Information Engineering Express
International Institute of Applied Informatics
2015, Vol.1, No.4, 107 – 118

Automatic Dictionary Extraction and Content Analysis Associated with Human Values

Yasuhiro Takayama [*] , Yoichi Tomiura [†] ,
Kenneth R. Fleischmann [‡] , An-Shou Cheng[§] ,
Douglas W. Oard [¶] and Emi Ishita [||]

Abstract

This paper studies a method for identifying word unigrams and word bigrams that are associated with one or more human values such as freedom or innovation. The key idea is to deterministically associate values with word choices, thus permitting values reflected by sentences to be assigned using dictionary lookup. This approach works nearly as well on average as the most accurate existing methods, but the principal contribution of the new method is that the basis for the system's classification decisions are more easily interpreted by social scientists. The new method is based on using a Monte Carlo algorithm with simulated annealing to efficiently explore the space for optimal assignments of human values to unigrams and bigrams. Results are reported on an annotated test collection of prepared statements from witnesses at public hearings on the topic of net neutrality. The results include both accuracy comparisons with a previously reported approach and the use of emergent human coding to explain the classification process in a way that social scientists find to be useful as a way of characterizing the use of word pairs to express human values in this context.

Keywords: Simulated Annealing, Natural Language Processing, Computational Social Science, Human Values.

1 Introduction

This paper describes a method for extracting a human values dictionary containing word unigrams and bigrams for detecting human values reflected within sentences in prepared statements related to the net neutrality debate. Human values such as freedom or justice are

[*] National Institute of Technology, Tokuyama College, Yamaguchi, Japan
[†] Kyushu University, Fukuoka, Japan
[‡] The University of Texas at Austin, Texas, USA
[§] National Sun Yet-sen University, Kaohsiung City, Taiwan
[¶] University of Maryland, College Park, Maryland, USA
[||] Kyushu University, Fukuoka, Japan

useful to explain people's opinions. The values are indicated by certain linguistic expressions, typically words and word sequences in texts which writers' appeals to specific vales are reflected.

Human values have proven to be a useful construct for explaining people's choices. Social scientists have used human values to explain attitudes and behaviors [19]. For example, someone who values innovation over wealth might advocate open-source over proprietary software, while someone who values freedom over social order might resist efforts for gun registration. In the net neutrality debate, support of net neutrality is positively correlated with valuing innovation and negatively correlated with valuing wealth [4].

Sentiment analysis (sometimes called opinion mining) is a related research field in which the task is typically focused on detecting the sentiment polarity (positive, negative, neutral, ambivalent, $\cdots$) [16]. Although sentiment analysis alone can be useful in several applications, social scientists have also sought to look more deeply for factors that might help to explain sentiment. The analysis of human values provides one such opportunity [4], [7], [8].

Lexical features such as words and word stems have been shown to be a useful basis for detecting human values [12], [21]. We take a step in that direction by constructing a values dictionary for use in automated detection of human values that are expressed or reflected by the word choice in specific sentences. This values dictionary in turn can facilitate qualitative analysis by social scientists of the association between values and word choice by providing cue expressions in an easily used form (specifically, as a list of word or stem unigrams and bigrams).

In this paper, we focus on a method for automatic construction of the values dictionary, on an empirical demonstration that the resulting values dictionary can be used to achieve classification results that are quite competitive with ones of the existing state of the art, and on illustrating the usefulness of the resulting values dictionary through content analysis of the word bigrams that are associated with each human value.

2 Related Work

Content analysis is one approach to analyzing the effects of human values [4], [7], [8], [24]. One approach to content analysis is for social scientists to personally examine naturally occurring content and assign codes to that content that reflect their interpretations using some coding scheme. Subsequent statistical analysis can then be done on the assigned codes, rather than on some less rich and less well tailored model of the meaning of the content. Hsieh and Shannon [11] refer to the combination of human interpretation and a coding scheme as a "directed approach," the scale of which may be limited by annotation costs. Automation can in principal help with at least a part of this process. Early in the annotation process, social scientists often tailor some pre-existing coding scheme to better match the details of the phenomena to be coded that actually exist in the collection. Our view of the role of automation is focused on what happens after that, once coding guidelines have stabilized and a sufficient amount of annotated content is available. At that point, we might reasonably hope to at least partially automate the remainder of the annotation process by training a text classifier using that initially annotated content [20].

