual International Conference on Digital Government Research '06, May 21–24, 2006, San Diego, CA, USA.

Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

[38]. Ultimately, we intend to combine the (near-)duplicate detection and the functionalities described in this paper into a single user-interface tool, for use by rule-writers in their exploration and analysis of a large collection of comments.

In this paper, we provide multidimensional text analysis exploring various aspects of text, currently focusing on topic, structure, and opinion. We analyze the writer’s argument structure about the proposed regulation, which is composed of a main claim over the proposal and reasoning to make its claim acceptable or justified. More specifically, the main claim is classified into “in favor of” or “against” the regulation, and the topics and concerns in the argument are grouped into several interesting subtopic codes. The problem is split into four subtasks: subtopic code detection, argument structure analysis, opinion analysis, and semantic frame analysis. We define each problem as a classification task and apply a supervised machine learning method using various features.

The rest of the paper is organized as follows: Section 2 defines the sub-problems and approach, Section 3 shows our manual annotation experience, Section 4 describes our automatic system and evaluation, and Section 5 concludes.

2. Approach

To handle the huge amount of comments, each document is translated into a unified structure and then integrated into collective result. We define several individual tasks to find semantic information, and tackle each problem separately. First, we define subtopic codes and classify each information unit (sentence) into appropriate codes. Second, we find a hierarchical argument structure to give preview in different detail, and then the opinion over the regulation is detected and analyzed into assent or opposition. Next, we provide the semantic frame analysis where each frame slot is representing the semantic role in a sentence. The final step is to integrate the output of each step and provide new collective result. By combining the argument structure with the opinion analysis, we can detect the reasons of each opinion. The specific tasks are described in the following sections.

2.1 Subtopic Codes

To develop the main idea, many topics or other concerns are introduced in public comments. We categorize the information into several subtopic codes defined in Table 1. For each information unit, the appropriate codes are selected (where applicable, multiple subtopic codes are assigned). These subtopic codes were developed by a political scientist (Stuart Shulman) and sociologist (Stephen Zavestoski) working deductively on previous theory-driven empirical work with public comments submitted to federal agencies in controversial environmental rulemakings.

Table 1. Subtopic Codes

Code	Text Description
Economic	Invokes economic concepts, such as cost, burden, benefits, growth, markets, efficiency, consumers, competitiveness
Environment	Invokes environmental preservation, intrinsic values, nature
Government responsibility	Invokes the responsibility of government to protect the public interest, preserve the rule of law, create procedural equity, transparent, consistent, clear-cut standards
Health	Invokes concerns about human health
Legal	Cites a particular statute, legal proceeding, administrative rule, or court case
Policy	Calls for a particular policy, such as the maximum achievable control technology (MACT) or a cap and trade approach
Pollution	Invokes the concept of pollution, human-made threats, ecological harm
Science	Invokes science, scientists, a specific study or scientific finding
Technology	Invokes technology

2.2 Argument Structures

Our assumption about the commentary texts is that a document has an internal argument structure to make its claim acceptable or justified. There might be a question whether the text is a real “argument” from the point of logic [27], but at least the comment contains claims and other statements supporting the claim, even if it may not be logical or valid.

We define the hierarchical tree structure of information units, each is a smallest part of text containing the coherent content as a claim or reasoning of the argument. The information unit is mostly a sentence but possibly a clause or a phrase. The hierarchical structure is limited to three levels, and each unit is classified into *root*, *subroot*, and *leaf* depending on its role in the argument. *Root* is a main claim over the proposal or conclusion of the argument. Multiple *roots* are possible in a document when more than one different claims are found, while there is only one root in most documents. Since the document is a comment on the proposed rule, we choose the most proposal-related claim as *root*. Each *root* has links to supporting reasoning, which is interpreted as a set of trees, each with *subroot* and *leaves*. *Subroot* is a direct supporting reason of the *root* and *leaves* are more detailed descriptions or examples of the *subroot*. All *leaves* under the same *subroot* share one topic.

In addition, we specify the types of relations between *root* and *subroot*, and between *subroot* and *leaf*. We classify a relation into three categories: *support*, *oppose*, and *restate*. *Restate* denotes the repetitive or paraphrasing statement of the parent. For example, when there are two important *roots* in a document but both of them make the same claim, we set one as a *root* and the other as a child of the *root* with the relation of *restate*. The *support* or *oppose* link shows whether the child supports or opposes the parent. Most links are *support* because people usually use positive reasoning for their claim. However, some texts contain conflicting reasons, for example, when a person mentions the pros and cons of his claim, or gives opposite example of his claim as in the following sentence: (*T1* is linked to *T2* as a child with the relation of *oppose*.)

Although [the acid rain program that this cap and trade proposal is based on has been successful]_{T1}, [it is simply an ineffective and inappropriate way to deal with mercury emissions.]_{T2}

Figure 1 shows an example of argument structure where each information unit is specified as *root*, *subroot*, and *leaf*, and *leaves* are expressed as subtopic codes.

