MIS Quarterly

METHODS ARTICLE

A TOOL FOR ADDRESSING CONSTRUCT IDENTITY IN LITERATURE REVIEWS AND META-ANALYSES

Kai R. Larsen
Leeds School of Business, University of Colorado, Boulder, 995 Regent Drive,
Boulder, CO 80309-0419 U.S.A. {kai.larsen@colorado.edu}

Chih How Bong
Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak,
94300 Kota Samarahan, Sarawak, MALAYSIA {chbong@unimas.my}

Appendix A

Natural Language Processing Designs

This appendix provides background details on the NLP designs used in our paper. For each, the steps necessary for implementation are outlined.

Latent Semantic Analysis

The underlying idea of latent semantic analysis (LSA) (Deerwester et al. 1990) is that the aggregate of all of the contexts in which a given word does and does not appear provides a set of constraints that determines the similarity of meanings for words, and sets of words, to each other (Landauer et al. 1998). Thus, when two terms occur in contexts of similar meaning, even if they never occur in the same passage, LSA represents them as having similar meanings.

LSA Steps

Step 1. Preparing term-document matrix. For the LSA process, we employed minimal pre-processing. Starting with the paragraphs from the *MIS Quarterly* and *Information Systems Research* articles that contained the sample of constructs, we removed non-alphanumeric characters and converted all words to lower case; we then stemmed the words using the Porter (1980) algorithm and weighted the term-document matrix using log-entropy and normalization.

Step 2. Creating semantic space. After weighting and normalization, the matrix A is decomposed using singular value decomposition (SVD), a mathematical algorithm similar to a factor analysis, with the result being a *semantic space* containing term vectors, each infused with an "understanding" of the term.

Step 3. Projecting items into the semantic space. Given the query q, which is a construct item, a query vector $\vec{q}$ is obtained through an aggregation of term vectors relevant to the item. In our research, we project every item into the semantic space as a query vector $\vec{q}$, and that vector is saved as $\vec{q}_n$ for future item-item analysis, where n is the total number of items. This improves speed of the analytics and enables separately stored solutions for search-engine purposes.

Step 4. Finding similar items. To find similar items to $\vec{q}$, the query vector is compared against all the items stored inside the semantic space, $\vec{q}_n$, using the cosine similarity measurement. We deem vectors in $\vec{q}_n$ that yield the highest cosine scores corresponding to $\vec{q}$ to be more semantically relevant; we deem those with low cosine scores to be relatively irrelevant.

Semantic Space Selection

"The selection of dimensionality in LSA is fraught with problems" (Kakkonen et al. 2008, p. 280), and in fact, researchers have yet to find a generalizable solution to dimensionality selection (see, e.g., Bingham and Mannila 2001; Globerson and Tishby 2003; Landauer et al. 1998). In IS, this issue was prominently featured in a very thorough LSA exposition by Siderova et al. (2008), where domain-specific 5-, 13-, and 100-dimensional solutions were proposed and analyzed. While research has suggested that 300-dimensional solutions work well (Landauer and Dumais 1997), that result was not based in theory and is not likely to generalize perfectly into the psychometric domain. The purpose of our article is to show that natural language processing works to address the construct correspondence problem rather than to fully optimize such designs. This work was simply conducted using a 300-dimensional solution. Our own examinations have shown that when used for construct items, LSA semantic spaces between 200 and 500 are roughly equivalent, but higher dimensions do marginally better (although with associated increases in processing time).

Latent Dirichlet Allocation

We translated all words in each item into k topical probabilities according to the posterior distribution. Hence, each item was depicted as a k-dimensional vector representing its topic distribution. To compute the similarity of the items, we computed the cosine of the item vector. The following list of steps illustrates the process.

Step 1. Building the LDA model. We used the same paragraphs used for LSA to build the LDA model.

Step 2. Preprocessing. We preprocessed paragraphs by removing stop words,[1] punctuation, and non-alphabet characters.

Step 3. Computing topic distribution. An advantage of online LDA is its ability (1) to process the whole document collection in a single, full pass; (2) to converge rapidly; and (3) to yield an accurate topic estimation. In this step, we defined k to present the number of topics. We set k to 300 after examination of values from 50 to 500 in increments of 50. That examination follows in the next section.

Step 4. Building an item subspace. Once the LDA model was ready, we projected each item into the model and transformed it into the k-topic distribution format. Similar to LSA, this represented the model subspace for all of the items.

Step 5. Finding similar items. It is our assumption that two items are semantically similar if they share identical topic distribution. Thus, we defined the similarity as the angle cosine of the two item vectors in the subspace.

Topical Probabilities

We evaluated 10 LDA designs using different numbers of topical probabilities, where k = (200, 250, 300, 350, 400, 450, 500). Improvements are apparent when increasing k from 50 to 300, with the greatest improvements seen in the change from 150 to 200. However, increasing k beyond 300 yielded only insignificant improvements along with longer convergence time. A close examination revealed that the k values around 300 were able to transform marginal false negatives to true positives and eliminate a high number of false positives. LDA should work well theoretically for the problem at hand, but the nature of the texts used to measure a construct (the construct measurement items) is quite different from the majority of paragraphs in an academic paper. In addition, we are aware of no other research that uses LDA in quite this way. Unlike LSA, the representation of items in LDA is not able to take into account synonymy effectively. While LDA may be better fit for detecting latent topics, our experiments show that LDA did not perform well for establishing similarity. One of the reasons for this discrepancy is the representation of topic distribution, as the short vectors may diminish its performance during calculation of similarity. For example, LDA similarity for "task" and "job" yielded 0.00, whereas in LSA, it yielded over 0.9.

[1]We used the University of Glasgow stop word list (http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words). A pdf copy of this list is available from the authors for archival purposes.

Knowledge-Based Approaches: LI

The semantic similarity of item pairs is computed by taking the path length and the depth of two words in a hierarchical semantic knowledge base, i.e., WordNet (Miller 1995; Poli et al. 2010). WordNet is a hierarchical lexical reference system consisting of "English nouns, verbs, and adjectives…organized into synonym sets, each representing one underlying lexical concept" (Miller et al. 1993, p. 235). Later work added adjectives to WordNet. LI relies on WordNet to measure word similarity because it is one of the richest and most accurate lexical dictionaries ever crafted, correlates well with human judges (Miller 1995), is readily available, and does not adhere to a specific domain.

Word-order similarity is factored in, turning each sentence into a vector by ordering the words as they appear and computing the difference of the word orders. Finally, in order to separate the informative words from those that are not, information content of each word is normalized onto each similarity score from the Brown University Standard Corpus of Present-Day American English (Marcus et al. 1993).

