Similarity Measures in Formal Concept Analysis

Faris Alqadah and **Raj Bhatnagar**
University of Cincinnati
Cincinnati, OH 45221
alqadaf@mail.uc.edu

Abstract

Formal concept analysis (FCA) has been applied successively in diverse fields such as data mining, conceptual modeling, social networks, software engineering, and the semantic web. One shortcoming of FCA, however, is the large number of concepts that typically arise in dense datasets hindering typical tasks such as rule generation and visualization. To overcome this shortcoming, it is important to develop formalisms and methods to segment, categorize and cluster formal concepts. The first step in achieving these aims is to define suitable similarity and dissimilarity measures of formal concepts. In this paper we propose three similarity measures based on existent set-based measures in addition to developing the completely novel zeros-induced measure. Moreover, we formally prove that all the measures proposed are indeed similarity measures and investigate the computational complexity of computing them. Finally, an extensive empirical evaluation on real-world data is presented in which the utility and character of each similarity measure is tested and evaluated.

1 Introduction

Formal concept analysis (FCA) has been studied and applied successively in many diverse fields such as data mining (Mohammed J. Zaki 1998) (Alqadah & Bhatnagar 2009) (Li *et al.* 2007) , conceptual modeling (Priss 2006), software engineering (Tonella 2004), social networking (Snasel, Horák, & Abraham 2008) and the semantic web (Y. Ding 2002). However, one drawback of FCA is the fact that the set of concepts tends to be quite large in dense datasets making reasoning about the concepts difficult (Pfaltz 2007). To overcome this shortcoming, it is essential to develop formalisms and methods to segment, cluster and categorize the concepts; yet as far as we know, these issues have been addressed marginally by few unrelated works (Sylvain Blachon & Gandrillon 2007).

A vital and important step in any clustering algorithm is the selection of a suitable similarity or dissimilarity measure. Nonetheless, only three previous works have attempted to define such measures for formal concepts. In (Formica 2007) the author focuses on defining a single similarity measure geared specifically towards the semantic web. Furthermore, the measure makes extensive use of a-priori knowledge in the form of a lexicographical database. The authors of (Y. Ding 2002) also define a measure based loosely on the Jaccard index, nonetheless the measure is not formally shown to be a similarity measure. Finally, the clustering algorithm presented in (Sylvain Blachon & Gandrillon 2007) makes use of a dissimilarity measure for concepts that is derived in terms of the symmetrical difference of sets, yet other measures were not studied or utilized in conjunction with the algorithm. In all these previous works the measures introduced were not formally shown to be similarity or dissimilarity measures in addition to being largely application driven. All other work in this field has focused on fuzzy concept analysis (Belohlavek 2000) (Belohlavek 2002) (R. Belohlavek 2004), while we prefer a deterministic approach.

Similarity measures for concepts in ontologies has been wideley studied (Ichise 2009), yet this is a fundamentally different problem. Concepts in ontologies are expressed as labels for data; whereas a Formal Concept of a dataset contains no label, and simply refers to a maximal biclique of objects and attributes in the dataset. Similarity measures for concepts in ontologies include string-based, graph based and knowledge based similarity measures. The string-based measures take advantage of the concept label and utilize edit distance, prefix, suffix, and n-gram similarity measures. Graph-based measures make use of the tree-structure of ontologies and integrate graph similarity along with concept similarity, which again, depends on the label of the concept (Melnik, Garcia-Molina, & Rahm 2002)(Giunchiglia, Shvaiko, & Yatskevich 2004). Finally, knowledge based similarity utilizes external knowledge sources such as a dictionary to calculate similarities.

In this paper we propose several similarity measures for unlabeled Formal Concepts, based on existent similarity measures such as the Jaccard index, Sorensen similarity index and symmetric difference that generalize the measures previously introduced for such Formal Concepts. Additionally, the novel zeros-induced index is proposed to take advantage of the fact that concepts represent maximal submatrices of 1s in the data matrix. The measures are formally shown to satisfy all the properties of similarity measures, and their computational costs are explored. Finally, an experimental study is presented in which the utility, similarity matrix characteristics and practical computational cost of all measures are compared and analyzed on real-world datasets. In the next section, we review the basic notation and definitions of FCA, while section 3 develops the novel similarity measures, and finally the empirical study is presented in section 4.