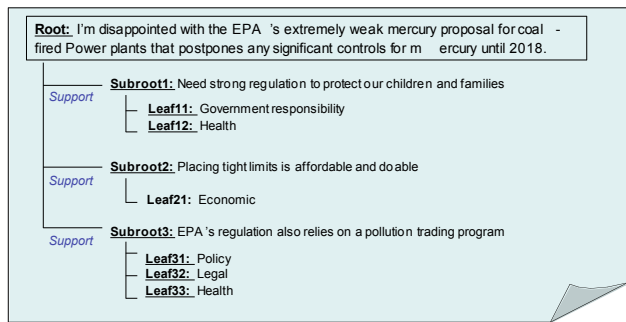


Figure 1. Example Argument Structure of a document: leaves are shown as subtopic codes.

There have been various approaches to discourse structure analysis. Mann and Thompson [24] defined tree structure of discourse units having the status of *nucleus* or *satellite* with the rhetorical relations such as circumstance, contrast, and sequence. Teufel and Moens [32] defined non-hierarchical structure of scientific articles, more like a segmentation tagged with rhetorical function (background, aim, contrast, etc.) Our argument structure is similar to the tree structure of *nucleus* and *satellite* in [24], but the structure is the abstraction of argument about the regulation, so that the *root* and *subroot* relations are defined more globally in text rather than nested correlations between each discourse unit. In other words, finding *root* and *subroot* is to extract the claim and main reason in the document, which is comparable to extracting the “thesis statement” in [6].

2.3 Opinions

The public comments in our domain express the opinion or stance over the proposed rule. Our goal is to classify the comments if they are in favor of, or against the proposal, which is similar to finding the polarity (positive or negative) in previous research. Much work has addressed the problem of analyzing opinions from texts, including detecting subjectivity [35][39], classifying semantic orientation (polarity) of words [32], phrases [37], sentences [19][39], or documents [28]. Most approaches are based on lexical subjectivity and find dominant (more frequent or stronger) positive or negative expressions.

However, finding all positive and negative expressions is not enough. The example text (A) in Table 2 shows supportive (positive) expressions but actually opposes the rule because it compliments the alternative of the rule in question. Recognizing such cases requires topic analysis in addition to polarity analysis. Often, the topic is not available locally (expressed in a short phrase); instead, the specific contents of the rule or the alternative are described.

Further, one document possibly covers multiple topics and multiple opinions over the specific issues in the regulation (see the example (B) in Table 2). We attempt to detect every different

opinion and to find corresponding reasons for each. Given the argument structure described in Section 2.2, we can assume that the main claim (opinion) about the proposed rule is in *root*. When there are different opinions in a document, each would be a separate *root*, and we determine the polarity of the *root* instead of whole text.

The opinion is classified into positive or negative attitude to the regulation. However, some texts suggest an alternative to the proposal or outside issues as in the example (C) in Table 2, so we define the opinion into three categories: *support the regulation*, *oppose the regulation*, or *propose a new idea*. The rule-makers may want to refer to the original full text when it is classified to *propose a new idea*.

Table 2. Opinion from Comments

(A)	I support the recommendation by EPA staff scientists that both long- and short-term standards for fine particles need to be strengthened because scientific studies show serious health effects -- even death -- can occur at concentrations below the current standards.
(B)	The previous use of cap and trade methods for SOX removal is a good idea, and will work under the new mercury regulations. <u>I support the cap and trade method</u> , even if it may produce hot spots where more mercury settles... The current plan proposed by President Bush lacks toughness when preventing mercury to be emitted into our air. The regulations set to 30% and 70% reduction by 2010 and 2018, respectively, are <u>too lenient</u> on power plants. Since the maximum available control technology can reach 70 to 90% mercury removal from stack gases if the removal is done efficiently, the regulations should be stricter.
(C)	We request that you extend the comment period either until June 30, or until 30 days after the completion and public availability of any new analysis, whichever is later.

2.4 Semantic Frames

We perform the sentence structure analysis based on frame semantics [15], expecting to capture semantically important part in a sentence. Although there are several approaches on the semantic structure and available corpus, we adopt FrameNet frames since FrameNet uses self-explanatory frame and role names. Each sentence is analyzed as one or more frame(s) composed of a main predicate and associated roles.

The FrameNet project [2] defines the frames that organize the semantic information in a sentence. Each frame consists of the target predicate and a set of slots (frame elements) with corresponding semantic roles. FrameNet release 1.2 (June 2005) defines 608 frames with the 7,389 predicates. Figure 2 shows a frame example from the FrameNet.

Frame: Taking_sides

A *Cognizer* has a relatively fixed positive or negative point of view toward an *Issue* (or a *Side* in a debate concerning an *Issue*).

Example Sentence:

In interviews, it seems like everyone is completely *against* this expenditure.



Figure 2. Frame Example from FrameNet

In our work, we perform frame analysis for all verbs in a sentence. A sentence is analyzed in several ways for each verb occurrence.

