

Information Acquisition using Multiple Classifications

Namhee Kwon and Eduard Hovy

University of Southern California
Information Sciences Institute
Marina del Rey, CA, USA
{nkwon,hovy}@isi.edu

ABSTRACT

In order to obtain a coherent overall picture of a large collection of documents, we often need to extract various aspects of information and integrate them. Especially for subjective documents addressing a single topic, traditional summarization techniques are inadequate for differentiating and clustering similar information. We perform multiple classifications to handle diverse aspects, including subtopic identification, keyword extraction, argument structure analysis, and opinion classification, in order to provide a summarized overview of the collection, complete with distributional information. From this overall summary, system users can effectively obtain more fine-grained information. Our methods for individual modules significantly outperform the baseline and achieve human-level agreement.

Categories and Subject Descriptors

I.2.7 Natural Language Processing – *Text analysis*

General Terms

Algorithms.

Keywords

Classification, Topic identification, Keyword extraction, Argument structure, Opinion classification.

INTRODUCTION

There is substantial need to understand and manage a large collection of documents today, requiring an ability to extract the “important” information and to structure it into a coherent summary. Many traditional summarization techniques extract important information by counting frequency of occurrence, identifying redundancies, and clustering similar contents. However, in some applications, especially for subjective documents on a contentious topic, people may be more interested in uncommon or unique opinions, and clustering may not work since two documents may share many words, yet argue or claim different opinion.

The following example, from public responses to a pro-

posed regulation change in the government, illustrates that lexical analysis is not enough to cluster similar contents. Though the sentences are lexically quite similar, they actually have opposite meanings:

- *I support the strict controls on mercury pollution from power plants.*
- *I oppose the implementation of strict controls on mercury pollution from power plants.*

Further, the following two sentences have common issues, but one has to check the relations between two concepts *hot spots* and *cap-and-trade program*:

- *I am also concerned that a cap and trade program will result in hot spots.*
- *Although hot spots were a concern then, cap-and-trade in this situation would be ideal.*

To support the interpretation of various aspects/levels of information and to obtain global insight over a large amount of subjective documents, we attempt to detect topics, opinions, reasons, and relations between document contents, and then to classify and summarize them, including numbers of occurrences. This approach classifies and generates indices of document contents in several complementary aspects: opinions, subtopics, and arguments.

This multidimensional classification not only provides a summarized overview of the data collection but also enables subsequent sequential or hierarchical search of information by following the link from the summarized output to the individual file output. In this paper, we address several aspects of texts and implement each appropriate classifier showing valuable performance.

Our experiment is performed in the context of public comments sent to government officials in response to proposed regulations. Every year thousands of regulation changes are reviewed by the public. In exceptionally controversial cases, such as the Environmental Protection Agency’s recent rulemaking about mercury, over 536,000 e-mail messages and additional 4,264 paper submissions were collected. Because of the government’s practical need to analyze and respond to this huge amount of comments, we perform analysis in the domain of public comments.

The rest of the paper is organized as follows: first, we describe the overall architecture of the system and the individual components recognizing important information units in various levels, next, provide the experimental result and evaluation, and finally, conclude and discuss future work.

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.

K-CAP ’07, October 28–31, 2007, Whistler, British Columbia, Canada.
Copyright 2007 ACM.

OVERVIEW

The system is composed of four sub-modules identifying different aspects of text: subtopic categorization, keyword extraction and clustering, argument structure analysis, and opinion classification. From the classified result, we provide a table that summarizes the whole collection and links to associated documents highlighted where relevant. The related documents can be accessed by clicking the grouped data. The overall structure is shown in Figure 1.

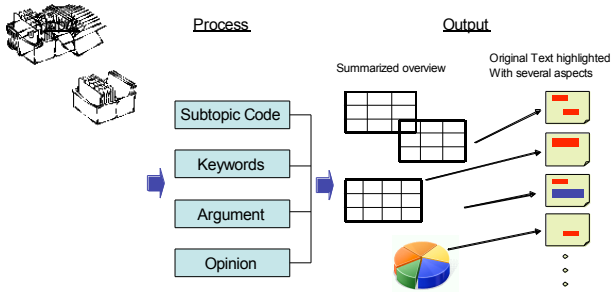


Figure 1. System Overview

Subtopic Categorization

We attempt to classify topics into predefined classes which are generally applicable and support readers' interest in our test domain. The subtopic codes are defined by a social scientist and a political scientist: economic, environment, health, legal, policy, pollution, science, and technology.