2 Formal Concept Analysis

A **context** $\mathbb{K} = (G, M, I)$ consists of two sets G, M and a relation I between G and M. The elements of G are referred to as **objects** and the elements of M as **attributes** and we assume that $G \cap M = \emptyset$. A context may be depicted as a $|G| \times |M|$ binary matrix, where the objects of G form row labels and the objects M form column labels. Let $mat(\mathbb{K})$ denote the matrix representation of $\mathbb{K}$, then we may fully specify the entries of this matrix as

$$mat(\mathbb{K})_{ij} = \begin{cases} 1 & \text{if } g_i I m_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Moreover, $\mathbb{K}$ may also be viewed as a bipartite graph, denoted as $grph(\mathbb{K})$, with vertex set $G \cup M$, and edge set I. Therefore $mat(\mathbb{K})$ is the adjacency matrix of $grph(\mathbb{K})$.

For a set $A \subseteq G$, called an **object-set**, we define

$$A' = \{m \in M | gIm \, \forall g \in A\} \tag{2}$$

the objects of M common to the objects in A. For a set $B \subseteq M$, called an **attribute-set** we also have

$$B' = \{g \in G | gIm \, \forall m \in B\} \tag{3}$$

Definition 1. *A **concept** of the context (G, M, I) is a pair $\mathbf{C} = (A, B)$ with $A \subseteq G, B \subseteq M$, such that $A' = B$ and $B' = A$. We call A the **extent** and B the **intent** of the concept (A, B). $\mathfrak{B}(G, M, I)$ denotes the set of all concepts of the context $\mathbb{K} = (G, M, I)$.*

The above definition can be shown to yield two closure systems on G and M which are dually isomorphic to each other (Gamter & Wille 1999). For every set $A \subseteq G$, A' is an intent of some concept, since (A'', A') is always a concept. Utilizing the binary matrix representation, a concept (A, B) can be represented by a maximal rectangle full of 1's under suitable permutations of the rows and columns.

Furthermore, the concepts of a context form a natural hierarchical structure.

Definition 2. *If (A_1, B_1) and (A_2, B_2) are concepts of a context, (A_1, B_1) is called a **subconcept** of (A_2, B_2), provided that $(A_1 \subseteq A_2)$ (which is equivalent to $B_2 \subseteq B_1$). In this case, (A_2, B_2) is a **superconcept** of (A_1, B_1), and we write $(A_1, B_1) \leq (A_2, B_2)$. The relation $\leq$ is called the **hierarchical order** of the concepts. A concept (A_2, B_2) is called an **upper neighbor** of (A_1, B_1) if $(A_1, B_1) \leq (A_2, B_2)$ and there is no concept (A_3, B_3) in $\mathbb{K}$ fulfilling $(A_2, B_2) \leq (A_3, B_3) \leq (A_1, B_1)$, this is denoted by $(A_2, B_2) \succ (A_1, B_1)$. The set of all concepts of (G, M, I) ordered by the hierarchical order is denoted as $\underline{\mathfrak{B}}(G, M, I)$ and is called the **concept lattice** of the context (G, M, I).*

The Basic Theorem on Concept Lattices (Gamter & Wille 1999) states that the concept lattice $\underline{\mathfrak{B}}(G, M, I)$ is a complete lattice in which the infimum and supremum are given by:

$$\bigwedge_{t \in T} (A_t, B_t) = \left(\bigcap_{t \in T} A_t, \left(\bigcup_{t \in T} B_t \right)'' \right) \tag{4}$$

$$\bigvee_{t \in T} (A_t, B_t) = \left(\left(\bigcup_{t \in T} A_t \right)'', \bigcap_{t \in T} B_t \right) \tag{5}$$

where T is an index set.

Example 1. *Consider the context depicted in figure 1(a). It contains 10 concepts, depicted as a concept lattice in figure 1(b).*

It can be easily shown that in the worst case the number of concepts in a context $\mathbb{K} = (G, M, I)$ is $2^{\min\{|G|,|M|\}}$, although this rarely occurs in real-world data, the number of concepts is still large. For example, the $8,124 \times 120$ *Mushrooms* context available from the UCI machine learning repository (Asuncion & Newman 2007) contains 238,709 concepts, far less than 2^{120}, but still an excessive number of concepts to reason with.

3 Similarity Measures

In this section we introduce several similarity measures to evaluate the similarity of concepts and segment or cluster concepts.