Knowledge-Based Approaches: MI

Building on WordNet, MI's (Mihalcea et al. 2006) word specificity refers to the specific meaning of words (e.g., collie and sheepdog) versus generic concept words (e.g., animal and mammal). The similarity measure gives higher weight to precise, specific-meaning words than to abstract, generic-concept words. Word specificity is computed using the inverse document frequency (IDF) algorithm (Jones 1986), which was applied to the British National Corpus. IDF assumes that rare words have greater discriminatory weight than common words. Additionally, IDF similarity scores are normalized to reflect word specificity in the sentence. MI averages word-similarity metrics from six different sources: Jiang and Conrath (1997), Leacock and Chodorow (1998), Lesk (1986), Lin (1998), Resnik (1995), and Wu and Palmer (1994). These metrics were created to measure word relatedness and similarity by calculating the shortest distance between given words' synsets (sets of synonymous words) in the WordNet hierarchy; the shorter the distance between words, the higher the similarity score.

MI was designed to measure short text similarity and has been used at the sentence level by Mihalcea et al. (2006). Like LI, the MI similarity score is a number between 0 and 1, where 0 indicates no semantic overlap and 1 indicates exact match. The MI sentence similarity measure is computed for two candidate sentences, S_1 and S_2 as follows:

Step 1. Identifying part-of-speech (POS). The process begins with tokenization and part-of-speech tagging of all the words in the sentence into their respective word classes (noun, verb, adverb, and adjective, as well as cardinal numbers).

Step 2. Calculating word similarity. Each word in the sentence is evaluated against all the words from the other sentence to find the highest semantic similarity (*maxSim*) from the six word similarity metrics: Jiang and Conrath (1997), Leacock and Chodorow (1998), Lesk (1986), Lin (1998), Resnik (1995), and Wu and Palmer (1994). These metrics were originally created to measure concept likeness rather than word likeness, but they are adapted in our study to compute word similarity, by computing the shortest distance between given words' synsets in the WordNet hierarchy. The word–word similarity is computed only on the words from the same word class, which are either from noun or verb word classes, because WordNet contains separate semantic trees for nouns and verbs; it is thus not possible to obtain similarity between nouns and verbs using WordNet distance. For other word classes such as adverb, adjective, cardinal, and unknown words, whole word matching is used instead. The word-word similarity measure is directional. It begins with each word in S_1 being computed against each word in S_2 and then vice versa.

Step 3. Calculating sentence similarity. Once the highest semantic similarity (*maxSim*) for each word in the sentences is computed, it is normalized with inverse document frequency (IDF) to weight rare and common terms. The normalized scores are then totaled for a sentence similarity score, Sim_{MI}, as follows:

$$Sim_{MI}(S_1, S_2) = \frac{1}{2} \times \left(\frac{\sum_{(w \in S_1)} maxSim(w, S_2) \times IDF(w)}{\sum_{(w \in S_1)} IDF(w)} + \frac{\sum_{(w \in S_2)} maxSim(w, S_1) \times IDF(w)}{\sum_{(w \in S_2)} IDF(w)} \right) \quad (1)$$

where $maxSim(w, S_2)$ is the score of the most similar word in S_2 to w and IDF (w) is the inverse document frequency of word w.

The CID Hybrid Approaches: The CID$_1$ Model

The process steps for calculating CID$_1$ similarities of two given items are as follows:

Step 1. Forming both items into a joint text. The comparing item and the joint item are formed into a matrix to allow each word from different items to be compared. Table A1 shows that process for the items, "RAM keeps things" and "the CPU uses RAM." The word-word similarity here is computed using LSA cosine similarity scores.

Step 2. Identifying the maximum similarity score. For each column (the columns represent the element of the lexical word vector), the maximum similarity score is identified, which, in turn, is normalized using information content from the Brown Corpus.

Step 3. Forming a lexical word vector for the first item. Only the similarity scores above the preset threshold of 0.2 (developed for LI) are selected to form a lexical word vector. Scores below the threshold are set to zero, based on empirical findings that such words are too dissimilar to score (Li et al. 2006). In this study, we did not test whether a different threshold from LI's 0.2 would work better for CID$_1$. Extensive differences between the LI and LSA similarity measures suggest that testing alternative thresholds might improve the CID$_1$ algorithm, but we did not expect it to change the outcome of the evaluations. Thus the lexical word vector for the first item is $\vec{S}_1$ = [0.39, 0.33, 0.179, 0.00, 0.074, 0.23].

Step 4. Forming a lexical word vector for the second item. The step is identical to step 3, but using Table A2. This yields $\vec{S}_2$ = [0.19, 0.00, 0.16, 0.03, 0.389, 0.04].

Step 5: Deriving the word-order vectors using joint word. The word-order vectors are also derived using joint word, such that words in the vector are assigned with unique indices according to the order in which they appear in candidate items. For example, the word-order vector for the joint words "*RAM keeps things The CPU uses* " is [1,2,3,4,5,6].

To derive the word-order vector for "RAM keeps things," we assign the word-order index to the corresponding words in the candidate item. The first word "RAM" has an index of 1, and "keeps" has 2, etc. However, since the item does not have the words "The," "CPU," and "uses," the index will be the most similar word from the index that is computed in Step 2. For example, according to Table D1, "The" is similar to no other word and thus it has an index value of 0, whereas "CPU" and "uses" are most similar to "things," which has an index of 3. Thus the word-order vector for the first item is $\vec{O}_1 = [1, 2, 3, 0, 3, 3]$. The same process applied to the second item yields, $\vec{O}_2 = [1,0,3,4,5,6]$. The minimum similarity threshold also applies here. For any word that has similarity less than the preset threshold, the value is zero. Likewise, the other comparing item is also derived using the same approach.

Step 6. Computing the similarity of word-order information to distinguish the meaning of two items attributed to syntactic differences. The word-order similarity score is then computed as follows:

$$Sim_{order}(S_1, S_2) = 1 - \frac{\|\vec{O}_1 - \vec{O}_2\|}{|\vec{O}_1 + \vec{O}_2\|} \tag{2}$$

Step 7. Obtaining items' similarity score. Similarity scores are obtained by calculating the cosine angle of the lexical word vectors,

$$Sim_{semantic}(S_1, S_2) = \frac{\vec{S}_1 . \vec{S}_2}{\|\vec{S}_1\| . \|\vec{S}_2\|} \tag{3}$$

Step 8. Calculating overall sentence similarity score. Finally, the overall item similarity is defined as the combination of lexical semantic similarity and word-order similarity, which are operationalized in in the following:

$$Sim_{Li}(S_1, S_2) = \gamma\, Sim_{semantic}(S_1, S_2) + (1-\gamma) Sim_{order} \tag{4}$$

where γ is a relative contribution of semantic similarity and word-order similarity to the overall similarity. According to Li et al, γ is a value greater than 0.5.