Using text classification is not a new idea. Most notably, in social psychology the LIWC lexicon [23] (Linguistic Inquiry and Word Count) associates general terminology with categories that seek to reflect the psychological "meaning" of words, and LIWC is thus a natural basis for building rule-based classifiers. What distinguishes the LIWC lexicon from our proposed values dictionary is that we focus on human values and on automating the construction of a collection-specific values dictionary.

Several efforts for automatic values classification have been tried. In early work, Ishita et al. [12] found that a k-NN (k nearest neighbor) classifier using word unigrams as features could be trained to detect human values. Takayama et al. [21] subsequently compared k-NN, naive Bayes, and SVM (Support Vector Machine) classifiers [13], using word stem unigram features, finding the SVM to be the best choice. They also sought to further improve the SVM's effectiveness by expanding the feature set using associated words identified based on statistical similarity (from unlabeled text) or from a hand-crafted thesaurus, but further improvements proved elusive.

Our work in this paper is inspired in part by new so-called "topic modeling" approaches that seek to model natural language text using probabilities [1], [10], [22]. One limitation of this family of techniques, however, is that in an effort to optimize classification effectiveness they use techniques that can make their operation rather opaque from the point of view of social scientists. As a simple example, a naive Bayes classifier infers category labels from the contribution of many words, each of which may have only a small probability of being associated with that label. More sophisticated methods such as LDA (Latent Dirichlet Allocation) [1], [10], in which the association between words and "topics" (in our case, human values) is also modeled probabilistically, suffer from the same problem: nothing is known until everything is known.

The content analysis process engaged in by social scientists is typically more reductionist, however, in that social scientists performing content analysis might wish to associate a value with a short passage, or even with just a single word. Therefore, rather than seeking only to optimize classification accuracy, we additionally want to build explainable classifiers in which we associate words (or word bigrams) with specific values in ways that can then be aggregated to predict sentence-level values using an easily explained process. In this way, we produce a values dictionary that can be used as a basis for qualitative analysis of the ways in which language is used to express specific values.

To do this, we draw on an optimization method known as simulated annealing [14]. Simulated annealing has previously been applied to parsing [18] and to word sense disambiguation (WSD) [6] in natural language processing. Our application of simulated annealing is similar to that which has been applied for WSD, since we can view values as being in some way parallel (computationally) to sense labels. Many implementation details differ, however, since in our task we must infer the association between words and values from annotated associations between values and entire sentences.

When social scientists perform qualitative analysis, it is much easier to interpret relationships between values and sentences if the values of the sentence deterministically depend on just the occurrences of specific words (word sequences) in that sentence. Therefore, we will propose a new method for extracting values dictionary that is applicable to automatic values estimation for unseen sentences and demonstrate the dictionary is applicable for qualitative analysis.

3 Test Collection

The section briefly describes the test collection that we use in this paper. The test collection is available at http://nlp.inf.kyushu-u.ac.jp/values.html. This collection was originally

Figure 1: Example sentences from the Net Neutrality test collection.

value	sentence
freedom	We must ensure that local franchise requirements are not a barrier to competition.
honor	American content companies like Amazon.com are world leaders today.
innovation	We also welcome broadband network operators' innovations within the network.
justice, *s-order*	The process of enforcement could undermine basic Constitutional protections as well.
freedom, *wealth*	And we all depend on choice and competition to deliver these services at the lowest possible price and at the best possible quality.
freedom, *justice*	We must ensure that consumers have unfettered access to all internet content free from discrimination.

developed by Cheng [5]. It includes 102 written prepared statements from public hearings held by the U.S. Congress and Federal Communications Commission (FCC) on net neutrality. The sentences annotated with zero or more of six human values: *freedom*, *honor*, *innovation*, *justice*, *social order*, and *wealth*. These human values were selected by Cheng from the meta-inventory of human values based on their prevalence in the collection and the reliably with which they could be coded [3]. The test collection contains 8,660 sentences from 102 complete documents, and all of the words have been stemmed using the Porter stemmer. The average sentence length in the collection is 10.3 word stems.