Since the documents cover diverse topics to assess or discuss the problem or to suggest some ideas, we identify subtopic codes for each sentence. For example, the following sentence is identified with the codes of *health* and *economic*.

The risks to the health of future generations of Americans, and the associated costs that will be borne by our families and communities, far outweigh the reasonable cost of prevention to utilities.

We interpret the problem as a classification of yes or no for each subtopic code. Based on the assumption that the codes are independent of each other, we perform each separate classifier and assign all related subtopic codes (from none to multiple codes) given a sentence.

The classifiers are built using a support vector machine (SVM) [22] implemented in SVM-light. SVM is a machine learning method widely used in classification problems showing sound performance in many applications, which finds a hyper-plane that separates the positive and negative training examples with a maximum margin in the vector space. In this problem, the vector space is composed of the following features:

- Word's lexeme, bigram.
- Named entity and label obtained by BBN Identifinder [4]
- Synonyms of the first sense of word in WordNet [12]

Keyword Extraction and Clustering

In contrast to subtopic categorization that searches text for predefined subtopic code clues, we find salient or frequent keywords from the raw documents with no prior knowledge.

Lexical Chaining

Rather than selecting frequent words, we search for important concepts by identifying lexical chains. A lexical chain is a sequence of semantically related word occurrences in a document, representing lexical cohesion. A set of lexical chains is considered showing underlying discourse structures [11] and it is used as an intermediate representation of text in many natural language applications such as summarization [3][19], information retrieval [20], and intelligent spell checking [8].

Morris and Hirst [11] first presented a computational algorithm identifying lexical chains using relations in Roget's Thesaurus. However, their algorithm did not include sense disambiguation, which resulted in the confusion where all thesaurus entries of a polysemous word were mixed together. Later, several other models using WordNet [12] relations were suggested for word sense disambiguation in lexical chaining: [3][5],[8],[19].

Our approach is similar to Galley and McKeown's algorithm, which was reported as efficient and accurate. We generate a graph of noun or noun phrase senses linked with the weights computed by WordNet relations and distances in a document. From the graph, we identify the sets of linked senses with higher weights as lexical chains.

The first step is to find all possible sense interpretations of each noun (phrase) occurrence, and build a list of senses with a weight 1. Checking the sibling and hypernym relations defined in WordNet, we link the related senses and assign the corresponding weights defined in Table 1, which is adopted from [5].

Table 1. Weights between two noun (phrase) occurrences using the relations in WordNet and the distances in text.

Semantic relation	1 sent.	3 sent.	1 par.	Other
Synonym	1	1	0.5	0.5
Hypernym/hyponym	1	0.5	0.3	0.3
Sibling	1	0.3	0.2	0

The next step is to disambiguate word senses in context. We assume one sense per word in a document. For each word, we select the most likely sense having the highest computed weight in the first step. In case of tie, we choose the first sense defined in WordNet (more frequent sense). Since the weights are computed by all related senses regardless of the actual sense of the word, we adjust the weights assigned on the link to the unselected senses whenever we determine the word sense. This iterative process is different from Galley and McKeown's algorithm.

After assigning the proper sense per word, we link the related words traversing the links between senses in the graph built in the first step.

In other related work, various factors have been used to compute the strength to select more important chains, including repetition, density, length, homogeneity index, etc. However, we do not select important lexical chains for an individual file but we choose salient lexical chains for all documents after clustering all lexical chains.

Clustering the Lexical Chains

We perform hierarchical clustering for lexical chains: we assume all lexical chains as individual clusters and combine two closest clusters iteratively until the distance is far enough (using the experimentally determined threshold).