Table A1. Deriving Semantic Vector for Item 1 (S_1)

S_1	RAM	keeps	things	the	CPU	uses
RAM	1	0.05	0	0	0	0
keeps	0.15	1	0	0	0	0
things	0.33	0.39	1	0	0.34	0.83
Sim	1	1	1	0	0.34	0.83
Weight	I(RAM) I(RAM)	I(keeps) I(keeps)	I(things) I(things)		I(CPU) I(things)	I(uses) I(things)
$\vec{S}_1$	0.390	0.330	0.179	0	0.074	0.23

Table A2. Deriving Semantic Vector for Item 2 (S_2)

S_1	RAM	keeps	things	the	CPU	uses
The	0	0	0	1	0	0
CPU	0	0	0	0	1	0
uses	0	0	0.25	0	0	1
RAM	1	0	0	0	0.33	0
Sim	1	0	0.25	1	1	1
Weight	I(RAM) I(RAM)		I(things) I(uses)	I(The) I(The)	I(CPU) I(RAM)	I(uses) I(uses)
$\vec{S}_2$	0.190	0	0.16	0.03	0.389	0.04

The CID Hybrid Approaches: The CID_2 Model

While based on the MI sentence-similarity algorithm, CID_2 does not perform part-of-speech (POS) tagging, and word similarity is directly calculated using LSA. Similarity scores are then normalized with inverse document frequency (IDF) to reflect rare and common terms (Mihalcea et al. 2006). The following steps illustrate the sentence-similarity score process developed for CID_2.

Step 1. Computing word similarity using LSA. The process begins directly with word-word similarity. Given two words, the cosine similarity is calculated based on the LSA semantic space.

Step 2. Computing sentence similarity. Once the word-similarity scores are obtained for each sentence, they are normalized with correspondent IDF, and the sentence-similarity scores are computed by replacing the original equation with the LSA similarity measure, which yields the following equation for CID_2:

$$CID2(S_1,S_2)=\frac{1}{2}\times\frac{\sum_{(w\in S_1)} Sim\_LSA(w,S_2)\times IDF(w)}{\sum_{(w\in S_1)} IDF(w)}+\frac{\sum_{(w\in S_2)} Sim\_LSA(w,S_1)\times IDF(w)}{\sum_{(w\in S_2)} IDF(w)} \qquad (5)$$

While we had expected CID_2 to outperform LSA, careful consideration of the evidence suggests a good reason for CID_2's performance. We examined a number of variables using two separate linear regressions on the relative ranking of item pairs for CID_2 versus MI, split on whether the item pairs were correctly or incorrectly classified. In both analyses, the max(IDF) for each item pair emerged as the most important negative influence on CID_2 design performance. In a separate experiment, we removed the IDF weighting from CID_2 and compared its performance against all designs on a sample set of item pairs, which resulted in CID_2 moving up from the bottom to the third best performer (just behind CID_1 and LSA, respectively). Detailed examination also indicated that the weighting of IDF in CID_2 did not scale well with LSA (between common and informative words) because the geometrical scaling of word occurrences is less meaningful in LSA; the virtue of its similarity scores is maneuvered through algebraic operations in the semantic space. CID_2 was not included in the final CID tool in lieu of LI, because LI serves the function of providing a WordNet knowledge-based design for the final tool, which is important for future contexts of use.

Appendix B

Overview of Gold Standard

Data collection was performed by a team assembled for a larger project funded by the National Science Foundation. The team consisted of advanced undergraduate and master's level graduate students at a large, research university in the western United States. Students were selected for interviews based on previous research experience, GPA, academic awards, and interest in research. Selected applicants received a two-hour training session. This training was followed by an hour-long construct-extraction task consisting of the extraction of all variables, definitions, and items from an academic article. Student performance on the task was then graded against a gold standard of previously extracted variables, definitions, and items, and validated by our Chief Senior Research Assistant. The top scorers were hired as research assistants (RAs), resulting in the top 3 percent of applicants accepting positions. Hired RAs received an additional 10 hours of training along with careful monitoring of their first 100 hours of extraction. After 200 to 300 hours of extraction experience, the best 20 to 30 percent of RAs were promoted to senior RAs. All articles were first extracted by a junior or senior RA and then audited by a senior RA.

Categorization Process

Categorization was carried out by a research team, which consisted of one experienced faculty researcher, three doctoral students, and four experienced research assistants (RAs with at least 500 hours of experience in construct extraction).

Overall, the categorization task was divided into two stages: rough categorization and refined categorization. Rough categorization involved grouping constructs with highly similar properties into the same cluster to make the task cognitively tractable. The refined categorization process resulted in hierarchies, sub-hierarchies, and categories of the groups generated by rough categorization. The refined categorization process began by labeling the hierarchical clusters. The taxonomy in this study was restricted to two hierarchical levels. For example, the main category, "Trust," had the subcategories, "Trust in Benevolence," "Credibility," and "General Trust." The hierarchical structure was used solely to reduce the cognitive load on the participants. Ultimately, the only important consideration was whether a pair of constructs should be placed in the same category (correspondence) or different categories (independence).

Once each construct was assigned to one of the categories, annotator teams took turns examining all categories to ensure that constructs had similar contexts and fit the definition of correspondence. Some clusters—usually those with a high number of constructs—required several examinations by different teams. Each category was examined by at least two teams, with the team containing the faculty member always doing the final examination. If any team did not fully agree with the previous team, that team would reorder and move constructs, and a third team would reexamine the category. Constructs were assigned only to one category. Both the rough and refined categorization tasks were carried out in stages across six days, and they consumed about 200 person-hours.

Evaluation

Finally, in recognition of the complexity of constructing a taxonomy for such an intricate dataset, two researchers, having a combined 15+ years of research experience and no exposure to the taxonomy process, were chosen to evaluate the dataset. The experts were given a semi-randomized set of 300 construct pairs, where each pair had a 0.5 probability of being either correspondent or independent. The inter-rater agreements of correspondence versus independence between the two experts and the taxonomy were at 85 percent and 90 percent. The resulting Cohen's (1960) kappas were 0.68 and 0.79, indicating agreement levels that are considered "substantial" and close to "almost perfect" by Landis and Koch (1977).

The Taxonomy

We share this taxonomy with the caveat that it represents only a partial view of constructs in the two journals *MIS Quarterly* and *Information Systems Research*. It consists of constructs reported with a minimum of three items during the period from 1983 to 2009. The taxonomy is irrelevant to the construct-identity-detection designs in this article, as the designs operated at the lowest level of the taxonomy and did not consider the hierarchical structure of the taxonomy. However, the hierarchical structure was invaluable in reducing the cognitive strain of the categorization exercise reported in the evaluation section.