3. Data Preparation

In order to train an automatic classification and recognition system, we require appropriately annotated material. A manual annotation scheme was developed in collaboration between USC-ISI and the University of Pittsburgh. It was deployed through an iterative, trial and error process using coders working in Pitt’s Qualitative Data Analysis Program (QDAP) during the summer and fall of 2005. We asked the annotation of the same document to at least two coders in parallel. We compute the inter-human agreement by Cohen’s kappa coefficient [10] and F-measure. Cohen’s kappa is an agreement score considering chance agreement, and F-measure is a traditional method to average precision and recall. We compute precision and recall assuming one coder’s annotation is a real answer, hence F-measure means the ratio of matches over the other’s annotation.

Initially, the QDAP coders employed ATLAS.ti¹, a commercial off-the-shelf qualitative data analysis application. The QDAP coders’ existing familiarity with the application of subtopic codes to mercury rulemaking public comment texts using ATLAS.ti enabled the achievement of reasonably high levels of inter-rater reliability (Cohen’s kappa coefficient is 0.7 and F-measure is 0.67). However, this annotation scheme was not satisfactory in its ability to reliably capture the linkages that define the argument structure. While ATLAS.ti does allow coders to assign hyperlinks to text segments, it was cumbersome and hence unreliable as a means for building the kinds of complex links necessary for the automatic system.

In order to capture the argument structure of the comments, we developed a java-based annotation tool (“ERuleClient”) through which QDAP coders could more easily annotate both the subtopic codes and argument linkage structures within a given comment. One of the challenges in the coding lab involved moving from a familiar proprietary coding system to a custom built system. Since the ERuleClient was ergonomically quite different for coders who had become fluent in coding using ATLAS.ti. A comparison of the coders’ annotation from the first iteration with the new ERuleClient to the previous coding in Atlas.ti shows a drop in the reliability of the application of the subtopic codes (Cohen’s kappa is 0.51 and F-measure is 0.63).

Table 3. Annotation on long documents

Code	Human1	Human2	Match	Kappa	F-measure
------	--------	--------	-------	-------	-----------

¹ <http://www.atlasti.com>

Gov. Resp.	4	8	3	0.50	0.50
Economic	36	64	29	0.54	0.58
Legal	10	14	6	0.49	0.50
Health	49	52	44	0.86	0.87
Science	35	118	20	0.18	0.26
Policy	194	158	125	0.58	0.71
Technology	59	79	39	0.51	0.57
Environment	52	50	22	0.37	0.43
Pollution	191	121	97	0.48	0.62
Total	630	664	385	0.50	0.60
Arg. Struct.	Human1	Human2	Match	F-meas.	F-measure w/ type
Root (Restate)	19 (4)	11 (2)	9	0.60	0.53
Subroot	56	52	20	0.37	0.37
Leaf	145	234	90	0.47	0.47

Table 4. Annotation on short emails

Code	Human1	Human2	Match	Kappa	F-measure
Gov. Resp.	17	25	13	0.57	0.62
Economic	19	24	17	0.76	0.79
Legal	7	14	7	0.65	0.67
Health	49	64	48	0.78	0.85
Science	13	19	10	0.59	0.62
Policy	57	48	42	0.71	0.80
Technology	17	17	16	0.93	0.94
Environment	27	43	23	0.57	0.66
Pollution	85	99	80	0.72	0.87
Total	291	353	256	0.70	0.80
Arg. Struct.	Human1	Human2	Match	F-meas.	F-measure w/ type
Root (Restate)	15 (6)	17 (4)	15	0.94	0.94
Subroot	33	32	24	0.74	0.74
Leaf	31	30	23	0.75	0.75

In subsequent rounds of testing using the ERuleClient tool, we made updates on the fly which significantly improved the coders’ satisfaction with the tool. Among the many changes was the addition of a visualization option to allow coders to review the argument trees that they were creating using the tool. Despite the advances in the software, the agreement scores between coders did not increase. It was likely attributable to the length of texts to be annotated. As we selected longer texts (average 107 sentences per text) for review, in hopes of capturing more interesting argument structures, the task itself became more complex. Table 3 shows the agreement on subtopic code and argument structure. For the argument structure agreement, we consider the restating *root* or *subroot* (child linked with the relation of “restate”) as same as the *root* (or *subroot*). Among the children of the *roots* showing agreement, *subroot* and *leaf* agreement is computed. “F-measure with type” denotes the agreement not only on the role in the structure (“root”, “subroot”, and “leaf”) but also on the link type (“support the regulation”, “oppose the regulation”, and

“propose a new idea” for *root*; “support” and “oppose” for *subroot* and *leaf*).

In the most recent round of testing, the PIs selected email texts (average 10 sentences per text). Table 4 shows significantly improved scores for both subtopic code application and argument structure annotation.

Since we require enough data for training and testing, we use the data from each round fully or partially. The specific usage is described in Section 4.

4. System

For each module described in Section 2, the individual system was implemented using the data in Section 3.

4.1 Subtopic Codes

4.1.1 Implementation

To categorize the subtopic into predefined codes, we interpreted a problem as a classification of yes or no, for each subtopic code. Since we assumed that the codes were independent of each other, we performed each separate classifier and assigned all the subtopic codes (from none to multiple codes) given a sentence.

We built a classifier using a support vector machine (SVM) [34]. SVM is a machine learning method widely used in classification problems showing sound performance in many applications. It finds a hyper-plane that separates the positive and negative training examples with a maximum margin in the vector space.

