A split-merge framework for comparing clusterings

Xiang, Qiaoliang

2013

Xiang, Q. (2013). A split-merge framework for comparing clusterings. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/55194>

<https://doi.org/10.32657/10356/55194>

A SPLIT-MERGE FRAMEWORK FOR COMPARING CLUSTERINGS

XIANG QIAOLIANG

SCHOOL OF COMPUTER ENGINEERING

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Master of Engineering

2013

Acknowledgments

Thank my academic supervisor Dr. Ivor Wai-Hung TSANG for his support and guidance through my academic research. Thank all the reviewers for their precious time and great feedbacks for improving the thesis. I also want to thank all my friends and members in the Center for Computational Intelligence Lab (C2i) for their great helps and inspiring discussions. Special thanks are given to my parents and my girlfriend for their unconditional understanding, support, and love.

Contents

Acknowledgments	i
Abstract	v
1 Introduction	1
2 Preliminaries	7
3 Conditional Normalization	9
4 A Survey of Existing Measures	12
4.1 Pair Counting Measures	14
4.2 Cluster Counting Measures	17
4.2.1 Extract Matching	17
4.2.2 Overlap Matching	18
4.3 Set Matching Measures	20
4.3.1 Local Matching	20
4.3.2 Global Matching	23
4.4 Information Theoretic Measures	24
5 A Generative Model for Measures	28
5.1 Aspect	28
5.2 Operation	30

5.2.1	Precondition and Postcondition	31
5.2.2	Combination Operation	33
5.2.3	Similarization Operation	34
5.2.4	Normalization Operation	35
5.2.5	Symmetrization Operation	36
5.3	Application	37
6	A Graph-Based Model of Clusterings	41
6.1	Component	41
6.2	Decomposition on Components	44
7	The Split-Merge Framework	46
7.1	Subcomponent	46
7.1.1	Subcomponent Similarity Measure	48
7.2	Subcomponent Pair	49
7.2.1	Subcomponent Pair Similarity Measure	52
7.3	Properties of the Framework	53
7.3.1	Time Complexity	53
7.3.2	Symmetry	54
7.3.3	The Best Clustering and Worst Clusterings	54
7.3.4	Normalization	56
7.3.5	Top and Bottom	57
7.3.6	Split and Merge	57
7.3.7	Cluster Homogeneity and Cluster Completeness	59
7.3.8	Decomposition and Consistency	59
7.3.9	Bounds	60
7.4	Comparisons with Other Frameworks	61

7.4.1	The Arithmetic Mean Framework	61
7.4.2	The Direct-sum Framework	62
7.5	Examples of the Framework	66
7.5.1	Reusing Existing External Measures	66
7.5.2	Reusing Existing Internal Measures	68
7.6	Comparisons with Existing Measures	69
7.6.1	Theoretical Comparisons	69
7.6.2	Empirical Comparisons	70
8	Conclusions	73
	Appendix A Conditional Normalization	74
	Appendix B Component-Based Decomposition	79
	References	92

Abstract

External clustering evaluation measures are often used to evaluate the performance of different clustering algorithms on a collection of data sets. Traditional normalization property is no longer suitable for this task and a conditional normalization property is proposed based on the fact that one clustering is the ground-truth. Even existing measures have been proposed from different points of view, we study them from the normalization point of view. Besides, we propose a new category of cluster counting measures and further group set matching measures into two subcategories according to how the matching is performed. Furthermore, we propose a generative model to study how existing measures are generated as well as producing new measures according to application requirements. In order to understand the intrinsic properties of a measure, a graph-based model is presented to model two clusterings as a directed bipartite graph, which can be decomposed into weakly connected components. A measure can be expressed as a conic combination of scores on components, and different weights are assigned to components when aggregating their scores. Based on the graph-based model, we propose a split-merge framework by breaking components into subcomponents and combining the scores of any two related subcomponents. It is conditionally normalized while existing measures are not. It also has many nice properties compared to other existing frameworks. We give some examples of the framework and compare one example with a few representative measures theoretically and empirically on a coreference resolution data set.