In the case of the present taxonomy, a bottom-up process was employed, which tends to guide the taxonomy toward the larger topics. In other words, initial groups such as *adoption* and *development* will tend to grow faster and attract constructs earlier, before other established groups have a chance to spawn on their own. These originally smaller groups will become even smaller hierarchies as their remaining constructs are organized. For example, constructs related to top management involvement, which were categorized into the *IS development* hierarchy, may have ended up in the *leadership* hierarchy in other categorization processes where such seemingly related constructs as leader/subordinate relationships exists. Further, the lean hierarchy named *academic* could have attracted more constructs under other circumstances as well as a larger hierarchical structure if a classification exercise had been undertaken with a preexisting top-level structure. Nevertheless, any topic of reasonable size is likely to be represented as a hierarchy, and the hierarchies and their construct categories represent a potentially useful step toward the development of an ontology in which such construct categories would need to be recreated with links to related construct categories. With these caveats, we describe here the taxonomy of IS constructs created for this project.

By size, the hierarchies of the taxonomy are *IT adoption* (412 constructs), *IS development* (102), *trust* (63), *information/data* (48), *task/job* (45), *interorganizational* (43), *IS function* (43), *communications* (40), *organizational* (39), *learning* (23), *purchase* (22), *group* (19), *knowledge* (17), *judgment and decision making* (13), *leadership* (10), *strategy* (10), *general psychology* (7), *ethics/morals* (6), and *academics* (4).

That the IS field is focused on *IS adoption* (41%)—perhaps unsurprisingly to anyone involved in IS research—is among the most immediate findings. Within the adoption hierarchy, the largest sub-hierarchy by far is the one we termed *technology factors*, constructs that focus on user perceptions about technology features such as *ease of use* and *usefulness*. The second largest sub-hierarchy, which could have fit in another taxonomy under *technology factors,* was named *affective factors,* which focuses on user affect and attitude, including immersion in a technology. The third largest, *use factors*, focuses on constructs related to amount and extent of use, intended or self-reported. The last sub-hierarchies, *social/external factors* and *efficacy factors*, round out the hierarchy with a focus on facilitating conditions and technology self-efficacy.

The second largest hierarchy, *IS development* (10.2%), focuses on the process of developing information systems, primarily *participation/support*, which is about (1) the perceived level of user participation during the process and (2) (top-)management involvement in the same. *Process methodology* includes methodology, along with risk factors such as requirement-focused constructs. Less obvious is the focus on *trust* (6.3%), perhaps suggesting a level of uncertainty or even insecurity about the interface between technology and users, as well as between technology users and other individuals/organizations mediated through technology. The *information/data* (4.7%) hierarchy speaks to a focus on the data and their transformation process, with quality and understanding representing key foci. Other hierarchies of interest included *communication*, *privacy*, *learning,* and *purchase*. Smaller hierarchies are not discussed in this analysis.

What stands out in this gold standard is the great variety in level of analysis within the construct set (Burton-Jones and Gallivan 2007). The majority of research is plainly at the individual level (*IS adoption* being a typical example), but we found research directed at the *task/job* (4.5%), *group* (1.9%), *IS function* (4.3%), *organization* (3.9%), and *interorganizational (4.3%)* levels of analysis. In Table B1, we share the whole set of hierarchies. Readers are warned against the spurious belief that construct items and definitions of constructs are predictable based on construct, category, sub-hierarchy, and hierarchy names alone.

Table B1. The Gold Standard Taxonomy

Hierarchy	Sub-hierarchy	Category	# Constructs
Academic		Masculinity/femininity values	1
		Revising activities	1
		Writing importance for tenure/promotion	2
Communications	Amount of communications	Individual Information quality	2
	Channel characteristics	Communication channel characteristics	4
		Virtual co-presence	1
	Communication knowledge	Knowledge sharing to/with groups outside of the organization	6
		Knowledge sharing obligation	1
		Loss of knowledge power	1
	Communications psychology	Conflict	4
		Self-worth from knowledge sharing	1
	Communications relationship	Anticipated reciprocal relationships through knowledge sharing	1
		Expectation of reciprocity in knowledge sharing	1
		Shared understanding of role within organization	2
		Strength of ties	1
	Communication satisfaction/rewards	Economic incentive	1
	Communications quality	Communication quality	7
		Interaction quality	1
		Quality of communication interface	1
	Etc.	Communication ease	1
		Communication seeking and reception	1
		Experimental similarity	1
		Governmental contention	1
		Vertical coordination	1
Ethics/morals		Ethical behavior	1
		Internet ecology	1
		Moral intention	3
		Predisposition to justice	1
General psychology	Negative	Negative affectivity	2
		Trait anxiety	1
	Neutral	Effort to change mental model	1
		Long-term orientation	1
	Positive	Need for cognition	1
		Self-monitoring	1
Group		Control culture	1
		Dynamic capabilities	1
		Employee versus job orientation culture	2
		Group cohesion	7
		Inclusivity culture	1
		Innovative culture	2
		Normative versus pragmatic work norms	1
		Perceived outcome quality	1
		Relational capital	1
		Satisfaction with group	1
		Team member accountability	1

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
Information/Data		Accessibility	1
		Argument quality persuasiveness	2
		Compatibility	4
		Completeness of information and breadth of information exchange	4
		Currency and availability of information	2
		Data collection	1
		Information quality	9
		Information quality satisfactory	3
		Information quality specific yet broad satisfaction	3
		Information quality importance	2
		Information reliability	4
		Information reliability importance	1
		Information that meets needs	2
		Information understanding	5
		Information usefulness	1
		Locatability	2
		Understandability and reliability	2
Interorganizational	Extent/strength of collaboration/relationship	Collective sanctions	1
		Cooperative norms expectations to work together	1
		Degree of IS outsourcing	2
		Interorganizational process integration	6
		Mutual adoption	1
		Supplier responsiveness to customer needs	2
	Market factors	Asset specificity	3
		Contracting flexibility	1
		Coordination costs between firm & vendor	2
		Cost/benefit of external service provider	1
		Exchange safeguards	1
		Interorganizational relationship standards	1
		Interorganizational relationship efficiency	1
		Network effectiveness	1
		Supplier choice	1
		Transaction cost	1
	Psychological factors	Motivation to comply in general	1
		Persistent expectation of subordinance due to previous contract	1
		Relationship satisfaction	1
	Success	Effectiveness of product development work unit	1
		Relationship performance benefits from interaction	2
	Task/process factors	Interorganizational process modularity	1
		Trading partner readiness	1
	Etc.	Customer obligation	1
		Interorganizational pressures to adopt technology	2
		Obligation for accurate project scoping	1
		Obligation for building effective team	1
		Obligation for clear authority structure	1
		Obligation for taking charge	1
		Relationship management skills	1
		Restricted network access for strategic purposes	1