4 Method

Our proposed method is inspired by a Monte Carlo algorithm [17], which is more general than popular MCMC (Markov Chain Monte Carlo) algorithms such as Metropolis-Hastings or Gibbs sampling. We adopt a bit pattern representation for multiple human values. Then we apply a simulated annealing method [14] to associate an (approximately) optimal set of human values to specific word stem unigrams and bigrams. Our proposed algorithm optimizes the values assignment to maximize the F_1 measure, which is the harmonic mean of precision and recall.

4.1 Preparation and Notation

A sentence $\mathbf{w}$ is a sequence of N word stems denoted by $\mathbf{w} = (\, w_1, w_2, \ldots, w_n, \ldots, w_N)$, where w_n is the n-th word stem in the sequence. The sentence $\mathbf{w}$ has sentence-level values v, where $v \in \{0,1\}^6 = \{000000, 000001, \ldots, 111111\}$. Each bit in the sequence represents one of the six values; in order, they represent *freedom*, *honor*, *innovation*, *justice*, *social order*, and *wealth*, respectively. The bit sequence 000000 corresponds to a sentence that is not annotated with any human values.

The human values for a word stem (unigram) w are denoted by $v(w)$ and the human values for a word stem w_2 under the influence of its previous word w_1, (i.e., the values for a word stem bigram $(w_1\ w_2)$) are denoted by $v(w_2; w_1)$. The values dictionary that we are constructing is a collection of $v(w)$ and $v(w_2; w_1)$ for all word stems and word stem bigrams. We restrict $v(w)$ *and* $v(w_2;\ w_1)$ to be an element in χ , where $\chi = \{000000, 000001, 000010, 000011, 000100, \cdots, 110000\}$ is the set of all bit vectors that have at most two non-zero values. The cardinality of χ is 22. Restricting the number of values with which a word can be associated limits sparsity. Whether an at-most-two model is a good choice is an empirical question. In preliminary experiments, the model with less restrictive constraints (at-most-three) showed no further improvement.

The sentence-level values depend on values assigned to each word stem (which we call word-level values). The word-level values of w_n are either $v(w_n)$ or (if it exists) $v(w_n;w_{n\text{-}1})$. We give preference to the values $v(w_n;w_{n-1})$, preferring assignments that are informed by more evidence. Thus we define the word-level values as follows:

$$x_n = \begin{cases} v(w_n;w_{n-1}) & ;\ v(w_n;w_{n-1}) \neq 000000 \\ v(w_n) & ;\ v(w_n;w_{n-1}) = 000000. \end{cases} \quad (1)$$

The sentence-level values for a sentence **w** is then calculated as follows:

$$x_1 \oplus x_2 \oplus \cdots \oplus x_N, \quad (2)$$

where the symbol $\oplus$ is a logical bitwise OR operator.

We can treat the values dictionary construction, that is, the calculations of $v(w)$ and $v(w_2;w_1)$ for all words and word bigrams, as a combinatorial optimization problem for text annotated with sentence-level values. The objective function that we seek to optimize is the average (over all six values) of the F_1 measure of the resulting estimate of sentence-level values that results from equation (2). Exact optimization is impractical because there are 22 values for both $v(w)$ and $v(w_2;w_1)$ and the number of distinct word stems numbers in the thousands. Therefore, we find an approximate solution by a probabilistic search. Based on a preliminary manual analysis of our test collection, we make the following reasonable assumptions about the relationship between words and values to guide the selection of metar-parameters:

(a) Most words are not associated with any values, so only a few words will serve as values indicators.

(b) The values associated with most two-word sequences can be modeled compositionally based on the values associated with each word in the sequence.