We used annotated texts from every round: 118 documents as a training set and 22 documents as a test set. The training set of 118 documents was annotated by one or more annotators and the total instances of annotations were 274 documents. We used all 274 instances to obtain enough training data expecting that we could assign more confidence on features when they showed agreement between annotators.

The SVM-Light² implementation was adopted using the following semantically oriented features:

Lexeme of word
Bigram including stopwords
Bigram excluding stopwords
Named entity obtained by BBN Identifier ³ [5]
Named entity label obtained by BBN Identifier
Synonyms of the first sense of word in the WordNet 2.0 [14]

4.1.2 Evaluation

Table 5 shows the system performance, agreement with the human annotation, for subtopic code detection. Each code showed different performance: for example, “Government Responsibility” was not common for considering its broad definition, so it ended in low agreement, but “Technology” or “Health” achieved high agreement. The overall agreement is comparable to the human agreement in Table 3 and Table 4.

Table 5. System Performance on Subtopic Code Detection

Code	Human	System	Match	Kappa	F-measure
Gov. Resp.	17	14	4	0.23	0.26
Economic	38	25	18	0.54	0.57
Legal	19	11	7	0.45	0.47
Health	80	106	73	0.73	0.78
Science	46	33	21	0.49	0.53
Policy	133	148	102	0.60	0.73
Technology	26	20	19	0.82	0.83
Environment	40	48	17	0.32	0.30
Pollution	119	116	76	0.52	0.65
Total	518	521	337	0.52	0.65

4.2 Argument Structures and Opinions

For a hierarchical argument structure, first we extracted *root* by a classification method, second, segmented a document and selected a *subroot* from each segment. Next, we defined the linkage and relationship of *support*, *oppose*, and *restate* and assigned the final opinion on the root with one of *support the regulation*, *oppose the regulation*, and *propose a new idea*.

4.2.1 Main Root Identification

Root is the most important part of text containing the writer’s main claim over the regulation. Since multiple *roots* can exist, we defined the problem as a classification into root or not, given all sentences in a document. We used 105 documents annotated by multiple coders (in total, 173 documents of 8,464 sentences) for training. The SVM classifier was applied using the following features:

Word: Words in a sentence excluding stopwords.

Bigram: All pairs of consecutive words in a sentence.

Word’s Stem: Stem words obtained by Porter’s stemmer⁴.

Word frequency in the Summarized Proposal: Based on the assumption that people mention the proposed rule more in the main root, we computed the frequency ratio for each word from the proposal summary. We only considered verb, noun, adjective, and adverb.

Subjectivity: The subjective sentences tend to be a *root* sentence in commentary texts while other objective sentences are supporting reasoning of the *root*. We obtained manually annotated corpus for opinions or emotions, Multi-Perspective Question Answering Corpus⁵ (version 1.1), described in [36]. We extracted all words appearing in the subjective and objective sentences respectively, and applied a Naïve Bayes classifier [25] to compute the subjectivity score for a sentence as follows:

$$CM = |\log(p(\text{subjective}) + \sum_{i=1}^n \log(p(w_i | \text{subjective})) - (\log(p(\text{objective})) + \sum_{i=1}^n \log(p(w_i | \text{objective}))))|$$

² <http://svmlight.joachims.org/>

³ <http://www.bbn.com>

⁴ <http://www.tartarus.org/~martin/PorterStemmer/>

⁵ <http://www.cs.pitt.edu/~wiebe/pubs/pub1.html>

where $p(\text{subjective})$ is a probability of subjective sentences, $p(w_i|\text{subjective})$ is a probability of the i^{th} word's occurrences of total n words in subjective sentences, and same as for *objective*.

Position: Especially in well-written texts, the *root* sentence is highly related to the position in text. We indicated a position with three values: paragraph position, sentence position in a paragraph, and relative sentence position in a paragraph.

- *Paragraph position:* The position of a paragraph including a given sentence that is defined as the order of the paragraph in text.

- *Sentence position in a paragraph:* The number representing the order of the sentence in a paragraph.

- *Relative sentence position in a paragraph:* Since the paragraph size is different, the sentence position is represented as a relative position in a paragraph scaled to the interval [0,1].

Cue Phrase: Several cue phrases were utilized.

Please | The EPA should | You should | I hope you | I hope that you | I suggest | Do | In conclusion | In summary | I (we) support | I (we) oppose | I (we) request | I (we) urge | I (we) encourage |

Subtopic code "Policy": The subtopic code output from Section 4.1 was adopted as a feature. 11% of sentences whose topic was "policy" were *roots* in the training set, and the binary value was used to signal if the sentence covered the subtopic "policy".

Named Entity: Named Entity (organization, person, and location) recognized by BBN Identifier was used.

4.2.2 Subroot Identification

Subroot is similar to *root*, in terms that it expresses more important and informative material. We simplified the task, assuming that text was a sequence of subtopic segments. We segmented text into several subtopic groups and selected the most important sentence from each segment.