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
IS development	Participation/support	Perceived user participation	10
		Breadth of involvement in planning	1
		Client formal control over vendor	1
		Desire to participate in process/group	1
		Functional capability of the application service provider	1
		Management involvement in planning	7
		Obligation to ensure top management support	1
		Organizational support	1
		Perceived provider performance	1
		Planning cooperation	1
		Technical service guarantees	1
		Top management involvement - quality policy and goals	1
		Top management technology attitudes	3
	Process methodology	Amount of SDM use	2
		Barriers to implementing the methodology	1
		Frequency of use of coordination technology	1
		Methodology provides an effective planning process SDM	5
		Methodology provides a high-quality end product SDM	4
		Process standardization	3
		Quality of interactions	1
		Rewards for reuse	1
		Software development methodology type	2
	Risk factors	Performance estimation risk process	3
		Requirements instability	2
		Requirements unanalyzability	2
		Requirements uncertainty	1
		Requirement understanding	1
		Risk factors due to task size and team member's experience	1
		Risk perception	1
		Technology uncertainty	1
	Etc.	Absorptive capacity	2
		Code reusability	2
		Competitive advantage of development team	1
		Group coordination	1
		IS project success and performance	11
		Obligation to communicate clearly	1
		Open source software beliefs	1
		Open source software values	1
		Planning analysis	1
		Product/service portfolio differentiation/competitiveness in the market	1
		Project resource availability	4
		Staff skills team and leadership - knowledge & expertise	4
		System outcome	4
		Technology: customized versus canned/generic	1
		Technology evaluation competence	1
		User-IS function relationship	4

Table B1. The Gold Standard Taxonomy (Continued)			
Hierarchy	Sub-hierarchy	Category	# Constructs
IT adoption	Affective factors	Affect towards technology use	28
		Attitude towards technology usefulness	7
		Computer alienation	5
		Computer playfulness	4
		Focused immersion	6
		Importance of entertainment	1
		Personal Innovativeness	4
		Result demonstrability	4
		Technology escapism	1
		Technology invasion	1
		Technology loyalty	2
		Technology overload	1
		Technology visibility	2
	Efficacy factors	Computer anxiety	4
		Effort requirement	2
		Knowledge self-efficacy	1
		Lack of adequate skills meaninglessness	1
		Perceived control	4
		Technology-complexity inadequacy	2
		Technology knowledge	3
		Technology self-efficacy	12
	Social/ external factors	Absorptive Capacity	1
		Facilitating conditions	23
		Image	7
		Mimetic pressures	1
		Normative pressure	3
		Readiness of suppliers to do e-business	1
		Social influence	10
	Technology factors	Cognitive challenge of technology	1
		Ease of use	46
		E-shopping versus in-store shopping similarity	1
		Expectations from use of technology	5
		Flexibility, Organization	2
		IT support for contextualization	1
		IT performance	2
		Perceived monetary value	1
		Satisfaction with technology	15
		Social presence of technology	2
		System reliability	3
		Technology availability/accessibility	2
		Technology compatibility with prior experience	2
		Technology diagnosticity	5
		Technology encourages innovation	2
		Flexibility, Individual	4
		Technology functionality perceived	7
		Technology speed	4
		Technology quality	1
		Technology typicality	1

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
IT adoption (continued)	Technology factors (continued)	Trialability	3
		Usefulness to individual	67
		Usefulness to organization	12
		Visual appeal	5
	Use factors	Attitude toward IT	1
		Consensus on appropriation	1
		Faithfulness of appropriation	2
		Habit	1
		Individual use	1
		Intention to use	29
		IT use for specific purposes	7
		Resistance to use	1
		Search scope	2
		Self-reported use	5
		Switching costs	3
		Voluntariness	2
	Etc.	Convenience and confidence	2
		Importance of ease of use	1
		Importance of technology quality	1
		Organizational adoption enabling factors	2
		Technology compatibility with work style & morals/values	10
		Technology confirmation	5
		Time resources/constraints	1
IS function		Adequacy, quality, and amount of support	1
		Complexity evaluation of decision/process	1
		Departmental shared knowledge	1
		Features that determine service quality	2
		Importance of IS department attributes	3
		Importance of IT use in meeting organizational objectives	1
		IS strategic alignment	2
		IT strategy process uniqueness	1
		Mutual influence	1
		Organizational IT capability	7
		Organizational systems integration	2
		Process alignment	2
		Product design maturity	1
		Service quality: Empathy	5
		Service quality: General	3
		Service quality: Reliability	4
		Service quality: Responsiveness	4
		Technology standards	1
		Technology uncertainty	1
Judgement and decision making		Satisfaction with brainstorming outcome	3
		Satisfaction with decision making outcome	8
		Team performance	2

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
Knowledge		Diversity of knowledge	1
		Importance of IS knowledge/skills in business	2
		Importance of IS knowledge in technology	4
		Interorganizational knowledge creation	2
		IS/IT management knowledge of business practices	4
		Knowledge of access to knowledge	1
		Knowledge of team's abilities	1
		Top management IT/IS knowledge	2
Leadership		Importance of leadership skills	2
		Leader effectiveness	1
		Leader/subordinate relationship	1
		Organizational skills	1
		Personal responsibility for leadership	3
		Power	1
		Presence of leadership skills	1
Learning		Attitude toward knowledge sharing	1
		Course enjoyment	1
		Group work contributes to learning	2
		Learning frequency	1
		Learning orientation	1
		Motivation to learn	2
		Observational learning process	3
		Organizational learning	1
		Perceived learning	6
		Relative advantage	1
		Self-efficacy for doing school work	1
		Self-regulation during learning	1
		Skill development	1
		Subjective norm toward knowledge sharing	1
Organizational	Financials	Business performance	3
		Task non-contractability quality	2
	Money	Market orientation	1
	Organization-level characteristics	Business-process specificity: Human capital	2
		Business-process specificity: Intellectual capital	3
		Formalization	2
	People	Impact of system on organization	1
		Interactions between CIO and TMT	2
		IT hiring and retainment practices	1
	Perceptions of organization	Ambiguity	1
		Benefits uncertainty	1
		Fairness of labor division	1
		Information need fulfillment	1
		Norm of cooperation	1
		Organizational commitment	4
		Organizational uncertainty: Market turbulence	4
		Organizational uncertainty: Technological uncertainty	1
		Organizational uncertainty: Production volume	1
		Perceived competitive advantage	4