Definition 3. *A **similarity measure** S is a function with non-negative real values defined on the Cartesian product $X \times X$ of a set X*

$$S : X \times X \to \mathcal{R} \tag{6}$$

such that the following three properties are satisfied

1. $\exists s_0 \in \mathcal{R} : -\infty < S(x, y) \leq s_0 < +\infty, \quad \forall x, y \in X$
2. $s(x, x) = s_0 \quad \forall x \in X$
3. $s(x, y) = s(y, x) \quad \forall x, y \in X$

If in addition

1. $s(x, y) = s_0 \leftrightarrow x = y$
2. $s(x, y)s(y, z) \leq [s(x, y) + s(y, z)]s(x, z) \quad \forall x, y, z \in X$

*then S is called a **metric similarity measure***

In the pattern recognition and data mining communities, similarity measures have typically been defined on sets of real-valued or discrete-valued vectors. For discrete-valued vectors similarity measures are inspired by the comparison of sets and the cardinalate of sets. Some common set-inspired similarity measures for discrete-valued vectors include

$$\text{Jaccard index } S_{Jac} = \frac{|x \cap y|}{|x \cup y|} \tag{7}$$

$$\text{Sorenesen coefficient } S_{Sor} = \frac{2 * |x \cap y|}{|x| + |y|} \tag{8}$$

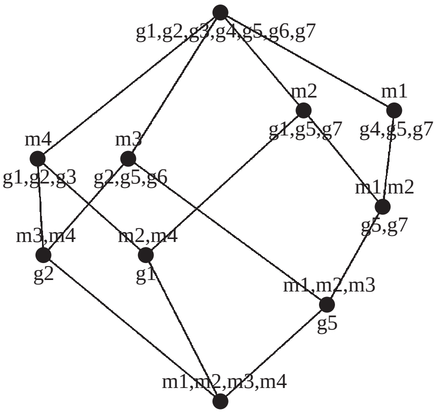
$$\text{Symmetric difference } S_{Xor} = 1 - \frac{|x \ominus y|}{|x \cup y|} \tag{9}$$

where $x \ominus y$ is the symmetric difference of x and y:

$$x \ominus y = (x \setminus y) \cup (y \setminus x) \tag{10}$$

	m_1	m_2	m_3	m_4
g_1	0	1	0	1
g_2	0	0	1	1
g_3	0	0	0	1
g_4	1	0	0	0
g_5	1	1	1	0
g_6	0	0	1	0
g_7	1	1	0	0

(a) Sample context depicted as binary matrix



(b) Sample concept lattice

Figure 1: Representations of a context, formal concepts and concept lattice

We now wish to extend these set-inspired similarity measures to concepts. A concept consists of two sets; therefore intuition suggests we weigh and combine set-based similarity measures to form a concept-based similarity measure. A similar approach was followed in (Y. Ding 2002), (Formica 2007) and (Sylvain Blachon & Gandrillon 2007) where they incorporated additional a-priori knowledge.

Definition 4. *Given concepts* $\mathbf{C_1} = (A_1, B_1)$,$\mathbf{C_2} = (A_2, B_2) \in \mathbb{K}$ *for any context* (G, M, I) *the* ***weighted concept similarity*** *of* $\mathbf{C_1}$ *and* $\mathbf{C_2}$ *is*

$$\mathcal{S}^w_S(\mathbf{C_1}, \mathbf{C_2}) = w * S(A_1, A_2) + (1-w) * S(B_1, B_2) \quad (11)$$

where $0 \leq w \leq 1$ *and* S *is the Jaccard index, Sorensen coefficient, or Symmetric Difference.*

Claim 1. *The weighted concept similarity* $\mathcal{S}^w_S$ *function is a similarity measure.*

Proof. **Case 1,** $S = S_{Jac}$:

1. By the properties of set union and set intersection $S_{Jac}(x, y) \leq 1 \quad \forall x, y$, thus by the definition of weighted concept similarity, $s_0 = 1$.
2. Property 2 is trivially satisfied by the fact that S_{Jac} is a similarity measure, thus $S_{Jac}(x, x) = 1$ and therefore

   $$\mathcal{S}^w_{Jac}(\mathbf{C_1}, \mathbf{C_1}) = w{*}1 + (1-w){*}1 = 1 \quad \forall \mathbf{C_1} \in \mathfrak{B}(G, M, I)$$

3. Property 3 is also satisfied by the fact that S_{Jac} is a similarity measure, so $S_{Jac}(x, y) = S_{Jac}(y, x)$ thus

   $$\begin{aligned} &\mathcal{S}^w_{Jac}(\mathbf{C_1}, \mathbf{C_2}) \\ &= w * S_{Jac}(A_1, A_2) + (1-w) * S_{Jac}(B_1, B_2) \\ &= w * S_{Jac}(A_2, A_1) + (1-w) * S_{Jac}(B_2, B_1) \\ &= \mathcal{S}^w_{Jac}(\mathbf{C_2}, \mathbf{C_1}) \end{aligned}$$

Case 2, $S = S_{Sor}$:

1. For any two sets x, y $|x| + |y| \geq 2 * (|x \cap y|)$, however if $x = y$ then $|x| + |y| = 2 * (|x \cap y|)$, thus $s_0 = 1$.
2. Analogous to (2) in case 1, due to the fact that $s_0 = 1$ and S_{Sor} is a similarity measure.
3. Analogous to (3) in case 1.