From assumption (a), any given word w is unlikely to be associated with any values. On the other hand, if word w_2 is associated with value k, then any sentence in which word w_2 occurs must express value k unless the word sequence w_1w_2 is associated only with values other than k. However, assumption (b) tells us that such exceptions will be rare, and thus that most sentences that include word w will express value k whenever the word w is associated with value k. Therefore, we can assume that the word w is likely to have a value k if many of sentences in the training part of the collection that contain w are annotated with value k. Thus we limit the search space of possible values for $v(w)$, as follows:

$$[v(w)]_k = \begin{cases} 0 \ \ or \ \ 1 & ;\ \frac{ns(w,k)}{ns(w)} \geq \alpha, \text{ and } ns(w) \geq FW \\ 0 & ;\ otherwise, \end{cases} \quad (3)$$

where $[b]_k$ represents the k-th bit in bit sequence b. The variable $ns(w)$ is the number of training sentences in which the word w occurs, and $ns(w,k)$ is the number of training sentences in which word w occurs that are annotated with value k. The meta-parameter α is the minimum ratio and the meta-parameter FW is the minimum number of sentences to make the observed counts sufficiently reliable. Maximizing bitwise to compute the $maxV(w)$ needed for our simulated annealing algorithm yields:

$$[maxV(w)]_k = \begin{cases} 1 & ;\ \frac{ns(w,k)}{ns(w)} \geq \alpha, \text{ and } ns(w) \geq FW \\ 0 & ;\ otherwise. \end{cases} \quad (4)$$

Similarly, word w_2 is less likely to have any values if the influence of the previous word w_1 is taken into account. If word w_2 has value k under the influence of the previous word w_1, all sentences in which the word sequence $w_1 w_2$ occur have the value k. We do not assume this always holds due to noise from the context, thus we limit the possible values for $v(w_2;w_1)$ as follows.

$$[v(w_2;w_1)]_k = \begin{cases} 0 \; or \; 1 & ; \; \frac{ns(w_1,w_2,k)}{ns(w_1,w_2)} \geq \alpha, \text{ and } ns(w_1,w_2) \geq FWW \\ 0 & ; \; otherwise, \end{cases} \qquad (5)$$

where $ns(w_1,w_2)$ is the number of sentences in which the bigram $w_1 w_2$ occurs, and $ns(w_1,w_2,k)$ is the number of sentences in the training collection in which the bigram $w_1 w_2$ occurs that are annotated value k. The meta-parameter FWW is the minimum number of sentences to make the observed bigram counts sufficiently reliable. Maximizing bitwise to compute the $maxV(w_2;w_1)$ needed for our simulated annealing algorithm yields:

$$[maxV(w_2;w_1)]_k = \begin{cases} 1 & ; \; \frac{ns(w_1,w_2,k)}{ns(w_1,w_2)} \geq \alpha, \text{ and } ns(w_2;w_1) \geq FFW \\ 0 & ; \; otherwise. \end{cases} \qquad (6)$$

4.2 The Algorithm

In this section, we describe the algorithm for constructing a values dictionary and the use of that values dictionary to estimate sentence-level values on previously unseen sentences.

4.2.1 *Values dictionary construction*

Our proposed algorithm for extracting human values is described in Figure 2. First, this algorithm starts from a reasonable assignment (the values we see most often with each word stem unigram or bigram in the annotated training sentences), $maxV(w)$ and $maxV(w_2;w_1)$, at line (1). Then the algorithm iteratively tries small variations on the values assignments until some large number of iterations have been performed (beyond which no further improvement of F_1 would be expected) in lines (2) to (18).

Each perturbation seeking more optimal values assignments in the search space for each bigram $w_1 w_2$ is done using a small probability at line (6). The same is done for each unigram w at line (16). The assignment stored in $v(w)$ or $v(w_2;w_1)$ is denoted by the status C. The new assignment (the status C') is selected by modifying only one bit for a bigram from the previous assignment by the function **gen_new_status** described in Figure 3 , which is called at line (8). The new assignment is within a neighborhood of the previous assignment by the proximate optimality property. The closed test at (7) and (9) means that calculation of F_1 is done only for training data, by estimating based on all $v(w_2;w_1)$ using formula (2), only for bigrams. At line (16), this estimation is done with all $v(w)$ for unigrams. The improvement rate ΔE between the F_1 for the current status C and the F_1 for the new status C' is calculated at line (10).