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
Organizational (continued)	Perceptions of organization (continued)	Perceived strategic risks	1
		Perceived supportive affect within work groups	1
		Work environment satisfaction	1
Privacy		Benefit overriding privacy concerns	2
		Concern over unauthorized access	3
		Concern over unauthorized secondary use	2
		Errors in private data/info	3
		General concerns about information privacy	16
		Importance of transparency info use policies	2
		Perceived protection by companies	2
		Personal control of information	2
		Risk of loss	2
		Technology security risk	2
		User awareness of security policies	3
Purchase		Buyer's intention to purchase	2
		Consumer's opinions of traveling to make a purchase	1
		Importance of shopping convenience	1
		Internet customer relation	2
		Monetary resources	1
		Need for uniqueness from purchasing	1
		Perceived product choice on the internet	1
		Product value	2
		Purchasing skills	1
		Satisfaction with purchasing experience	6
		Seller's past performances	3
		Societal benefit from complaining	1
Strategy		Business-process strategic criticality	2
		Locus of authority for strategic planning	1
		Strategic aggressiveness	1
		Strategic defensiveness	1
		Strategic intent	1
		Strategic investment rationale	1
		Strategy comprehensiveness	1
		Strategy proactiveness	1
		Strategy: Risk aversion	1
Task/job	Employment	Job turnover intention	3
	Job specifics	Fairness	2
		Improved job performance due to autonomy	2
		Job overload	9
	Outcomes/Performances/ Rewards	Benefit of efforts	1
		Determination of rewards	1
	Satisfaction	Individual rewards present	1
		Job satisfaction	2
		Rewards	2
	Task characteristics	Business process modularity	1
		Job autonomy	3
		Role ambiguity	2
		Role/task conflicts	3

Table B1. The Gold Standard Taxonomy (Continued)

Hierarchy	Sub-hierarchy	Category	# Constructs
Task/job (continued)	Task characteristics (continued)	Task difficulty	1
		Task interdependence	1
		Task uncertainty	1
		Task variety	4
		Uncertainty with information	1
	Etc.	Emotion toward customer	2
		Work-family conflict	3
Trust		Credibility Trust in ability and benevolence	9
		Individual-level trust: Other's ability	1
		Individual-level trust: Other's ability and trustworthiness	3
		Individual-level trust: Other's Benevolence	1
		Individual-level trust: Structural assurances	5
		Individual-level trust: Trust propensity	9
		Individual-level trust: Trustworthiness	19
		Organization-level trust: Other's ability	2
		Organization-level trust: Other's benevolence	2
		Organization-level trust: Other's integrity	2
		Organization-level trust: Reputation	2
		Trust in others' ability	8

Appendix C

Item to Construct Solutions

This appendix begins by examining approaches to the transformation of similarity scores from the item level to the construct level. It then discusses how such transformations may affect the analysis of formative versus reflective items.

Generalized Functions for Transforming Similarity Scores from the Item Level to the Construct Level

Regardless of the designs used, construct relationships can be predicted by comparing two items, where each construct item is treated as natural text and a similarity score is produced to indicate their degree of similarity. Item similarities are computed for all inter-construct item pairs, and the item scores are transformed to indicate a construct relationship. Two functions are formulated to subsume item similarities into the construct relationship. While future work should examine all possibilities, the first measure (*sec*) tested for this article used the second highest score among all inter-construct item relationships. This function is based on the notion that the two constructs' highest inter-item relationship can be misleading, while the use of the second highest relationship score results in a compromise between the rate of false positives and false negatives. The second function used to represent the construct similarity (*avg*) takes the average of the similarity of the most similar item pair scores. This function tends to compensate for item pairs that are constituted of both highly similar and very dissimilar constituent parts. Although we examined both measures, we anticipated that *avg* could better represent the construct relationship without the skewedness demonstrated by *sec*.

An analysis evaluating *sec* versus *avg* for LSA showed that both generalized functions were above the diagonal line (> 0.5), indicating that both functions *sec* (AUC = 0.759) and *avg* (AUC = 0.767) used after LSA performed better than chance. CROC tests, with $\alpha < 0.01$ against both generalized LSA functions, showed that both functions performed significantly better than chance and that the *avg* function significantly outperformed the *sec* function. This pattern was confirmed for tests on all algorithms. We show further evaluations only with the *avg* function results.

Reflective Versus Formative Constructs

The majority of constructs that we included in this study were reflective constructs such as *ease of use*, where the items are assumed to reflect an underlying latent unobservable phenomenon (Petter et al. 2007). Whether examining reflective constructs, composites of multiple measures (Petter et al. 2007), or formative constructs, different construct types may be measured using identical items. In our research, we came across several examples of this, including the measurement of *privacy concerns* (defined as "subjects' concerns with their privacy over the Internet") (Hui et al. 2007) versus *improper access* ("consumers' opinions about improper access in organizational practices") (Malhotra et al. 2004). Both of these constructs are measured by multiple items, including the shared item, "Computer databases that contain personal information should be protected from unauthorized access—no matter how much it costs."

Figure C1 illustrates how CID operates to overcome some of these problems. In the figure, reflective constructs A and B (for example, two different *ease of use* constructs) have three items each. After evaluating all nine potential inter-item relationships for the two constructs, CID detects and links the three highest-similarity item pairs, starting with the most similar pairs. CID then uses the average similarity score as an evaluator of the construct relationship. CID conducts the same evaluations between all four constructs, resulting in the detection of high-similarity item pairs between two of the three items for formative constructs X and Y (for example, user information satisfaction [UIS], which often contains an *ease of use* item and otherwise taps into a number of different phenomena). Of the formative constructs in the example, only construct X references an ease-of-use item (X_1).

In the example, X_1 is highly similar to all items for reflective constructs A and B, but it is allowed to establish a relationship only to its highest-similarity item within each alternative construct. For example, for both reflective and formative constructs the most similar items are connected. What this means is that even when a formative construct contains items that are quite dissimilar from other items within the construct, if another formative construct, contains identical or similar items (also dissimilar within the construct), the relationship between the two formative constructs will still be high, because only the most similar items are connected between the two. If the two formative constructs focus on different phenomenon, while any linked item pair will be the *most* similar of the pairs, it will still be very dissimilar, leading to a low inter-construct relationship. Per Figure C1, the relationship between items in X and B may be X_1—B_1 = 1, X_2—B_3 = 0, X_3—B_2 = 0). If each high-