The distance between clusters are determined by the cosine similarity between the vectors of noun senses with the weight computed in the previous lexical chaining step:

$$d_{i,j} = \frac{\sum_k w_{k,i} w_{k,j}}{\sqrt{\sum_k w_{k,i}^2} \sqrt{\sum_k w_{k,j}^2}}$$

where $d_{i,j}$ is a distance between vector i and vector j , $w_{k,i}$ is a weight of k^{th} element of vector i . When two clusters are combined, the weight for the newly combined cluster is recomputed by Ward's method as follows:

$$d_{k,i \cup j} = \frac{(n_i + n_j) \cdot d_{ki} + (n_j + n_k) \cdot d_{kj} - n_k \cdot d_{ij}}{n_i + n_j + n_k}$$

where $d_{k,i \cup j}$ is a distance between the arbitrary cluster k and the newly combined cluster of i and j , and n_k is the number of instances in the cluster k . The top n clusters of lexical chains are selected as a set of keywords of the document collection.

Argument Element Recognition

Although we have considered all sentences equally important so far, this is not in general valid. The discourse structure shows where more important information resides. Since the documents in our test domain are comments on the government's proposed regulation, they show argument structure of claim and reasons. In this module, we identify major argument elements of "main claim" and "sub-claim or main-support" for the main claim.

First, "main claim" is recognized by a binary classifier for each sentence, using the lexical and structural features. The features are designed by eyeballing the training data set and the final feature set is determined by several rounds of tests on the development set as follows:

- **Unigram, bigram, and word's lexeme:** Since the documents are from a single domain, frequent n -gram can help find claims. We also expect that these features detect popular patterns or (adverbial) phrases.

- **Subjectivity:** We assume the main claim is rather subjective than objective description, hence we count positive and negative words defined in the General Inquirer [6].

- **Position:** Especially in well-written texts, the main claim is highly related to the position in the text. We indicate a position with three values:

- *Relative paragraph position:* the relative position of paragraph that includes the given sentence scaled to the interval [0,1].
- *Sentence position in a paragraph:* the order of the sentence in a paragraph in the region of [1,n] where n is the total number of sentences in a paragraph.
- *Relative sentence position in a paragraph:* the sentence order represented as a relative position in a paragraph, scaled to the interval [0,1].

- **Subhead:** From the sentence parse tree (using Charniak's parser), the main predicate is obtained by taking the headword of the sentence. For the parent of the predicate (verb phrase (VP) for verb predicate), the direct children nodes are selected and the sequence of headwords of each child node is used. For the sentence of "I strongly urge you to withdraw the proposed rule", the subhead is identified as "urge you to".

- **Subtopic "Policy":** Since the arguments in this domain are about the proposed regulation, we assume the main claim will probably mention the regulation. We use the binary indicator of the subtopic "policy", which is the output of the first module, subtopic identification.

Second, we recognize the "sub-claim or main-support" of the main claim. We simplify the task, assuming that text is a sequence of subtopic segments. We segment text into several subtopic groups and select the most important sentence from each segment.

The segmentation is performed using Hearst's TextTiling [7], which utilizes lexical co-occurrence and distribution, to detect topic-shift. To define a single important sentence from a segment obtained, we compute the score using many features and select the one ranked as highest of all sentences in the segment. The similar features are used to the main claim identification including unigram, bigram, word's lexeme, main predicate, subcategorization, subhead, and slightly modified position (the relative position within a segment).

As a framework to interpolate these diverse features, we use the Support Vector Machine (SVM classifier for main claim identification and SVM ranker [9] for sub-claim identification implemented in SVM-Light) and boosting algorithms [10]. For boosting, we use the BoosTexter [17] implementation, which combines many simple, moderately inaccurate rules into a single accurate rule, by the sequential training where each rule is tweaked in favor of the instances misclassified by the preceding rules.

Opinion Classification

Having identified the main claims of the documents, we classify the claim into predefined classes of attitudes. Much research has been performed for subjectivity identification or polarity classification [16][21][23].

Instead of classifying claims into traditional polarities of positive, negative, and neutral, we classify the claim in terms of the attitude to the topic. We classify the claim into either “support” or “oppose” the given topic (regulation), and we add another class “propose a new idea” for claims not directly stating an opinion about the main topic but proposing some new idea.

As a resource of polarity detection, we first build positive and negative word (phrase) lists using a small set of seed words.

Building Subjective Clues

Each word can be characterized with many attributes and one of them may be positive and negative semantic orientation. For example, “good”, “honest”, and “happy” have positive orientation, and “bad”, “disturb”, “violence” have negative orientation. This semantic orientation is often used as a basis to recognize the polar opinion in sentence or document.