If the F_1 for C' is improved at line (11), the new assignment for $v(w_2;w_1)$ (or later for $v(w)$) is always adopted. When F_1 for C' is not improved, the new assignment can still be adopted, but in that case only with some small probability, and only if the number of values is decremented (as specified at line (15)). As line (17) specifies, the chance of modification when F_1 is not improved gradually becomes smaller. After terminating this algorithm, the values assignments stored in $v(w)$ and $v(w_2;w_1)$ (for unigrams and bigrams with one or more values) are extracted as the entries of the values dictionary.

4.2.2 Estimation of sentence-level values

Once the entries for values dictionary are extracted, estimation of sentence-level values for unseen sentences in new documents is quite simple. For each word or two-word sequence in a sentence, the word (sequence) is assigned the value(s) specified in the values dictionary. When both a word and a two-word sequence ending in that word match dictionary entries, the values assignment is calculated using equation (1). After the word-level values assignment process is finished, the sentence-level values are calculated using equation (2). This sentence-level values estimation is deterministic; therefore, it is easily explained and readily traceable.

Figure 2: Algorithm for values dictionary construction using Simulated Annealing

```
(1) Calculate the maximum possible values $maxV(w)$and $maxV(w_2;w_1)$ for all words.
(2) $TC \leftarrow 1.0$.
(3) **repeat**
(4)   **foreach** bigram $(w_1,w_2)$, apply the following
(5)     Generate a uniform random number $r_1$.
(6)     **if** ( $r_1 < smallProb$ (= 0.001) ) {
(7)       $oldF_1 \leftarrow$ closed test for the current status $C$.
(8)       $C' \leftarrow$ **gen_new_status**$(w_2;w_1)$.
(9)       $newF_1 \leftarrow$ closed test for a new status $C'$ .
(10)        $\Delta E \leftarrow 1/(1+newF_1) - 1/(1+oldF_1)$.
(11)        **if** ($\Delta E < 0$)  Set the status $C$ as $C'$.  // $newF_1$ is higher than $oldF_1$
(12)        **else** {
(13)          Generate a uniform random number $r_2$.
(14)          **if** ( $r_2 < exp(-\Delta E/TC)$)
(15)            Set the status $C'$ to $C$ if the number of values in $C'$ is fewer than one in $C$.}}
(16)  **foreach** unigram $w$, apply the same as (5)-(15).
(17)  $TC \leftarrow TC \times 0.999$.
(18) **until** loop-counter reaches $NITER$ (=100,000)
```

Figure 3: A function to generate a new status of values assignment

```
**gen_new_status**$(w_2;w_1)$:
(1) Select whether to increase or to decrease the number of the values
    for the word with equal probability (i.e. probability is 0.5, respectively).
(2) In increment,
    choose a $k$ where the $[maxV(w_2;w_1)]_k$ is 1 and $[v(w_2;w_1)]_k$ is 0 in equal probability,
    then set $[v(w_2;w_1)]_k$ to 1.
(3) In decrement, choose the a $k$ where $[v(w_2;w_1)]_k$ is 1 in equal probability,
    then set $[v(w2;w1)]_k$ to 0.
```

Note that the number of values assigned for a word must be less than or equal to 2. The values dictionary is a global variable. When this function is applied to a unigram instead of a bigram, read $(w_2;w_1)$ as w.

Table 1: Classifier effectiveness measures.