Case 3: We first show that S_{Xor} is a similarity measure, with $S_0 = 1$, with the rest of the proof being analogous to case 1.

1. For any two sets x, y we have

   $$\begin{aligned} x \setminus y &\subseteq x \\ y \setminus x &\subseteq y \\ (x \setminus y) \cup (y \setminus x) &\subseteq (x \cup y) \\ 1 - \frac{|x \ominus y|}{|x \cup y|} &\leq 1 \end{aligned}$$

2. By definition of symmetric set difference $x \ominus x = \emptyset$, thus $S_{Xor}(x, x) = 1$
3. Follows directly from the commutative property of symmetric set difference.

□

The set-based similarity measures are based on well-established similarity measures and are efficient to compute. Set intersection, union, and difference of any two sets x, y can be computed in $O(\min\{|x|, |y|\})$ time, thus the worst case time of all the set-based measures is $O(\min(\{|A_1|, |B_1|, |A_2|, |B_2|\})$ for any given pair of concepts (A_1, B_1) and (A_2, B_2). Although, all the weighted concept similarity measures enjoy the same theoretical computation cost, the practical cost of computing the measures differ significantly in real-world data, as will be illustrated by our empirical study.

The set-based measures encompass two shortcomings: First, setting the value of w greatly effects the measure, and thus computing similarity cannot be performed parameter-free. Second, the measures only consider the cardinalate of the sets and do not explicitly consider the amount of information shared between two concepts. For example, consider the concepts $\mathbf{C_1} = (\{g_5\}, \{m_1, m_2, m_3\})$, $\mathbf{C_2} = (\{g_2, g_5, g_6\}, \{m_3\})$, and $\mathbf{C_3} = (\{g_4, g_5, g_7\}, \{m_1\})$ of the context depicted in figure 1(a). Let $w = 0.5$, then we have

$$\begin{aligned} \mathcal{S}^{0.5}_{Jac}(\mathbf{C_1}, \mathbf{C_2}) &= \mathcal{S}^{0.5}_{Jac}(\mathbf{C_1}, \mathbf{C_3}) = 0.333 \\ &\text{and} \\ \mathcal{S}^{0.5}_{Sor}(\mathbf{C_1}, \mathbf{C_2}) &= \mathcal{S}^{0.5}_{Sor}(\mathbf{C_1}, \mathbf{C_3}) = 0.5 \end{aligned}$$

We see that the weighted concept similarity yields equivalent similarity between $\mathbf{C_1}, \mathbf{C_2}$ and $\mathbf{C_1}, \mathbf{C_3}$ in both cases. This result is reasonable in terms of the overlap between the sets of attributes and objects of each concept, yet closer inspection of the context would suggest otherwise. Comparing concept $\mathbf{C_2}$ to $\mathbf{C_1}$ we see that $\mathbf{C_2}$ dropped attributes m_1, m_2 but gained objects g_2, g_6; upon inspecting the context, we observe that these objects do not encompass attributes m_1 and m_2. On the other hand, comparing concept $\mathbf{C_3}$ to $\mathbf{C_1}$ we see that $\mathbf{C_3}$ dropped attributes m_2, m_3 and gained objects g_4, g_7; upon inspecting the context, we observe that object g_7 does encompass attribute m_7 (this can also be observed in the lower neighbor of $\mathbf{C_3}$). The fact that the objects of $\mathbf{C_3}$ encompass more attributes of $\mathbf{C_1}$ than the objects of $\mathbf{C_2}$ infers that the similarity between $\mathbf{C_1}$ and $\mathbf{C_3}$ should be greater than that of $\mathbf{C_1}$ and $\mathbf{C_2}$; however this is not reflected utilizing weighted concept similarity.

In order to consider all information shared between two concepts we look beyond set based similarity measures. Concepts may be viewed as maximal sub-matrices full of 1s in $mat(\mathbb{K})$, and thus combining any two concepts $\mathbf{C_1} = (A_1, B_1)$ and $\mathbf{C_2} = (A_2, B_2)$ to form a larger sub-matrix $\mathbf{D} = (A_1 \cup A_2, B_1 \cup B_2)$ must result in the introduction of zeros. We may then think of the similarity between $\mathbf{C_1}$ and $\mathbf{C_2}$ in terms of the number of zeros introduced.