We start with a small set of comparably clear polar (positive and negative) expressions defined in General Inquirer [6]. General Inquirer (GI) is a manually developed and publicly available dictionary defining various properties of words including positive and negative polarity. Among 8,720 lexical items (11,788 senses), 1,622 lexical items (1,915 senses) are annotated as positive and 1,992 lexical items (2,291 senses) as negative.

We extend this set of polar words to multiword phrases, since the presence of subjectivity and its polarity are often determined by the way the positive and negative clues are combined in context rather than by their frequency of a single word. A simple example is a negation where the polarity is inverted, such as “not good”, “no doubt” and “nobody likes”. Sometimes, a word can work as a subjective clue or as an intensifier (e.g., “a great future” vs. “a great disappointment”). It is even harder to determine the polarity when the expression is long (e.g., “there is no reason to believe” vs. “there is no doubt”; “the beautiful background of the monument” vs. “the background of a stagnant economic condition”).

The seed list of positive and negative words in GI is extended using paraphrases constructed from machine translation corpus. We obtain a large domain-independent paraphrase collection [24], which was automatically built based on the assumption that two different phrases of the same meaning may have the same translation in a foreign language. From a Chinese-English parallel corpus of 218 million English words, phrase pairs were aligned and extracted by the method described in [14],[15].

Given the paraphrase collection, we assign the polarity to the cluster of paraphrases (all phrases within a cluster share

the same meaning) by referring the positive and negative words defined in General Inquirer¹. When one or more than one paraphrase in a cluster is annotated as positive and no paraphrase as negative in General Inquirer, the cluster is assigned as positive. This is same for negative polarity. If both positive and negative paraphrases exist in one cluster, the cluster is defined as neutral².

Error! Reference source not found. shows the example clusters annotated with polarity. All phrases in a cluster are extended from the word defined in GI (“help” and “inadequate”). We obtain 4,592 positive clusters (24,031 phrases) and 3,629 negative clusters (17,893 phrases) by the process.

Table 2. Sample Paraphrase Clusters with Polarity

Cluster1: <i>Positive</i>
service will not only assist will not only facilitate it would not only help not only assists the not only been useful will not only enable would not only help will not only help not only helps to not only enables it will help to not only helps not only would not only help will help to have helped would help will help can help helps to it helps enables helps
Cluster 2: <i>Negative</i>
characterized by a lack of tangible progress lacking in anything worth mentioning two immediate concerns of the people have little to write home about nothing worth taking note nothing worth mentioning promoting it development lacking in intensity not worth mentioning nor merits to speak lack of progress inadequate

This extended repository seems similar to the n -gram approach, but differs in finding proper units (boundaries) of polar expression rather than the arbitrary length of n -grams. The extended list not only increases the coverage of the polar expressions but also detects the contextual polarity of phrases, so that we can improve the polarity classification.

Classification Features

Having built opinion-holding paraphrases, we need more features to classify the claim sentence in terms of opinion or attitude over the main topic. Many syntactic and semantic features, as well as polar expressions we obtained, are used, which are integrated in a machine learning framework (BoosTexter).

• **Positive & Negative words:** The positive and negative words defined in General Inquirer and their accumulated frequencies are used. For example, from the sentence of

¹ We do not perform sense disambiguation. If at least one sense has the polarity positive or negative, we assume the polarity for all senses of the lexeme. When a lexeme has both positive and negative polarities for different senses (15 entries found in total), we assign “neutral” to the lexeme.

² Since the automatically generated paraphrase collection contains errors, this simple binary decision for the cluster outperforms the probabilistic model using the frequencies of individual phrases.

“We also oppose the proposal to allow toxic mercury credit trading.”, “oppose” has negative polarity and “allow” and “credit” has positive polarity.

- **Positive & Negative phrases:** We use positive and negative phrases extended from General Inquirer using paraphrases obtained from a machine translation corpus and the number of occurrences. For example, “I believe there is no reason that the original stipulation should be altered.” does not contain any polar words defined in General Inquirer, but “there is no reason” is found from the polar phrase list. We find the longer polar expressions first and then consider shorter one for the remaining parts. For example, we count “evil forces” as a unit, but we do not count “evil” and “forces” separately although they are defined in our collection.

- **Main predicate:** The headword of the sentence is identified from the parsed data. In the above example sentence, “believe” is a main predicate.