Chapter 1

Introduction

Hierarchical clustering algorithms produce a nested series of partitions, while partitional (or flat) clusterings algorithms produce a single partition. An object can belong to different clusters in soft clustering, while hard clustering restricts each object to be grouped into only one cluster. In this thesis, we restrict to study hard partitional clusterings, which group objects into a set of disjoint clusters. The task of cluster validation is to evaluate the quality of clusterings using proper evaluation measures. It plays an important role in comparing different clustering algorithms, selecting the optimal number of clusters, and determining the optimal parameters of algorithms. Three types of evaluation measures have been developed so far. An external measure compares a clustering with respect to its true (or ground-truth) clustering, an internal measure quantifies the compactness or separation of a clustering based on the feature representations (i.e., vectors) of objects, and a hybrid measure takes both the true clustering and the feature representations of objects into account. External measures are preferred because they better reflect human evaluation [44], and they are independent of the algorithm or the way the clusterings are obtained [27]. Therefore, we mainly focus on studying external evaluation measures. From now on, a measure is considered as an external evaluation measure unless otherwise stated.

Clustering evaluation measures are usually used to compare the performance of various algorithms on a collection of selected benchmark data sets. The final score of an algorithm is based on aggregating (i.e., averaging) the scores on all data sets. Each data set has its own characteristics and true clustering, which reflect the degree of difficulty for clustering. In order to fairly combine the scores on different data sets, the scores on different data sets should be rescaled to the same range, such as the closed interval $[0, 1]$. This requires a measure to be irrelevant to the degree of clustering difficulty of every data set to avoid producing biased scores [24]. Many measures try to satisfy the normalization property so as to solve the issue [48, 47]. The normalization property focuses on the joint space of two clusterings and requires the scores to be scaled to the closed interval $[0, 1]$. In the context of external evaluation, the true clustering is given and fixed on a data set, and the scores of system-generated clusterings should be normalized with respect to the true clustering. Inspired from this, we propose an enhanced conditional normalization property to better suit the task of external evaluation in Chapter 3. An example of not conditionally normalized measure is the pairwise precision which give zero score for any predicted clustering when the true clustering is full of singletons since there are no shared pairs. To make it conditionally normalized, the number of shared singletons can be used when the true clustering is full of singletons.

Existing external measures can be roughly divided into three categories. Pair counting measures are based on counting the pairs of objects for which two clusterings agree or disagree. They are sensitive to parameters, such as the size of a cluster, the number of clusters, and the number of objects [48]. Set matching measures find a maximum matching between two clusterings. They make no assumption on how clusterings are generated, but they ignore those unmatched clusters [28]. Information theoretic measures do not suffer from the problems of pair counting and set matching measures. They have been analyzed extensively and systematically in recent years [28, 47]. Some measures

tend to give high scores in practice, so adjusted measures, such as adjusted Rand index [18] and adjusted mutual information [47], are proposed to address this issue, but they are not normalized because they may be negative [28]. To better study and compare measures from different categories, we extend the traditional categorization scheme by proposing a new category of cluster counting measures and introducing two subcategories for set matching measures based on how the matching is performed. As mentioned before, unnormalized measures are inappropriate for comparing clusterings across data sets [47, 48]. Even all existing measures have been proposed from different aspects under different assumptions, we give a comprehensive survey of them from the normalization point of view in Chapter 4.

In some applications, existing measures cannot take advantage of the domain knowledge and new measures that can be adapted to different application scenarios are wanted. To solve this issue, we propose a generative model for measures in Chapter 5. It helps us to understand how existing measures are generated. It can systematically generate many possibly new measures with different properties, and the measure whose properties are most desired from an application point of view can be selected. Broadly speaking, a measure is a measurement on two objects (i.e., cluster, clustering). The generative model contains two parts: aspect and operation. An aspect is a group of related properties, some of which are more desirable than others according to a given application. An operation is defined for each aspect to change undesirable properties to desirable properties. We defined four aspects. For instance, a similarization aspect indicates whether a measure is a similarity measure or a distance measure, a normalization aspect contains information about whether a measure is normalized or not, and a symmetrization aspect tells whether a measure is symmetric or asymmetric.