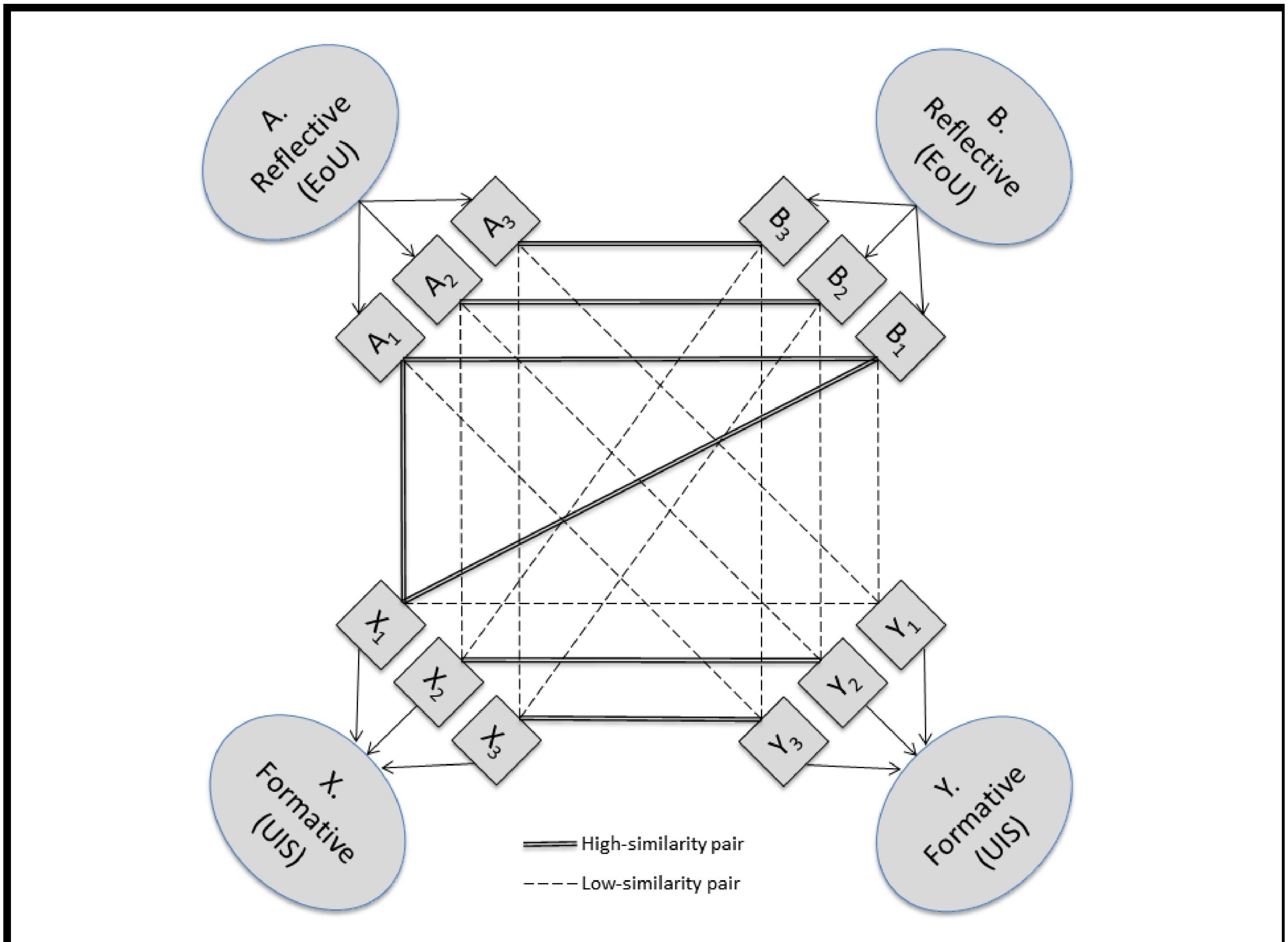

Figure C1. CID on Reflective Versus Formative Constructs

similarity pair (double lines) is scored as 1.0 and each low-similarity pair (dotted lines) is scored 0, the construct–construct scores become A—B = 1, X—Y = .66, X—A = .33, X—B = .33, Y—A = 0, and Y—B = 0. In short, the process is relatively invariant in terms of reflective and formative constructs. That is, it should work equally well for either construct type while exhibiting low-to-moderate relationships between reflective and formative constructs.

Appendix D

Evaluation of Automatic Cutoffs

To evaluate the potential impact of using the automatic F_1 cutoffs when evaluating CID_1, we conducted a separate assessment, in which a random draw resulted in 20 constructs that were used as starting points to find all constructs in the same category. We used CID_1 in the same manner as the other tests documented in this article, whereas two approaches were used to query EBSCO for each of the 20 construct categories. In the first approach, the words from the constructs' measurement items were weighted using log-entropy and the weights combined across all items in the construct; the top three words were then used to create queries. In the second approach, the name of the construct was used as a search query in up to two combinations for longer construct names. These procedures produced up to four queries for each EBSCO construct search task, of which the best performing query, as measured by maximum F_1-score, was elected to represent expert queries. The randomly selected tasks were expected to return between 1 and 56 constructs (average = 18.1 and median = 6.5, suggesting a few large tasks and several smaller tasks). CID_1 outperformed EBSCO on every assessment measure: average precision (.68 versus .46), average recall (.58 versus .39), average F_1 (.60 versus .32), average constructs found (7.50 versus 3.55), and percent of constructs found (52.82 versus 28.85), thereby supporting the original findings.

Appendix E

Supervised Machine Learning for CID

To examine the potential of supervised machine learning for CID, the six similarity designs were used to create input features for supervised machine learning using the gold standard as the target. Using the five-fold sampling process on construct pairs leads to problems because any randomly selected 80% of pairs (36 pairs) would still leave the network almost intact and result in a test set that would incorrectly represent the algorithm's predictive abilities. Therefore, a fully categorized sample of constructs was not an option. For a relatively simple test, we started with the gold standard of 1,004 constructs and randomly split the sample into five sets of 504 constructs for training (126,756 pairs) and 504 constructs for testing (again with 126,756 pairs). For each pair, the six similarity measures were used as features in the training and evaluation, and the synonymy status of each pair as the target variable.