- **FrameNet frame:** As a way to generalize the main predicate, we find possible frames of the main predicate defined in FrameNet [2]. For the main verb “believe”, four possible frames of *Awareness*, *Certainty*, *Trust*, and *Religious_belief*, are obtained.

- **Subcategorization:** The parsing rule that expands the parent of the main predicate, verb phrase (VP) for a verb predicate, is obtained.

- **Unigram, bigram, and trigram:** The traditional n -gram features are applied to find useful subjective expressions or topical information.

We only use the *positive* and *negative* words or phrases, and do not include clue words for “propose a new idea”. The decision on that class depends on the lexical information of bigram, trigram, and main predicates, combined with positive and negative polarity.

Evaluation

Each document is identified with multiple indices including subtopic, keyword, argument structure and classified opinion. We evaluated each module separately. The experiments were performed on the public comments on a regulation about a hazardous pollutant control.

The first experiment is about topic identification for each sentence. 160 comments were randomly selected and divided into training set (118 documents), development set (20 documents), and test set (22 documents). The data set is annotated by two human coders in parallel and the agreement between system and human is reported in F-measure. **In Error! Reference source not found.**, “Human” shows the agreement between two coders, “BL1” is a baseline to assign all codes to all sentences, “BL2” is to assign the most common code *pollution* to all sentences, “BL3” is to assign the code when a sentence contains a morphological variant of the corresponding code name, and “SYS” shows our SVM classification approach. Our system performance

is as high as human annotation agreement in some specific codes, and significantly outperforms the baselines.

Table 3. Evaluation on Subtopic Categorization (F-measure)

Code	Human	BL1	BL2	BL3	SYS
Economic	0.76	0.18	0	0.18	0.57
Legal	0.52	0.11	0	0.17	0.47
Health	0.82	0.36	0	0.41	0.78
Science	0.54	0.21	0	0.56	0.53
Policy	0.73	0.44	0	0.30	0.73
Technology	0.86	0.11	0	0.79	0.83
Environment	0.46	0.22	0	0.21	0.30
Pollution	0.75	0.48	0.78	0.46	0.65
Total	0.72	0.26	0.39	0.40	0.67

For our keyword extraction, the evaluation was conducted in two steps. We first evaluated the appropriateness of lexical chains we built, by checking Word Sense Disambiguation of noun occurrences. The lexical chaining process includes the word sense disambiguation, and correct disambiguation is a key issue in linking related words. Following Galley and McKeown’s evaluation [5], we experimented our lexical chaining on Semcor 2.0 (Semantic Concordance Corpus) where all words were tagged with WordNet senses. We checked the accuracy on word sense disambiguation for 103 files (43,603 noun instances) from the corpus. Our lexical chaining process (59.45%) performs qualitatively same as Galley and McKeown’s algorithm (59.06%) that has been reported showing the highest accuracy compared to other related work ($p = 0.005$).

Second, we evaluated the value of keywords generated using lexical chain clustering. To assess how much representative the selected keywords were, we checked the selected keyword occurrences in the user’s answers to the question about the contents. We provided 278 documents from the public comments with a general search tool to six users, and asked a question about the comments: *Identify as many as possible main reasons people use: list them and provide an illustrative example text for each reason: 1. for criticizing 2. for supporting 3. for principal/substantive suggestions or requests regarding the proposed regulation.* From the answers that six users provided, we computed the keyword occurrences with the weight assigned by the number of answers containing the corresponding words, following the Pyramid method in [13], that is an official summarization evaluation tool for Document Understanding Conference (DUC). We compared the Pyramid score with the baseline which extracts keywords based on *tf.idf* method. The result is shown in **Error! Reference source not found.**

Table 4. Evaluation on Keyword Extraction

# of Keywords	Baseline (<i>tf.idf</i>)	System (Lex Chain)
10 keywords	0.3934	0.5573
20 keywords	0.4225	0.9154

Error! Reference source not found. shows the performance of the argument element recognition module, represented in F-measure. 78 documents from the comments were annotated by two human coders, and 5-fold cross validation was conducted since the document set is not big enough to obtain independent training and test set. The data set was mixed with short and long documents and the average number of sentences per document was 30. There is no previous algorithm directly comparable to the system yet, we compared with a baseline to assign the first sentence of a document as a main claim, and first sentence of each paragraph as a sub claim, which is commonly used as a baseline in summarization task. Our system shows valid performance compared to baseline and human agreement.