Value	Precision			Recall			F_1		
	SVM	LVM	*SA*	SVM	LVM	*SA*	SVM	LVM	*SA*
wealth	0.7859	**0.7908**	0.7135	0.6977	0.7402	**0.8048**	0.7392	**0.7646**	0.7564
s-order	**0.8235**	0.7803	0.7149	0.7587	0.8174	**0.8554**	0.7898	**0.7984**	0.7788
justice	0.7275	**0.7823**	0.6527	0.5558	0.5483	**0.5977**	0.6302	**0.6447**	0.6240
freedom	0.7461	**0.7911**	0.7211	0.6654	0.6729	**0.7151**	0.7035	**0.7272**	0.7181
innov.	**0.8139**	0.7898	0.7336	0.5629	0.5756	**0.5923**	0.6655	**0.6659**	0.6554
honor	0.4324	**0.5085**	0.4643	**0.2019**	0.0946	0.0410	**0.2753**	0.1596	0.0754
average	0.7730	**0.7849**	0.7052	0.6510	0.6737	**0.7177**	0.7068	**0.7251**	0.7114

5 Classification Accuracy

In this section, we demonstrate that our approach can be used to produce classifier results that are competitive with those achieved by state of the art techniques.

5.1 Experiment Design

We use 102-fold document-scale cross-validation. 102-fold cross-validation seeks to model the case in which some set of 101 documents have been annotated as training data and we are interested in the degree to which the machine can automatically code all future documents. To select the meta-parameters for each fold, we use 100 documents for development training and one pseudo held-out document for development testing. We perform a parameter sweep by training on all sentences in the development training set and then testing on all sentences in the one development testing document to select the meta-parameter α that yield the best F_1, sweeping the meta-parameters α across 0.5, 0.55, and 0.6; FW across 4, 5, 6; and FWW across 8, 9, 10. These ranges were determined from preliminary experiments. Once the parameter choices are made for a particular fold, training is conducted on the full 101-document training set using the best α, and the resulting model is then used to classify the sentences in the single-document test set. As baselines, we report a SVM with a second-degree polynomial kernel with unigram features and LVM (Latent "Value" Model), both of which are described by Takayama et al. [22].

5.2 Results

Table 1 shows three measures of classifier effectiveness for our proposed method *SA* (Simulated Annealing) method, and for the SVM and LVM baselines. We can see that our proposed *SA* method generally yields F_1 results comparable to those of the baselines, slightly below those of LVM and slightly above those of SVM. The exception is *honor*, on which every classifier does poorly (almost certainly because there are too few training examples). Notably, the precision achieved by LVM is typically higher than that of *SA*, while the re-call of *SA* is typically higher than that of either SVM or LVM. We would intuitively expect higher recall to be most useful to researchers who are performing qualitative analysis because people arguably would find it easier to ignore false positives in a small population than to detect false negatives in a large population.

6 Content Analysis

Strong classification accuracy results serve to indicate that the values dictionary that we have created reflects the ways in which people use language to express values to a useful degree. Our ultimate goal in creating a values dictionary is to expose to social scientists the inner workings of our system in a way that allows them to draw useful insights. For this, we must look further into the nature of our values dictionary, and to what can be learned by using that dictionary. In this section, we first provide descriptive statistics for the resulting values dictionary, and then we follow that with the results of a content analysis of bigrams.

The content of the values dictionary depends on the training set, so the resulting dictionary entries could be different for each fold in the experiment described above. For our subsequent analysis we therefore merge all resulting dictionary entries using a bitwise OR operation. Although we limit the dictionary entries for each fold to at most two human values for any unigram or bigram, this merging process can sometimes generate values dictionary entries that generate more than two values.

The number of word stem unigrams in the resulting values dictionary (i.e., the number of stems that are associated with one or more values) is 461. Of these, 428 are associated with a single value, and 33 are associated with two (or more) values. The number of word stem bigrams in the values dictionary is 360, 284 associated with a single value and 76 associated with two or more values. As is common in text processing, the actual use of these entries decays quickly, with only a small handful of the unigrams and bigrams accounting for the vast majority of the value assignments. Rather reassuringly, we found that the names Cheng [5] had given to specific values were themselves included as terms associated with those values in two cases: “innov(ation)” and “freedom.”