Definition 5. *Given any two concepts $\mathbf{C_1} = (A_1, B_1)$, $\mathbf{C_2} = (A_2, B_2)$ of a context $\mathbb{K}$ then the **zeros induced** by $\mathbf{C_1}$ and $\mathbf{C_2}$, denoted as $z(\mathbf{C_1}, \mathbf{C_2})$, is the number of zeros enclosed by the sub-matrix induced by rows $(A_1 \cup A_2)$ and columns $(B_1 \cup B_2)$ in $mat(\mathbb{K})$.*

Computing $z(\mathbf{C_1}, \mathbf{C_2})$ consists of summing up the number of zeros in each row of the sub-matrix induced by $\mathbf{C_1}$ and $\mathbf{C_2}$:

$$z(\mathbf{C_1}, \mathbf{C_2}) = \sum_{a \in A_1 \cup A_2} |(B_1 \cup B_2) \setminus a'| \qquad (12)$$

Definition 6. *Given concepts $\mathbf{C_1} = (A_1, B_1)$ and $\mathbf{C_2} = (A_2, B_2)$ the **zeros-induced index** is*

$$S_z = \frac{|A_1 \cup A_2| * |B_1 \cup B_2| - z(\mathbf{C_1}, \mathbf{C_2})}{|A_1 \cup A_2| * |B_1 \cup B_2|} \qquad (13)$$

Claim 2. *The zeros-induced index is a concept similarity measure.*

Proof.

1. For any two sets x, y $x \setminus y \subseteq x$, thus $z(\mathbf{C_1}, \mathbf{C_2}) \leq |A_1 \cup A_2| * |B_1 \cup B_2| \quad \forall \mathbf{C_1}, \mathbf{C_2}$, implying that $s_0 = 1$.
2. For any concept $\mathbf{C} = (A, B)$, by definition $A' = B$ which implies
$$\begin{aligned} \forall a \in A \quad a' &\supseteq B \\ \rightarrow z(\mathbf{C}, \mathbf{C}) &= 0 \\ \rightarrow S_z(C, C) &= s_0 \end{aligned}$$
3. Property 3 is guaranteed by the commutative property of set union.

□

The zeros-induced index does not require any parameters and also considers all information relating the two sets of attributes and objects. Consider once again concepts $\mathbf{C_1} = (\{g_5\}, \{m_1, m_2, m_3\})$, $\mathbf{C_2} = (\{g_2, g_5, g_6\}, \{m_3\})$, and $\mathbf{C_3} = (\{g_4, g_5, g_7\}, \{m_1\})$. Applying the zeros-induced index we have

$$\begin{aligned} \mathcal{S}_z(\mathbf{C_1}, \mathbf{C_2}) &= \frac{9-4}{9} = \frac{5}{9} \\ &\text{and} \\ \mathcal{S}_z(\mathbf{C_1}, \mathbf{C_3}) &= \frac{9-3}{9} = \frac{2}{3} \end{aligned}$$

The example illustrates the more discerning result of assigning greater similarity to $\mathbf{C_1}$ and $\mathbf{C_3}$ is accomplished utilizing the zeros-induced index. Computing S_z, however, is much more expensive than any set-based measure. A direct implementation of equation 12 entails $O(|A_1 \cup A_2|)$ set differences resulting in $O(\max\{|A_1|, |B_1|, |A_2|, |B_2|\}^2)$ time complexity for each pair of concepts.