Given the fact that even the human agreement on selecting main or sub claim is not that high, we speculate that duplicate (rephrasing) sentences for one claim can cause disagreement. Hence we computed the cosine similarity of words appearing in the main and sub claim. The similarity between human and system is 0.71, when the similarity between two human coders is 0.78 and the baseline is 0.57.

Table 5. Evaluation on Argument Element Recognition

Test	Human	Baseline	SYS1 (SVM)	SYS2 (BoosTexter)
Main Claim	0.68	0.19	0.52	0.55
Sub Claim	0.60	0.27	0.50	0.50

Error! Reference source not found. shows the result of opinion classification. The classification between “support” and “oppose” is comparably easier to detect, but to find “propose a new idea” is more confusing. However, it still outperforms the baseline consistently. In this case, baseline is to assign “oppose the regulation”, the most common class. Although there is much related work on opinion classification, the definitions of the class and annotation differ for each work, hence we cannot directly compare with other algorithms. We report our system performance compared to the baseline and human annotation as in other previous work.

Table 6. Evaluation on Opinion Classification

Classes	Human	Baseline	System
2 (support/oppose)	1	0.86	0.91
3 (support/oppose/propose)	0.80	0.59	0.67

Results and Discussions

Having analyzed each document in several aspects, we provide integrated and summarized output for the whole data collection. From the individual output identifying subtopic, keywords, arguments, and opinion, we cluster similar contents for each aspect and display combined overview. A sample integrated output from 400 documents is shown in Figure 2. Given a class of opinion and a cluster of keywords, the related documents are linked and the main claims are displayed as representatives.

Opinion	Keywords (lexical chains)	Cnt	Main Sentences
propose	lake, river, stream, water, waterway	1	One of these criteria is "All other waters such as intrastate lakes, rivers, streams (including intermittent streams), mudflats, sandflats, wetlands, sloughs, prairie potholes, wet meadows, playa lakes, or natural ponds, the use, degradation or destruction of which could affect interstate or foreign commerce including any such waters..."
oppose	bay, canal, channel, entity, estuary, lake, orange, pond, pool, requirement, river, sea, something, source, stream, thing, water, waterway	1	... any other revisions are needed to the existing regulations on which waters are jurisdictional under the CWA."
		33	I strongly oppose EPA's efforts to eliminate Clean Water Act protections for many of the nation's waters, including streams, wetlands, small ponds, and other waters.
		1	This proposal is in direct conflict with the original intent of the Clean Water Act -- to maintain the chemical, physical and biological integrity of all waters of the United States.
--wetland, bottom, ground, land, soil, wetland		1	As a supporter of the environment and wildlife, I strongly object to the U.S. Environmental Protection Agency and the U.S. Army Corps of Engineers' recent actions to dramatically reduce the scope of the Clean Water Act.
		1	I am VERY STRONGLY opposed to the proposed change in the definition of "Waters of the United States" for the following reasons:
		1	In sum, NDEQ is seriously concerned about the implications of this ANPRM upon State water quality, the health of surface waters, the public interest, the State's economy, and complementary water quality programs.
		1	SWANCC clearly eliminates CW-A jurisdiction over isolated waters that are intrastate and non-navigable, where the sole basis for asserting CWA jurisdiction is the actual or potential use of the waters as habitat for migratory birds.
action, agency, condition, destruction, health, improvement, integrity, level, office, order, participation, point, position, preparation, restoration, state		1	... any other revisions are needed to the existing regulations on which waters are jurisdictional under the CWA."
		1	Our groups urge you to abandon your proposal to revise the CWA definition of "Waters of the United States" and to withdraw your seriously flawed Guidance Memorandum.
		1	Our organizations again urge you to abandon your proposal to revise the CWA definition of "Waters of the United States" and to withdraw your seriously flawed Guidance Memorandum.
		1	The protection of isolated systems is vital in maintaining the integrity of these adjoining water bodies.
acre, district, jurisdiction, region		1	I am strongly opposed to any such policy.
		1	Our groups urge you to abandon your proposal to revise the CWA definition of "Waters of the United States" and to withdraw

Figure 2. Sample Overview Output of a Data Collection using Multiple Classifications

From the overview page, system users can click the number and see the related file list that is linked to the individual processed file. For clustering similar claims, various clustering methods and levels can be applied, but we group using simple lexical similarity. As mentioned in the introduction section in this paper, lexical similarity is not really enough for this kind of opinionated documents, however, since the claims are already classified by opinion, even the lexical similarity check can work well. More sophisticated clustering methods using structural or semantic information surely can be investigated, but we leave it for future work.