We have previously reported results for a thematic analysis of word stem unigrams in Fleischmann et al. [9], and here we extend that work with a content analysis of the word stem bigrams in our values dictionary using closed-coding content analysis [15] based on the set of categories that were those identified in [9]. Two authors of this paper, each experienced social scientists, examined each bigram that was associated with a specific value. The category set includes “other” and “unrelated” categories were also noted for singleton (in [9]) or infelicitous associations, respectively. The categories into which bigrams were coded are shown in Table 3.

Krippendorff’s α [15] is a way to chance-correct agreement, characterizing how much the agreement between coders differs from what would be estimated to result simply from chance. Of the 23 categories resulting from the thematic analysis in [9], 20 categories (84%) yielded enough data to analyze agreement. Using the cutoffs proposed by Krippendorff [15], strong agreement (≥ 0.8) was achieved for 7 categories (30%), and at least tentative agreement (≥ 0.67) was reached for a total of 13 categories (57%). Unsurprisingly, we were not able to achieve good agreement on the ill-defined “other” and “unrelated” categories. These results serve to illustrate one of the ways in which the type of explainable associations between language use and human values that our new method has learned can be used in social science research.

7 Conclusion

Our method, which is based on simulated annealing, has obtained classification accuracy (F_1) scores comparable to those of the strong baselines (specifically, SVM and LVM). Fur-

Table 2: Themes and Krippendorff's α for bigrams for each of the five values.

wealth		social order		justice		freedom		innovation	
Theme	α	Theme	α	Theme	α	Theme	α	Theme	α
unrelated	.49	unrelated	.54	unrelated	.20	unrelated	.37	unrelated	*
other	.28	other	.55	other	.44	other	.44	other	-.02
riches	.85	public sctr	.82	fairness	.72	choice	.73	innovator	*
poverty	.00	private sctr	.47	unfairness	.64	restriction	.93	technology	.90
buyer	.76			censorship	.67	centr	*	creativity	.81
seller	-.07			openness	.54	decentr	*	evolution	1.00
public sctr	.80					private sctr	.69	entrepreneur	.59
private sctr	.32					public sctr	.75		

*: Krippendorff's α is undefined for this variable due to invariant values.
sctr: sector, centr: centralization, decentr: decentralization

ther, we have demonstrated that a values dictionary constructed using our proposed method can be used as a basis for thematic analysis, a research method commonly used in the social sciences [2]. So far, we have only confirmed that our proposed method works well on a single collection, so in future work we plan to apply our method to other collections, on other topics, that have been annotated for human values.

Acknowledgments

This work was supported in part by NSF IIS-0725459, Japan grant-in-aid for scientific research (B) 25280118, and DARPA contract HR0011-12-C-0015.

References

[1] D.M. Blei, A.Y. Ng, and M.I. Jordan, "Latent Dirichlet Allocation," J. Machine Learning Research, Vol. 3, No. 4-5, pp. 993-1022, 2003.

[2] V. Braun and V. Clarke, "Using thematic analysis in psychology," Qualitative Research in Psychology, Vol. 3, pp. 77-101, 2006.

[3] A.-S. Cheng and K.R. Fleischmann, "Developing a meta-inventory of human values," Proc. American Society for Information Science and Technology (ASIST2010), Vol. 47, No. 1, pp. 1-10, 2010.

[4] A.-S. Cheng, K.R. Fleischmann, P. Wang, E. Ishita, and D.W. Oard, "The Role of Innovation and Wealth in the Net Neutrality Debate: A Content Analysis of Human Values in Congressional and FCC Hearings," J. American Society for Information Science and Technology, Vol. 63, No. 7, pp. 1360-1373, 2012.

[5] A.-S. Cheng, "Values in the Net neutrality debate: Applying content analysis to testimonies from public hearings," Doctoral Thesis, Univ. of Maryland, College Park, 2012.

[6] J. Cowie, J. Guthrie, and L. Guthrie, "Lexical Disambiguation using Simulated Annealing," Proc. 14th Conf. Computational linguistics (COLING '92), Vol. 1, pp. 359-365, 1992.