The subtopic segmentation was performed using Hearst's TextTiling [17], which utilized lexical co-occurrence and distribution. To obtain more concentrated and concrete subtopic group, we used a smaller token-sequence size 6 than the default size 20 when computing the similarity between adjacent groups of token-sequences.

To define a single important sentence from a segment obtained, we applied SVM ranker [18] to each segment. We used the same training set as in Section 4.2.1, but extracted the subtree of *subroot* and *leaves* (2,654 sentences). We compared the score within a segment, and selected the one ranked as highest of all sentences in the segment.

The same features in the root identification task in Section 4.2.1 were applied but the position features were altered slightly. Instead of position within a whole text, the relative position within a segment was used.

4.2.3 Link and Link type Identification

After obtaining *root* and *subroot*, we linked all the other topic units (topic-assigned sentences) in the segment to the *subroot*, and linked all the *subroots* to the *root*. When there were more than one *roots* in text, we chose the most semantically similar and locally closest root.

The similarity between topics was obtained by computing cosine similarity. The cosine similarity metric is widely used in information retrieval [29] between documents, but we computed it between two target sentences. The similarity between sentence S_1 and S_2 was defined as follows:

$$Sim(S_1, S_2) = \frac{\sum_i w_{i,S_1} w_{i,S_2}}{\sqrt{\sum_i w_{i,S_1}^2} \sqrt{\sum_i w_{i,S_2}^2}}$$

$$w_{i,S} = tf_{i,S} \log\left(\frac{N}{sf_i}\right)$$

where $w_{i,S}$ is a weight of term i in sentence S . The term weight is defined as a *tfidf* for each word and bigram. Since this is about the similarity between sentences, *idf* is replaced by "inverse sentence frequency" that counting the frequency in each sentence. $tf_{i,S}$ is frequency of the term i in s , and sf_i is the number of sentences containing the term i , and N is the total number of sentences in text.

To assign a *linktype* (support, oppose, or restate) to each link, we searched "restate" and "oppose" while setting "support" as the default type, since the writers in our domain did not mention contradicting issues a lot.

Restate Link: We define the relation "restate" to signal restating or paraphrasing the same contents. As described above, we computed the cosine similarity between parent and child sentences, and assigned "restate" when the similarity was larger than the empirically found threshold (similarity > 0.15).

Other than between parent and child, the similarity was also computed between pairs of *roots*, and if they were similar, then one *root* (having lower probability for a *root*) was linked to the other *root* as a child using the *linktype* "restate".

Oppose Link: At this point, only simple cue phrases were used. When two topics were not similar at all and the child contains "even if, even though, although", we assigned it as "oppose".

4.2.4 Root type (opinion) Classification

We classify the opinion into three categories (*support the regulation*, *oppose the regulation*, and *propose a new idea*) based on the content of *root* sentences. As most previous research, the positive and negative expressions were checked but we considered semantic units rather than words within a fixed size window or a syntactic clause or phrase.

Each sentence was analyzed into a list of frame elements described in Section 4.3, and "Topic score" and "Polarity score" were computed for each frame element. "Topic score" was defined as a measure of relatedness to the given proposal and "Polarity score" as a measure of polarity of positive and negative expressions as follows:

* *Topic Score*: The sum of each word’s frequency in the proposed rule summary.

* *Polarity Score*: From the opinion annotated corpus [36], positive and negative expressions were extracted, and naïve bayes classifier was built with stem words of polar expressions.

$$CM = |\log(p(\text{positive})) + \sum_{i=1}^n \log(p(w_i | \text{positive})) - (\log(p(\text{negative})) + \sum_{i=1}^n \log(p(w_i | \text{negative})))|$$

Based on these scores, the final *roottype* (opinion) was determined by simple heuristically derived rules. The detailed procedure is explained in the following:

Given a root sentence,

1) Identify frame elements described in Section 4.3.2

2) For each Frame Element (FE):

- build a 2-tuple (P, T) where P is polarity score and T is topic score

- Sum tuples into two categories: polarity for topic (topic, polarity score) and polarity for something else (other, polarity score)

3) For a main predicate verb of the sentence:

compute polarity score

4) determine the final *roottype* by the following rules

Predicate (negative) => oppose

Predicate (positive) + FE (topic) => support

Predicate (Neutral) + FE (topic, positive) => support

Predicate (Neutral) + FE (topic, negative) => oppose

Predicate (Neutral) + FE (other, positive) => propose

Predicate (Neutral) + FE (other, negative) => oppose

4.2.5 Evaluation

To evaluate the argument structure, first, we compared the *root* with the human annotated *root*, second, checked the *subroots* given the agreed *root*. Since the roots having the same claim were linked with “restate”, if either of *root* or *restating root* matched then we considered it as agreement. Table 6 shows the performance of our system on the argument structure. When we consider the low agreement on the argument structure in long documents (Table 3), the system performance is encouraging although we believe there is space for improvement.