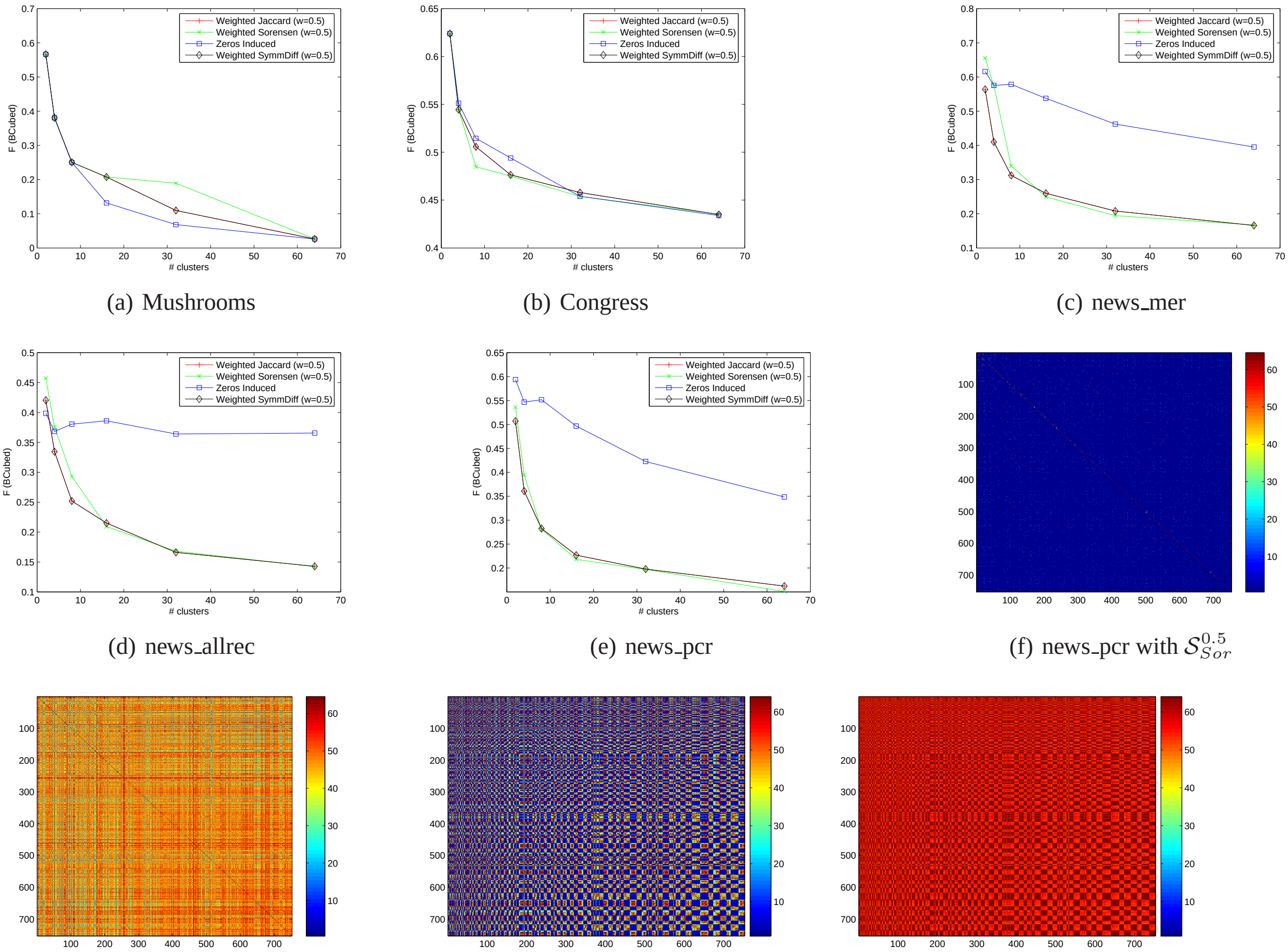
4 Experiments

Name	Dimensions	Density	Num. classes
Congress	435×48	0.33	2
Mushrooms	8124×120	0.1917	2
news_mer	2000×892	0.003	2
news_pcr	1997×1025	0.0026	2
news_allrec	3124×1671	0.0014	4

Figure 2: Datasets used in empirical evaluation

Several experiments were performed to empirically compare the utility of each similarity measure. Real-world, labeled datasets were obtained from the UCI machine learning repository (Asuncion & Newman 2007), and are summarized in figure 2. The concepts of each context were enumerated via an implementation of the concept enumeration algorithm described in (Berry, Bordat, & Sigayret 2007), and the similarity matrix of the concepts was computed for each similarity measure. Finally, the CLUTO agglomerative clustering algorithm (Clu 2009) was applied to the similarity matrices while varying the number of desired clusters.

To compare the utility of the similarity measures, the cluster validity of each clustering was determined via the $F(BCubed)$ extrinsic cluster validity metric; this metric



(a) Mushrooms

(b) Congress

(c) news_mer

(d) news_allrec

(e) news_pcr

(f) news_pcr with $\mathcal{S}_{Sor}^{0.5}$

(g) news_pcr with S_z

(h) Congress with $\mathcal{S}_{Sor}^{0.5}$

(i) Congress with S_z

Figure 3: Clustering results and similarity matrices

combines the B^3Prec and B^3Rcl of a cluster using the F_1 score. It was illustrated in (Enrique Amig & Verdejo 2008) that the $F(BCubed)$ metric maintains the desirable properties of cluster homogeneity, cluster completeness, rag bag, and cluster size vs quantity while other popular cluster validity metrics such as precision, inverse-precision, entropy and mutual information do not. Moreover, $F(BCubed)$ accounts for both hard and soft clusterings of objects. Formally, for any object e of G, there exists a set of ideal categories (class labels), denoted by $L(e)$ to which e belongs. Also let $C(e)$ denote the set of clusters that e belongs to. Given this we may define the multiplicity precision and multiplicity recall between any two objects e and e' as follows:

$$MultPrec(e, e') = \frac{\min(|C(e) \cap C(e')|, |L(e) \cap L(e')|)}{|C(e) \cap C(e')|}$$

$$MultRcl(e, e') = \frac{\min(|C(e) \cap C(e')|, |L(e) \cap L(e')|)}{|L(e) \cap L(e')|}$$