[7] K.R. Fleischmann: "Information and Human Values," Morgan & Claypool, 2014.

[8] K.R. Fleischmann, A.-S. Cheng, T.C. Templeton, J.A. Koepfler, D.W. Oard, J. Boyd-Graber, E. Ishita, and W.A. Wallace, "Content Analysis for Values Elicitation," Proc. ACM SIGCHI 2012 Conf. on Human Factors in Computing Systems, Workshop on Methods for Accounting for Values in Human-Centered Computing, Austin, TX, USA, 2012.

[9] K.R. Fleischmann, Y. Takayama, A.-S. Cheng, Y. Tomiura, D.W. Oard, and E. Ishita, "Thematic Analysis of Words that Invoke Values in the Net Neutrality Debate," Proc. iConference 2015, Newport Beach, CA, USA, 2015.

[10] T.L. Griffiths and M. Steyvers, "Finding Scientific Topics," Proc. National Academy of Sciences of the United States of America, Vol. 101 (Suppl. 1), pp. 5228-5235, 2004.

[11] H.-F. Hsieh, and S. Shannon, "Three Approaches to Qualitative Content Analysis," Qualitative Health Research, Vol. 15, No. 9, pp. 1277-1288, 2005.

[12] E. Ishita, D.W. Oard, K.R. Fleischmann, A.-S. Cheng, and T.C. Templeton, "Investigating Multi-Label Classification for Human Values," Proc. American Society for Information Science and Technology (ASIST2010), Vol. 47, No.1, pp. 1-4, 2010.

[13] T. Joachims, "Learning to Classify Text using Support Vector Machines," Springer Science+Business Media, New York, 2002.

[14] S. Kirkpatrick, C. D. Gelatt, and M.P. Vecchi, "Optimization by Simulated Annealing," Science, Vol. 220, pp. 671-680, 1983.

[15] K. Krippendorff, "Content Analysis, an Introduction to Its Methodology", 3rd ed., Thousand Oaks, CA: Sage Publications, 2013.

[16] B. Liu, "Opinion Mining and Sentiment Analysis," In Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data, Data-Centric Systems and Applications, pp. 459-526, Springer-Verlag Berlin Heidelberg, 2011.

[17] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, "Equation of State Calculations by Fast Computing Machines," J. Chemical Physics, Vol. 21 No. 6, pp. 1087-1092, 1953.

[18] G. Sampson, "A Stochastic Approach to Parsing," Proc. 11th Conf. Computational linguistics (COLING '86), pp. 151-155, 1986.

[19] S.H. Schwartz, "Value Orientations: Measurement, Antecedents, and Consequences across Nations," In R. Jowell, C. Roberts, R. Fitzgerald, and G. Eva eds., Measuring attitudes cross-nationally: Lessons from the European Social Survey, London, England: Sage, 2007.

[20] F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, Vol. 34, No. 1, pp. 1-47, 2002.

[21] Y. Takayama, Y. Tomiura, E. Ishita, Z. Wang, D.W. Oard, K.R. Fleischmann, and A.-S. Cheng, "Improving Automatic Sentence-Level Annotation of Human Values Using Augmented Feature Vectors," Proc. Conf. Pacific Association for Computational Linguistics (PACLING 2013), Tokyo, Japan, Sept. 2013.

[22] Y. Takayama, Y. Tomiura, E. Ishita, D.W. Oard, K.R. Fleischmann, and A.-S. Cheng, "A Word-Scale Probabilistic Latent Variable Model for Detecting Human Values," Proc. 23rd ACM Int'l Conf. Information and Knowledge Management (CIKM 2014), Shanghai, China, Nov. 2014.

[23] Y.R. Tausczik, and J.W. Pennebaker, "The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods." J. Language and Social Psychology, Vol. 29, No. 1, pp. 24-54, 2010.

[24] E. Woodrum, "Mainstreaming Content Analysis in Social Science: Methodological Ad vantages, Obstacles, and Solutions," Social Science Research, Vol. 13, pp. 1-19, 1984.