Table 6. System Performance on Argument structure

Type	Human	System	Match	F-measure
Root (restate)	33(7)	22(4)	15	0.55
Subroot	29	45	24	0.65

The agreement in *link type* and *root type* is highly restricted to the *root* and *subroot* agreement since the *link type* is determined for the given parent-child link in the previous step. When the system agreed on the parent-child relations with the human annotation, the system showed the perfect human-system agreement on the *linktype*, where all links had “support” relations (Note that we already considered “restate” relation for the argument element detection of *root* and *subroot*). Since human coders rarely found

“oppose” link in text and they did not agree on the parent-child relations in those cases, it was hard to evaluate the proper agreement on *linktype* between humans and with system, rather we could conclude most links “support” the parent in our domain.

Table 7 shows the accuracy on *roottype* classification. As a baseline to compare, we computed only “polarity score” for a sentence and determined the type (“support the regulation” or “oppose the regulation”). “3-type classification” includes “propose a new idea” which generated more disagreement in human annotation.

Table 7. System Performance on Root Type Classification

Task	Baseline	System
3-type classification	N/A	0.60
2-type classification	0.42	0.77

4.3 Semantic Frames

Focusing on the frames of verbs, we selected *main verbs* from a sentence parsed with the Charniak parser⁶ and chose all verbs having a path of *S-VP+-VB** from a root node *S*⁷. For example, in the text “The present administration has shown inadequate determination to maintain present standards, or to raise them where justified by cost and benefit analysis.”, we extracted “shown”, “maintain”, “raise”, and “justified” and provided the frame structure for all four predicates.

We selected 120 verb lists from our training data (used in subtopic code classifier in Section 4.1), and searched FrameNet to find 98 corresponding frames including 191 roles. All predicate targets associated with these frames were extracted for training and testing. We obtained 37,764 annotated sentences of the training set, 3,870 sentences of the development set, and 4,745 sentences of the test set.

A shallow semantic parser was implemented based on [21] and [22] using Maximum Entropy models [4]. Given a sentence with a predicate, the frame name was assigned first, and then frame elements were identified and appropriate roles were found for each element.

4.3.1 Frame Classification

The frame classifier was implemented with three feature sets: *lexical unit* (lexeme of word), *lexical type* (verb, noun, or adjective) of the predicate target, and *subcategorization*. *Subcategorization* is defined as a parsing rule that expands the VP (verb phrase) of the predicate verb.

4.3.2 Frame Element Identification

To find the frame element of a sentence, we classified each constituent from a parse tree as being a Frame Element or not. A MaxEnt classifier was implemented using many syntactic and semantic features, and most features were adopted from [22] and [3] (shown in Table 8).

⁶ <http://www.cs.brown.edu/people/#software>

⁷ These are POS (Part Of Speech) tags defined in Penn Treebank (<http://www.cis.upenn.edu/~treebank/>). S is for sentence, VP is for verb phrase, and VB* is for all verb forms including VB, VBD, VBG, VBN, VBP, and VBZ.

Table 8. Sets for Frame Element Identification

<i>Target:</i> Target word
<i>Lexunit:</i> Lexeme of target word + Target type (verb, noun, adjective)
<i>Path:</i> Path from constituent to target word in syntactic parse tree
<i>S Path:</i> Path from constituent to S of target word
<i>Head:</i> Head word of constituent
<i>Phrase Type:</i> Phrase type of constituent in parse tree (ex. NP, VP)
<i>Logical Function:</i> Governing phrase type (S or VP) of NP
<i>Position:</i> Relative position of constituent to the target word
<i>Voice:</i> Active or Passive voice of target phrase
<i>First Word:</i> First word of constituent
<i>First POS:</i> Part of Speech tag of first word of constituent
<i>Last Word:</i> Last word of constituent
<i>Last POS:</i> Part of Speech tag of last word of constituent
<i>Left Head:</i> Headword of left sibling constituent
<i>Right Head:</i> Headword of right sibling constituent
<i>Named Entity:</i> Named Entity tag of constituent
<i>Head POS:</i> Part of Speech of headword of constituent
<i>Partial Path:</i> Path when constituent is under the same “S” in parse tree
<i>S Count:</i> Number of “S” tags from constituent to target in parse tree
<i>Subcategorization:</i> List of constituent labels under the VP of target

4.3.3 Role Labeling

With identified frame elements, non-overlapping frame element lists were constructed by selecting constituents having higher probability as a frame element when there was overlap between identified frame elements. Role tagging was performed with features including sentence-wide features, and all feature sets are described in Table 9.

Table 9. Feature Sets for Role Tagging

<i>Target:</i> Target word
<i>Lexunit:</i> Lexeme of target word + Target type (verb, noun, adjective)
<i>Head:</i> Head word of constituent
<i>Phrase type:</i> Phrase type of constituent in parse tree (ex. NP, VP)
<i>Logical function:</i> Governing phrase type (S or VP) of NP
<i>Position:</i> Relative position of constituent to the target word
<i>Voice:</i> Active or Passive voice of target phrase
<i>Order:</i> Relative position of frame element in a sentence
<i>Syntactic Pattern:</i> Pattern generated from target word, phrase type, and logical function in a sentence
<i>Previous Class:</i> Role of n th previous constituent

4.3.4 Evaluation

Evaluation on frame analysis was performed on a held-out test set from FrameNet as well as a test set of eRulemaking comments. The results are shown in Table 10 and Table 11. Because of longer and more complicated sentence structures in our domain data, which are different from the structures from FrameNet, the performance dropped by 6% in frame element identification and tagging.