Compared to the text-only summarization, this multidimensional classification approach can show clearer relations between elements and provide an easy-to-see overview picture. Further, system users can focus on the specific aspects that they are interested in, and search for more information regarding the aspect or class by following the link from the overview page to the individual page.

Conclusion and Future Work

To facilitate understanding of a large collection of documents, we have partitioned the analysis process into independent complementary dimensions (namely subtopic categorization, keyword extraction, argument element recognition, and opinion classification) and provide an integrated output that integrates these aspects of analysis. Each module outperforms the relevant baseline significantly, and shows valuable output.

The integration of multiple classifications not only provides a summarized overview of the data collection but also enables sequential or hierarchical search for each user's interest. The suggested modules can detect generally important information especially for subjective documents and can be applicable to other domains.

This novel approach to analyze the documents with multiple classifications can be pursued in two directions in future. First, we can explore the methods to improve the individual modules and develop more general module, and second, visualization techniques and clustering of similar contents to integrate the output can be investigated.

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] Barzilay, R. and Elhadad, M. Using Lexical Chains for Text Summarization. In *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97)*, ACL, Madrid, Spain. (1997)
- [4] 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).
- [5] Galley, M. and McKeown, K. Improving Word Sense Disambiguation in Lexical Chaining. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, Poster paper, Acapulco, Mexico. (2003).
- [6] General Inquirer. <<http://www.wjh.harvard.edu/inquirer/>> (2002).
- [7] Hearst, M. TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages. *Computational Linguistics*, 23 (1), pp. 33--64. (1997).
- [8] Hirst, G. and St-Onge, D. Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms. In *WordNet: An Electronic Lexical Database*. MIT press. (1998).
- [9] Joachims, T. Optimizing Search Engines Using Click-through Data, In *Proceeding of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, ACM, Edmonton, Alberta, Canada. (2002).
- [10] Meir, R. and Ratsch, G. An Introduction to Boosting and Leveraging. *Advanced Lectures on Machine Learning*. Springer-Verlag New York, Inc. (2003).
- [11] Morris, J. and Hirst, G. Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. *Computational Linguistics*. 17(1):21-48. (1991).
- [12] Miller, G. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235-312. (1990).
- [13] Nenkova, A. and Passonneau, R. Evaluating Content Selection in Summarization: the Pyramid Method. In *Proceedings of NAACL-HLT*, Boston, MA. (2004).
- [14] Och, F.J. and Ney, H. A Systematic Comparison of Various Statistical Alignment Models, *Computational Linguistics*, 29(1):19-51. (2003).
- [15] Och, F.J. and Ney, H. The Alignment Template Approach to Statistical Machine Translation, *Computational Linguistics*. 30(4). (2004).
- [16] Pang, B., Lee L., and Vaithayanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of EMNLP*, Philadelphia, PA. (2002).
- [17] Schapire, R. and Singer, Y. BoosTexter: A Boosting-Based System for Text Categorization. *Machine Learning*. 39(2/3):135-168. (2003).
- [18] Shulman, S.W. E-Rulemaking: Issues in Current Research and Practice. *International Journal of Public Administration* 28: 621-641. (2005).
- [19] Silber, G. and McCoy, K. Efficiently Computed Lexical Chains as an Intermediate Representation for Automatic Text Summarization. *Computational Linguistics*, 29(1). (2003).
- [20] Stairmand, M. *A Computational Analysis of Lexical Cohesion with Applications in Information Retrieval*. Ph.D. Dissertation, Center for Computational Linguistics UMIST, Manchester. (1996).
- [21] 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).
- [22] Vapnik, V. N. *The nature of Statistical Learning Theory*, Springer. (1995).
- [23] Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of HLT-EMNLP*, Vancouver, Canada. (2005).
- [24] Zhou, L., Lin, C., Munteanu, D.S., and Hovy, E. <<http://www.isi.edu/~liangz/DEMO/PARA>> (2006).