Note that $MultPrec$ is only defined when e and e' share a cluster and $MultRcl$ is only defined when e and e' share a category. Intuitively, $MultPrec$ grows if there is a matching category for each cluster where two objects co-occur; $MultRcl$ grows when we add a shared cluster for each category shared by two items. Thus if we have fewer shared clusters than needed, we lose recall; if we have fewer categories than clusters we lose precision. From these measures the BCubed measures are derived as:

$$B^3Prec = Avg_e \left[Avg_{e', C(e) \cap C(e') \neq \emptyset} \left[MultPrec(e, e') \right] \right]$$

$$B^3Rcl = Avg_e \left[Avg_{e', L(e) \cap L(e') \neq \emptyset} \left[MultRcl(e, e') \right] \right]$$

Figure 3 illustrates the results of the clustering experiment and reveal an interesting trend. All similarity measures lead to comparable clustering results in the two dense datasets of Mushrooms and Congress, with a slight edge to $\mathcal{S}_{Sor}$ and S_z respectively. However, on all the newsgroup datasets, which were quite sparse, the S_z measure consistently produced superior clustering results. Specifically, we found that for any number of clusters, the recall was significantly larger when the zeros induced index was utilized on these sparse datasets. This trend points out the advantage of the fine-grain approach of the zeros induced index as opposed to the set-based measures particularly for sparse

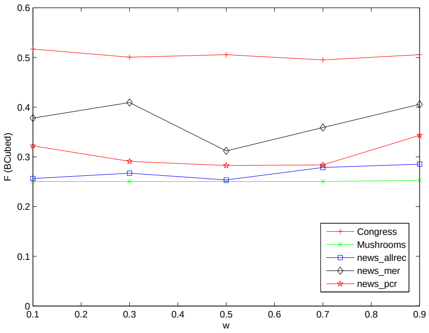


Figure 4: Effect of w on set-based measures

Dataset	Similarity Measure	CPU Time (seconds)
Mushrooms	Weighted Jaccard	545.23 ± 3.45
	Weighted Sornensen	300.35 ± 1.64
	Weighted SymmDiff	961.62 ± 2.13
	Zeros Induced	4125.22 ± 3.76
Congress	Weighted Jaccard	522.24 ± 4.2204
	Weighted Sornensen	289.89 ± 0.69
	Weighted SymmDiff	885.89 ± 2.77
	Zeros Induced	3233.54 ± 3.45
news_allrec	Weighted Jaccard	3.9170 ± 0.0440
	Weighted Sornensen	2.6630 ± 0.0517
	Weighted SymmDiff	6.1900 ± 0.0474
	Zeros Induced	8.2050 ± 0.1203
news_mer	Weighted Jaccard	0.7700 ± 0.0067
	Weighted Sornensen	0.5100 ± 0.0176
	Weighted SymmDiff	1.2270 ± 0.0134
	Zeros Induced	1.9720 ± 0.0225
news_pcr	Weighted Jaccard	0.7680 ± 0.0092
	Weighted Sornensen	0.5040 ± 0.0158
	Weighted SymmDiff	1.2280 ± 0.0235
	Zeros Induced	1.8530 ± 0.0183

Figure 5: CPU times for computing similarity measures

data. Exploring the characteristics of the similarity matrices (figures 3(f) -3(i)) further explains this trend. The set-based measures produce very sparse similarity matrices in sparse data (figure 3(f)) due to the fact that concepts that do not explicitly share an object or attribute are penalized, and thus assigned low scores. On the other hand, the zeros-induced index produces dense similarity matrices in both sparse and dense datasets (figures 3(g) and 3(i)); this is due to the fine-grain approach of accounting for all the 1s in the induced sub-matrix of each pair of concepts. When the similarity matrices of all measures were of comparable densities (in dense datasets) the clustering quality was also consistently comparable. However, the additional information retained by the zeros-induced similarity matrices in sparse data constantly lead to both higher recall and overall higher quality of clusters.