Measures should be independent of some inherent structures (e.g., such as the number of objects) of two clusterings [28]. After surveying many existing measures, we find most

measures treat the clustering task from different points of view, such as binary classification, multi-class classification, information retrieval, and information loss and gain, because they were originally designed for different domains. The generative model mentioned before can produce lots of measures with different external properties according to application requirements. We need a tool to study the intrinsic properties of measures, so we propose a graph-based mode of clusterings in Chapter 6 that is independent of any application. We model the relation between two clusterings as a directed bipartite graph. One major reason of using a directed graph to model two clusterings is that a directed graph can be decomposed into several weakly connected components, which help us to study the intrinsic property of a measure in a divide-and-conquer manner.

Based on the aforementioned graph-based model, in Chapter 7, we propose a split-merge framework of similarity measures that can be tailored to different applications [16] instead of focusing on designing a new evaluation measure. We decompose every component into subcomponents and combine two related subcomponents into a subcomponent pair. Once the subcomponent similarity measures are given, the split-merge framework defines a scheme to obtain a subcomponent pair measure. The final similarity is the summation of the similarities of all subcomponent pairs. The split-merge framework is easy to use, flexible, and extensible. It also has several desirable properties. Experimental results on a real world data set demonstrate that one instantiated measure is monotonically decreasing and conditionally normalized, while others are not.

The contributions of this work are summarized below. The work that differ from that of the confirmation report are emphasized.

- The traditional normalization property focuses on the joint space of two clusterings, which is not suitable for the task of external evaluation. We propose a conditional normalization property in Chapter 3 to consider the space of clusterings with respect to any ground-truth clustering. *In addition, we propose a two-step method*

to determine whether a measure is conditionally normalized or not, and we give the conditional normalization properties of existing measures in Appendix A.

- In chapter 4, we extend the three traditional categories of measures by introducing a new category of cluster counting measures and proposing two subcategories for set matching measures. *We review existing measures from the normalization point of view, and we also present the pattern or strategy used by each category or subcategory to generate measures, which allows us to better understand the similarities and differences between existing measures.*
- In chapter 5, we propose a generative model to produce different application-specific measures. *Measures can be generated in a systematic way, which is helpful to design and select the best measure under different application scenarios according to how the scores are aligned with the ground-truth scores.*
- In chapter 6, we propose a graph-based model of clusterings to study the intrinsic properties of measures by decomposing a measure into a canonical combination of scores on the disjoint components of the graph. *We find a method to decompose a normalized similarity measure based on the decomposition of its basic unnormalized measure, and the decompositions of existing measures are derived in Appendix B.*
- In chapter 7, we propose a split-merge framework. The framework enjoys many nice properties of the variation of information [28]. Most importantly, it is conditionally normalized while existing measures discussed in the thesis are not conditionally normalized. *We also compare the framework with two existing frameworks.* The framework is flexible and easy to use. We propose some example of the framework by reusing existing external and internal measures.

The rest of the thesis is organized as follows. Chapter 2 introduces a few basic concepts and notations that are relevant to set, clustering and measure. We propose the conditional normalization property and discuss how to check whether a measure is

conditionally normalized or not in Chapter 3. In Chapter 4, we present a comprehensive survey of existing measures from the normalization point of view. We also propose a new categorization scheme for the purpose of better studying and comparing existing measures. And we find all existing measures discussed in the thesis are not conditionally normalized. In Chapter 5, we propose a generative model to generate measures according to different applications. We propose a graph-based model of clusterings to study the intrinsic properties of measures in a divide-and-conquer manner in Chapter 6. Based the study, we propose the split-merge framework of similarity measures in Chapter 7. We define the framework in Section 7.2, study its properties in Section 7.3, and compare it with other existing frameworks in Section 7.4. Some examples of the framework are presented in Section 7.5, and a representative example is compared with a few selected representative measures theoretically and empirically in Section 7.6. We conclude the thesis in Chapter 8.