Table E1. Evaluative Measures for Supervised Learning

J48	Precision	Recall	F_1	Correspondent Constructs Found	Total Constructs Returned
Sample 1	0.715	0.182	0.29	246	344
Sample 2	0.594	0.199	0.298	259	436
Sample 3	0.781	0.121	0.21	210	269
Sample 4	0.605	0.159	0.252	187	309
Sample 5	0.603	0.135	0.22	173	287
Average	**0.659**	**0.159**	**0.254**	**215**	**329**
Random Forest					
Sample 1	0.783	0.196	0.313	314	401
Sample 2	0.638	0.169	0.267	220	345
Sample 3	0.696	0.127	0.215	220	316
Sample 4	0.596	0.145	0.233	170	285
Sample 5	0.651	0.148	0.241	190	292
Average	**0.672**	**0.157**	**0.253**	**222.8**	**327.8**
Naïve Bayes					
Sample 1	0.146	0.416	0.216	636	4348
Sample 2	0.12	0.441	0.188	575	4799
Sample 3	0.147	0.407	0.216	705	4799
Sample 4	0.098	0.417	0.159	490	4982
Sample 5	0.111	0.45	0.178	579	5234
Average	**0.124**	**0.426**	**0.191**	**597**	**4832.4**
CID_1					
Sample 1	0.322	0.259	0.287	397	1232
Sample 2	0.411	0.258	0.317	336	818
Sample 3	0.4111	0.24	0.303	416	1011
Sample 4	0.373	0.239	0.291	281	754
Sample 5	0.409	0.255	0.314	328	803
Average	**0.385**	**0.250**	**0.302**	**351.6**	**923.6**

For each training set, a predictive model was created using three Weka version 3.7.9 algorithms with default parameters: *Naïve Bayes, J48,* and *Random Forest* (Bouckaert et al. 2013; Hall et al. 2009).[2] The test datasets were used to evaluate the success of the three algorithms against the success of the CID_1 design. As may be seen from Table E1, while CID_1 outperforms the other algorithms—especially Naïve Bayes—substantially on average F_1-scores, the precision of both J48 and Random Forest is quite impressive. Unfortunately, the recall scores for both algorithms are low. Nevertheless, the results suggest that work on supervised machine learning holds great potential for future improvements.

References

Bingham, E., and Mannila, H. 2001. "Random Projection in Dimensionality Reduction: Applications to Image and Text Data," in *KDD '01 Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* , New York: ACM, pp. 245-250.

Bouckaert, R. R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A., and Scuse, D. 2013. "WEKA Manual for Version 3-7-8," University of Waikato, New Zealand.

Burton-Jones, A., and Gallivan, M. J. 2007. "Toward a Deeper Understanding of System Usage in Organizations: A multilevel Perspective," *MIS Quarterly* (31:4), pp. 657-679.

Cohen, J. 1960. "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement* (20:1), pp. 37-46.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. 1990. "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science* (41:6), pp. 391-407.

Globerson, A., and Tishby, N. 2003. "Sufficient Dimensionality Reduction," *The Journal of Machine Learning Research* (3:7/8), pp. 1307-1331.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. 2009. "The WEKA Data Mining Software: An Update," *ACM SIGKDD Explorations Newsletter* (11:1), pp. 10-18.

Hui, K.-L., Teo, H. H., and Lee, S.-Y. T. 2007. "The Value of Privacy Assurance: An Exploratory Field Experiment," *MIS Quarterly* (31:1), pp. 19-33.

Jiang, J. J., and Conrath, D. W. 1997. "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy," in Proceedings of the International Conference on Computatioal Linguistics (ROCLING X), Taiwan.

Jones, K. S. 1986. *Synonymy and Semantic Classification*, Edinburgh, Scotland: Edinburgh University Press.

Kakkonen, T., Myller, N., Sutinen, E., and Timonen, J. 2008. "Comparison of Dimension Reduction Methods for Automated Essay Grading," *Educational Technology & Society* (11:3), pp. 275-288.

Landauer, T. K., and Dumais, S. T. 1997. "A Solution to Plato's Problem: The Latent Semantic Analysis Theory of the Acquisition, Induction, and Representation of Knowledge," *Psychological Review* (104:2), pp. 211-240.

Landauer, T. K., Foltz, P. W., and Laham, D. 1998. "An Introduction to Latent Semantic Analysis," *Discourse Processes* (25:2-3), pp. 259-284.

Landis, J. R., and Koch, G. G. 1977. "The Measurement of Observer Agreement for Categorical Data," *Biometrics* (33:1), pp. 159-174.

Leacock, C., Miller, G. A., and Chodorow, M. 1998. "Using Corpus Statistics and Wordnet Relations for Sense Identification," *Computational Linguistics* (24:1), pp. 147-165.

Lesk, M. 1986. "Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone," in *Proceedings of the 5th Annual International Conference on Systems Documentation*, New York: ACM, pp. 24-26.

Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., and Crockett, K. 2006. "Sentence Similarity Based on Semantic Nets and Corpus Statistics," *IEEE Transactions on Knowledge and Data Engineering* (18:8), pp. 1138-1150.

Lin, D. 1998. "An Information-theoretic Definition of Similarity," in *Proceedings of the 15th International Conference on Machine Learning*, Madison, WI, pp. 296-304.

Malhotra, N. K., Kim, S. S., and Agarwal, J. 2004. "Internet Users' Information Privacy Concerns (IUIPC): The Construct, the Scale, and a Causal Model," *Information Systems Research* (15:4), pp. 336-355.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. 1993. "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational Linguistics* (19:2), pp. 313-330.

Mihalcea, R., Corley, C., and Strapparava, C. 2006. "Corpus-Based and Knowledge-Based Measures of Text Semantic Similarity," in *The 21st National Conference on Artificial Intelligence*, Boston: MIT Press, pp. 775-780.

Miller, G. 1995. "WordNet: A Lexical Database for English," *Communications of the ACM* (38:11), pp. 39-41.

Miller, G. A., Beckwith, R., Fellbaum, C., Fross, D., and Miller, K. 1993. "Introduction to WordNet: An On-Line Lexical Database," *International Journal of Lexicography* (3:4), pp. 235-244.

---

[2]Weka 3.7.9 default parameters used: Naïve Bayes: All False; J48: -C 0.25 –M 2; Random Forest: -I 10 –K 0 –S 1 –num –slots.

Petter, S., Straub, D., and Rai, A. 2007. "Specifying Formative Constructs in Information Systems Research," *MIS Quarterly* (31:4), pp. 623-656.

Poli, R., Healy, M., and Kameas, A. 2010. "WordNet," in *Theory and Applications of Ontology: Computer Applications,* C. Fellbaum (ed.), New York: Springer, pp. 231-243.

Porter, M. F. 1980. "An Algorithm for Suffix Stripping," *Program* (14:3), pp. 130-137.

Resnik, P. 1995. "Using Information Content to Evaluate Semantic Similarity in a Taxonomy," in *IJCAI-95: Proceedings of the 14th International Joint Conference on Artificial Intelligence–Volume 1*, C. S. Mellish (ed.), San Francisco: Morgan Kaufmann Publishers Inc.

Siderova, A., Evangelopoulos, N., Valacich, J. S., and Ramakrishnan, T. 2008. "Uncovering the Intellectual Core of the Information Systems Discipline," *MIS Quarterly* (32:3), pp. 467-482.

Wu, Z., and Palmer, M. 1994. "Verbs Semantics and Lexical Selection," in *ACL '94 Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, Stroudsburg, PA, pp. 133-138.