The effect of w on the performance of the set-based measures was also investigated. Figure 4 displays the result of this experiment. Clearly, the effect of w is highly dataset dependent; for example in the news_mer dataset the quality of the clustering fluctuates with every different value of w, while the opposite is true in the Mushrooms dataset. Setting the value of w thus remains a challenge and disadvantage when utilizing the set-based measures. Finally, several performance tests were conducted to investigate the practical computation cost of each measure. Each similarity matrix was computed ten times on a 2.7 GHz 2x AMD Athlon CPU with 6 GB of main memory, and the average CPU time is reported in figure 5. Although, all the set-based measures have the same theoretical time complexity, we clearly see that the weighted Sorensen measure is the least costly. This is attributed to the fact that only a single set intersection needs to be computed per concept pair, without computing the set union. The theoretical computational cost of the zeros-induced measure was quadratic compared to the linear cost of set-based measures, and this is demonstrated in the performance tests. In the dense datasets of Congress and Mushrooms the CPU time is at least an order of magnitude larger than the set-based approaches.

5 Conclusion

In this paper we have taken a necessary first step towards clustering formal concepts by specifying and studying four similarity measures. Three of these measures were inspired by existent set-based similarity and dissimilarity measures, while the completely novel zeros-induced index was introduced. This novel measure takes advantage of the fact that concepts form maximal sub-matrices of 1s in the data matrix by computing the ratio of 1s to the total area of the sub-matrix induced by joining two concepts. All four measures were formally proven to be similarity measures and the computational cost of each measure was analyzed. Initial empirical studies indicate that the zeros-induced index leads to superior clustering results in sparse data, and comparable results in dense datasets. However, the computational cost of the measure in time is substantially more expensive than all other measures. Future work may focus on formally specifying more similarity measures by incorporating the structure of the concepts as established by the hierarchical order and the concept lattice.

References

Alqadah, F., and Bhatnagar, R. 2009. Discovering substantial distinctions among incremental bi-clusters. In *Proceedings of the 2009 SIAM International Conference on Data Mining*.

Asuncion, A., and Newman, D. 2007. UCI machine learning repository.

Belohlavek, R. 2000. Similarity relations in concept lattices. *Journal of Logic and Computation* 10 (6):823–845.

Belohlavek, R. 2002. Combination of knowledge in fuzzy concept lattices. *International Journal of Knowledge-Based Intelligent Engineering Systems* 6 (1):9–14.

Berry, A.; Bordat, J.-P.; and Sigayret, A. 2007. A local approach to concept generation. *Annals of Mathematics and Artificial Intelligence* 49:117–136.

2009. Cluto: Family of data clustering software tools.

Enrique Amig, Julio Gonzalo, J. A., and Verdejo, F. 2008. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval* Online.

Formica, A. 2007. Concept similarity in formal concept analysis: An information content approach. *Knowledge-Based Systems* 21:80–87.

Gamter, B., and Wille, R. 1999. *Formal Concept Analysis: Mathematical Foundations*. Berlin: Springer-Verlag.

Giunchiglia, F.; Shvaiko, P.; and Yatskevich, M. 2004. S-match: an algorithm and an implementation of semantic matching. 61–75.

Ichise, R. 2009. Evaluation of similarity measures for ontology mapping. 15–25.

Li, J.; Liu, G.; Li, H.; and Wong, L. 2007. Maximal biclique subgraphs and closed pattern pairs of the adjacency matrix: A one-to-one correspondence and mining algorithms. *IEEE Trans. Knowl. Data Eng.* 19(12):1625–1637.

Melnik, S.; Garcia-Molina, H.; and Rahm, E. 2002. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. *Data Engineering, International Conference on* 0:0117.

Mohammed J. Zaki, M. O. 1998. Theoretical foundations of association rules. *3rd SIGMOD'98 Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD)*.

Pfaltz, J. L. 2007. Representing numeric values in concept lattices. *Fifth International Conference on Concept Lattices and Their Applications*.

Priss, U. 2006. Formal concept analysis in information science. *Annual Review of Information Science and Technology* 40.

R. Belohlavek, J. Dvorak, J. O. 2004. Fast factorization of concept lattices by similarity. In *Proceedings of Concept Lattices and their Applications*.

Snasel, V.; Horák, Z.; and Abraham, A. 2008. Understanding social networks using formal concept analysis. In *WI-IAT '08: Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*.

Sylvain Blachon, Ruggero G. Pensa, J. B. C. R. J.-F. B., and Gandrillon, O. 2007. Clustering formal concepts to discover biologically relevant knowledge from gene expression data. *In Silico Biology* 7:467–483.

Tonella, P. 2004. Formal concept analysis in software engineering. In *Proceedings of the International Conference on Software Engineering*.

Y. Ding, D. Fensel, M. K. B. O. 2002. The semantic web: yet another hip? *Data & Knowledge Engineering* 41:205–227.