Chapter 2

Preliminaries

We first introduce some set notations. A set is a collection of distinct elements. If x is a member of a set X , this is denoted $x \in X$, while $x \notin X$ means x is not a member of X . The size $|X|$ of a set X is the number of elements within the set. An empty set has no elements and the size is zero. Given a collection of sets, their union (denoted by $\cup$) is the set of all distinct elements in the collection, and their intersection (denoted by $\cap$) is the set of elements that appear in every set. The set difference $X \setminus Y$ is the set of elements in X but not in Y . A set X is a subset of a set Y (written $X \subseteq Y$) if all elements of X are also elements of Y . If $X \subseteq Y$ and X is not equal to Y , X is a proper subset of Y , denoted $X \subset Y$. The Cartesian product $X \times Y$ is the set of all ordered pairs (x, y) where $x \in X$ and $y \in Y$.

A data set $D = \{1, 2, \dots, n\}$ is a set of n objects. A cluster is a subset of D . An empty cluster $\emptyset$ has no objects, and a singleton contains only one object. A clustering is a set of nonempty pairwise disjoint clusters whose union is D . $\emptyset$ denotes an empty clustering which has no clusters and objects. The top $\top = \{D\}$ groups all objects together, while the bottom $\perp = \{\{i\} \mid i \in D\}$ treats each object as a singleton [15]. External clustering evaluation is usually performed on two clusterings: a true clustering $\mathbb{L}$ and a predicted clustering $\mathbb{C}$. The true clustering provides information about how objects are correctly organized, while the predicted clustering is usually produced by algorithms, programs

or human beings. A cluster $L \in \mathbb{L}$ is a true cluster and a cluster $C \in \mathbb{C}$ is a predicted cluster.

Let Ω be the set of all clusterings based on D . We introduce a few concepts from lattice theory [15]. If every cluster of a clustering is a subset of some cluster of another clustering, the first clustering is said to refine the second clustering. The refinement relation is a partial ordering on Ω . The meet between two clusterings is the largest clustering which refines them, while their join is the smallest clustering that is refined by them. The meet contains all nonempty intersections between every cluster from one clustering and every cluster from another clustering, and every cluster in the join is the smallest set that is exactly a union of a few clusters from two clusterings. The meet is almost equivalent to the contingency table or confusion matrix, based on which most evaluation measures are defined [47]. When an appropriate data structure is used, the meet and join can be computed in $O(n)$ time complexity [35]. Denote the meet and the join between $\mathbb{L}$ and $\mathbb{C}$ by $\mathbb{M}$ and $\mathbb{J}$, respectively. $\mathbb{M}$ can be analytically expressed as

$$\mathbb{M} = \{L \cap C \mid L \in \mathbb{L}, C \in \mathbb{C}, L \cap C \neq \emptyset\}.$$

A clustering evaluation measure is a function $f : \Omega \times \Omega \rightarrow \mathbb{R}$ that gives a real-valued score to a predicted clustering based on a true clustering, where the Cartesian product $\Omega \times \Omega$ is the set of all ordered pairs $\{(\mathbb{L}, \mathbb{C}) \mid \mathbb{L} \in \Omega, \mathbb{C} \in \Omega\}$, and $\mathbb{R}$ is the set of real numbers. There are $|\Omega| \times |\Omega|$ ordered pairs in total. A measure is a similarity (resp. distance) measure if a higher (resp. lower) score indicates the two clusterings are more similar. A measure is symmetric if the score of any ordered pair $(\mathbb{L}, \mathbb{C})$ is the same with that of its reversed ordered pair $(\mathbb{C}, \mathbb{L})$ and asymmetric otherwise. A measure is normalized if the scores are within the closed interval $[0, 1]$ and unnormalized otherwise. The similarity and distance measures are convertible, so we mainly focus on studying similarity measures in this thesis.