Table 10. Evaluation on test set from FrameNet

Process	Prec.	Recall	F-score
Frame Classification	Accuracy: 0.94		

Frame Element (FE) Identification	0.92	0.73	0.81
Role Tagging given FE	0.84	0.81	0.82
FE identification + Role Tagging	0.75	0.60	0.66

Table 11. Evaluation on test set from eRulemaking data

Process	Prec.	Recall	F-score
Frame Classification	Accuracy: .77		
FE identification + Role Tagging	0.67	0.55	0.60

4.4 Integration

The individual output of each module is integrated to a collective result for all comments. Figure 3 shows the combined result of the test set, and it is an example of the summarized output of multi-texts, which will be provided to rule-writers. All topics were summed up for each opinion category as supporting reasons. When the *linktype* is “oppose”, the topic is added as opposite of the root’s opinion, for example, if the *roottype* is “oppose the regulation” and the child topic is linked with “oppose”, then the topic is summed to “support the regulation”.

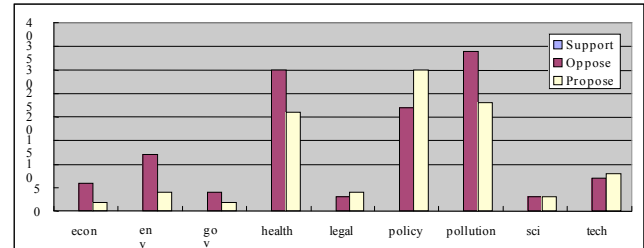


Figure 3. Integrated Output of subtopic code, argument structure, and opinion for test set

<p>Frame: <i>Request</i> - ask, demand, urge</p> <p><i>Speaker</i> [I, We, the Center, the Council]</p> <p><i>Addressee</i> [EPA, you]</p> <p><i>Message</i></p> <p>[to withdraw its rulemaking at the present time to implement strong controls on mercury emissions from coal-fired plants return to prior analyses and reduce the SOX cap to 2 million by 2009]</p>
<p>Frame: <i>Removing</i> - eliminate, evict, remove, take, withdraw</p> <p><i>Agent</i> [EPA, the Center]</p> <p><i>Source</i> [off the 112 (c) list, its rulemaking]</p> <p><i>Theme</i> [Mercury Pollution, power plants, the annual testing requirement]</p>
<p>Frame: <i>Reasoning</i> - demonstrate, prove, show</p> <p><i>Arguer</i> [EPA, EPA’s own analysis, Science, a comprehensive study]</p> <p><i>Content</i></p> <p>[the present administration that such standards were “appropriate and necessary” that there is questionable basis to regulate mercury emissions from power plants]</p>

Figure 4. Excerpts from Integrated Frame Output for test set

To provide a numerical indication showing agreement in the final graphical output, we computed the cosine similarity between human and system. The term weights were defined as topic frequency per file, subtopic code, and *roottype*. The value

between human and system was 0.48, compared to the value 0.67 between two humans.

Figure 4 shows the part of summed output of frame analysis. This shows semantic frames including semantic roles and real instances found in the comments.

5. Conclusion

We have described our system to extract various aspects of information from texts including the annotation process. We are planning to investigate a way to improve the individual step by defining the codes and the manual annotation task more clearly and by using more generalized pattern-based features. For a prototype system to be provided to rule-writers, we will conduct more analyses of the trends and new aspects of future public comment. Further, we plan to combine this work with that of our colleagues at CMU on near-duplicate detection and to create a system that performs multi-aspect analysis of rulemaking comments and provides a useful review tool for rule-writers.

6. ACKNOWLEDGMENTS

The researchers wish to acknowledge the EPA for providing the datasets on which this report is based. This work was supported by NSF grants IIS-0429293 and IIS-0429360.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the National Science Foundation.