Chapter 3

Conditional Normalization

Each data set has its own true clustering, which reflects the degree of difficulty for clustering. When comparing clusterings of different data sets or comparing the performances of different algorithms by aggregating (i.e., averaging) the performances on different data sets, a measure should be irrelevant to the degree of clustering difficulty on a data set to avoid generating biased scores, which can be achieved by using normalization [24]. The traditional normalization focuses on the joint space $\Omega \times \Omega$ of two clusterings and requires the scores to be scaled to the closed interval $[0, 1]$ [48, 47]. However, the primary purpose of external clustering evaluation is to rank all predicted clusterings against a given true clustering. [24] suggest to focus on the space Ω of predicted clusterings and demand the scores of all predicted clusterings with respect to a true clustering to be normalized with both extremes attainable. The two above mentioned normalizations only use the score information, the rank information is not explored. Besides, they only care about the scores and fail to consider the scores are achieved by what kinds of clusterings. Therefore, we propose a conditional normalization property based on a three-way partitioning of the set of all possible predicted clusterings conditional on any given true clustering.

Definition 3.0.1 (Conditional Normalization). A measure is conditionally normalized if all predicted clusterings Ω can be partitioned into three categories according to any true clustering $\mathbb{L} \in \Omega$: the best predicted clusterings $\Omega_{\mathbb{L}}^b$ whose scores are ones (resp.

zeros) for similarity (resp. distance) measures, the worst predicted clusterings $\Omega_{\mathbb{L}}^w$ whose scores are zeros (resp. ones) for similarity (resp. distance) measures, and the normal clusterings $\Omega_{\mathbb{L}}^n$ whose similarities are within the open interval $(0, 1)$. They must subject to the following constraints: $n \geq 1$, $\Omega = \Omega_{\mathbb{L}}^b \cup \Omega_{\mathbb{L}}^w \cup \Omega_{\mathbb{L}}^n$, $\Omega_{\mathbb{L}}^b = \{\mathbb{L}\}$ for $n \geq 1$, $\Omega_{\mathbb{L}}^w = \emptyset$ if and only if $n \leq 1$, and $\Omega_{\mathbb{L}}^n = \emptyset$ if and only if $n \leq 2$.

There is only one predicted clustering when $n = 1$. Consequently, it is considered to be the best, and the rest two sets $\Omega_{\mathbb{L}}^w$ and $\Omega_{\mathbb{L}}^n$ should be empty. When $n = 2$, there are only two predicted clusterings. One is the best while the other one is the worst, so $\Omega_{\mathbb{L}}^n$ should be empty. There are more than two predicted clusterings if $n > 2$, resulting the three categories to be nonempty. In summary, given a true clustering $\mathbb{L}$, there is only one best predicted clustering which is always the same with the true clustering $\Omega_{\mathbb{L}}^b = \{\mathbb{L}\}$, there exists at least one worst clustering when $n \geq 2$, and there exists at least one normal clustering when $n \geq 3$. Even from the score point of view, our definition is more stringent than that afforded by [24]. We demand that $\Omega_{\mathbb{L}}^n$ be nonempty for $n \geq 3$. This gives a gradation of scores from one to zero for similarity measures or from zero to one for distance measures as a clustering deteriorates from the best clustering to a worst clustering; it reflects how far a clustering is from the best clustering and the worst clusterings.

We usually use two steps to determine whether a measure is conditionally normalized or not. First, we check whether a measure is normalized. A conditionally normalized measure is also normalized, but not versa. As a result, an unnormalized measure is not conditionally normalized. A measure is not normalized if the two extremes (zero and one) are not reachable. Second, we need to verify whether the three categories of predicted clusterings satisfy the constraints imposed by Definition 3.0.1. Since the set of normal clusterings can be obtained using the formula $\Omega_{\mathbb{L}}^n = \Omega \setminus \Omega_{\mathbb{L}}^b \setminus \Omega_{\mathbb{L}}^w$, we only need to find

the set $\Omega_{\mathbb{L}}^b$ of best clusterings and the set $\Omega_{\mathbb{L}}^w$ of worst clusterings for any true clustering $\mathbb{L}$, which can be derived based on the best/worst clustering condition defined below.

Definition 3.0.2 (Best/Worst Clustering Condition). The best clustering condition of a normalized similarity (resp. distance) measure is a set of ordered pairs $(\mathbb{L}, \mathbb{C})$ whose scores are all ones (resp. zeros), while the set of ordered pairs $(\mathbb{L}, \mathbb{C})$ whose scores are all zeros (resp. ones) is the worst clustering condition.