7. REFERENCES

- [1] Altman, D., *Practical Statistics for Medical Research*. Chapman and Hall, 1991.
- [2] Baker, C.F., Fillmore, C.J., and Lowe, J.B., The Berkeley FrameNet project. In *Proceedings of COLING-ACL*, Montreal, Canada, 1998.
- [3] Bejan, C.A., Moschitti, A., Morarescu, P., Nicolae, G., and Harabagiu S., Semantic Parsing based on FrameNet. In *Proceedings of ACL-SENSEVAL workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, Spain, 2004.
- [4] Berger, A., Pietra D., and Pietra, V.D., A Maximum Entropy Approach to Natural Language. *Computational Linguistics*, 22(1), 1996.
- [5] Bikel, D., Schwartz R., and Weischedel, R. M., An Algorithm that Learns What's in a Name. *Machine Learning*, 34 (1-3), pp. 211--231. 1999.
- [6] Burstein, J., Marcu, D., Andreyev, S., and Chodorow, M., Towards automatic classification of discourse elements in essays. In *Proceedings of the 39th annual Meeting on Association for Computational Linguistics*, Toulouse, France, 2001.
- [7] Brill, E., Some Advances in Transformation-Based Part of Speech Tagging, In *Proceeding of the 12th National Conference on Artificial Intelligence*, Seattle, WA. 1994.
- [8] Coglianese, C. The Internet and Citizen Participation in Rulemaking. *I/S* 1(1): 33-57. 2005.
- [9] Coglianese, C. E-Rulemaking: Information Technology and the Regulatory Process. *Administrative Law Review* 56(2): 353-402. 2004.
- [10] Cohen, J., A coefficient of agreement for nominal scales. *Education and Psychological Measurement*, 43(6):37—46. 1960.
- [11] Craggs, R. and Wood, M., Evaluating Discourse and Dialogue Coding Schemes. *Computational Linguistics*, 31(3),pp. 289-295, 2005.
- [12] Cuéllar, M.F., Rethinking Regulatory Democracy. *Administrative Law Review* 57(2): 411-499. 2005.
- [13] Emery, F. Emery, A., A Modest Proposal: Improve E-Rulemaking by Improving Comments. *Administrative and Regulatory Law News*, 31(1): 8-9. 2005.
- [14] Fellbaum, C., *An Electronic Lexical Database*, The MIT press. 1998.
- [15] Fillmore, C.J., Frame Semantics and the Nature of Language. *Annals of the New York Academy of Science Conference on the Origin and Development of Language and Speech*, 280: 20-32. 1976.
- [16] Fleiss, J., *Statistical Methods for Rates and Proportions*. John Wiley & Sons, 1981.
- [17] Hearst, M., TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages. *Computational Linguistics*, 23 (1), pp. 33--64. 1997.
- [18] Joachims, T., Optimizing Search Engines Using Clickthrough Data, In *Proceeding of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, ACM, Edmonton, Alberta, Canada, 2002.
- [19] Kim, S. and Hovy, E., Determining the Sentiment of Opinions. In *Proceedings of COLING*, Geneva, Switzerland, 2004.
- [20] Krippendorff, K., *Content Analysis: An Introduction to Its Methodology*. 2nd ed. Sage, Beverly Hills, CA. 2004.
- [21] Kwon, N., Fleischman, M.B. and Hovy, E., FrameNet-based Semantic Parsing using Maximum Entropy Models. In *Proceedings of COLING-04*, Geneva, Switzerland. 2004.
- [22] Kwon, N., Fleischman, M., and Hovy, E., SENSEVAL Automatic Labeling of Semantic Roles Using Maximum Entropy Models. In *Proceedings of ACL-SENSEVAL workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, Spain, 2004.
- [23] Lau, G.T., Law, K.H., and Wiederhold, G., A Relatedness Analysis Tool for Comparing Drafted Regulations and Associated Public Comments. *I/S* 1(1): 95-110. 2005.
- [24] Mann, W.C. and Thompson, S., Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(2):243-281. 1988.
- [25] Mitchell, T., *Machine Learning*, McGraw-Hill. 1997.
- [26] Office of Management and Budget, Bush Administration Cuts Regulatory Cost Growth by 70%, Press Release available at: <http://snipurl.com/kofg>. 2005.
- [27] Possin, K., *Critical Thinking*. The Critical Thinking Lab. 2002.

- [28] Pang, B., Lee L., and Vaithayanathan, S., Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of EMNLP*, Philadelphia, PA. 2002.
- [29] Salton, G., Wong, A., Yang, C.S., A Vector Space Model for information Retrieval. *Communications of the ACM*. 18(11):613-620. 1975.
- [30] Shulman, S.W., E-Rulemaking: Issues in Current Research and Practice. *International Journal of Public Administration* 28: 621-641. 2005.
- [31] Shulman, S.W., The Internet Still Might (But Probably Won't) Change Everything. *I/S* 1(1): 111-145. 2005.
- [32] Teufel, S. and Moens, M., Discourse-level Argumentation in Scientific Articles: Human and Automatic Annotation. In *Proceedings of ACL workshop on Towards Standards and Tools for Discourse Tagging*, College Park, MD. 1999.
- [33] Turney, P., and Littman, M., Measuring Praise and Criticism: Inference of Semantic Orientation from Association. *ACM Transactions of Information Systems (TOIS)* 21(4):315-346. 2003.
- [34] Vapnik, V. N., *The nature of Statistical Learning Theory*, Springer. 1995.
- [35] Wiebe, J., Wilson, T., Bruce, R., Bell, M., and Martin, M., Learning Subjective Language. *Computational Linguistics* 30(3):277-308. 2004.
- [36] Wilson, T., Wiebe, J., Annotating Opinions in World Press. In *Proceedings of SIGdial-03*. Sapporo, Japan, 2003.
- [37] Wilson, T., Wiebe, J., and Hoffmann, P., Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of HLT-EMNLP*, Vancouver, Canada. 2005.
- [38] Yang, H. and Callan, J., Near Duplicate Detection for eRulemaking. In *Proceedings of the Sixth National Conference on Digital Government Research*, Atlanta, GA. 2005.
- [39] Yu, H. and Hatzivassiloglou, V., Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In *Proceedings of EMNLP-03*, Sapporo, Japan, 2003.