The best/worst clustering condition is usually expressed as a logical expression. If the logical expression of a best (worst) clustering condition is to evaluate to be true on an ordered pair $(\mathbb{L}, \mathbb{C})$, $\mathbb{C}$ is considered as a best (worst) clustering of $\mathbb{L}$. The best clustering condition of a conditionally normalized measure should be $\mathbb{L} = \mathbb{C}$, which implies $\Omega_{\mathbb{L}}^w = \{\mathbb{L}\}$ for any true clustering. This condition is usually fulfilled by normalized symmetric measures and violated by normalized asymmetric measures. If the worst clustering condition of a normalized measure is false for at least one true clustering when $n > 1$, the measure is not conditionally normalized because $\Omega_{\mathbb{L}}^w$ is empty for some true clusterings. In some situations, the best or worst bound condition is difficult to obtain. If we can find at least a true clustering whose corresponding set of normal predicted clusterings $\Omega_{\mathbb{L}}^n$ is empty when $n > 3$, we can prove a measure is not conditionally normalized. To prove a normalized measure is conditionally normalized, we need to make sure the set $\Omega_{\mathbb{L}}^w$ of worst clusterings associated with every true clustering is not empty when $n > 1$ and is a proper subset of $\Omega \setminus \Omega_{\mathbb{L}}^b$ when $n > 2$.

Chapter 4

A Survey of Existing Measures

A large number of existing similarity and distance measures have been summarized and compared by [28, 48, 47]. As mentioned in the Chapter 3, unnormalized measures are not suitable for comparing clusterings across data sets [47], and we find most measures are based on normalizing a few unnormalized measures. Therefore, we review existing measures from the normalization point of view. Assume an unnormalized similarity measure S is lower bounded by a measure S_l and upper bounded by a measure S_u , a normalized similarity measure $\hat{S}_s$ and a normalized distance measure $\hat{S}_d$ can be obtained using the following formulae

$$\hat{S}_s(\mathbb{L}, \mathbb{C}) = \frac{S(\mathbb{L}, \mathbb{C}) - S_l(\mathbb{L}, \mathbb{C})}{S_u(\mathbb{L}, \mathbb{C}) - S_l(\mathbb{L}, \mathbb{C})}, \quad \hat{S}_d(\mathbb{L}, \mathbb{C}) = \frac{S_u(\mathbb{L}, \mathbb{C}) - S(\mathbb{L}, \mathbb{C})}{S_u(\mathbb{L}, \mathbb{C}) - S_l(\mathbb{L}, \mathbb{C})}. \quad (4.1)$$

Unnormalized distance measures can be normalized in a similar way. For most measures, the lower bound $S_l(\mathbb{L}, \mathbb{C})$ is usually zero, and the upper bound $S_u(\mathbb{L}, \mathbb{C})$ can be expressed as a function of the two trivial upper bounds $S(\mathbb{L}, \mathbb{L})$ and $S(\mathbb{C}, \mathbb{C})$ that can be obtained from the measure $S(\mathbb{L}, \mathbb{C})$ itself. Some examples of symmetric upper bounds are the minimal, the geometric mean, the arithmetic mean, and the maximal of the two trivial upper bounds. The two trivial upper bounds are asymmetric.

Normalizing a measure has some side effects. Suppose that S is symmetric and the lower bound is zero, divided by a symmetric (resp. asymmetric) upper bound yields a

normalized symmetric (resp. asymmetric) measure. Even asymmetric measures can provide some insights into the closeness of clusterings to some extent, they are eventually turned into symmetric measures because they are inappropriate for the general task of comparing clusterings [48]. An asymmetric measure $S(\mathbb{L}, \mathbb{C})$ is always associated with a corresponding asymmetric measure $S(\mathbb{C}, \mathbb{L})$. Symmetric measures can be constructed from asymmetric measures using symmetric functions. A few commonly used functions are the harmonic mean, the geometric mean, and the arithmetic mean of the two asymmetric measures $S(\mathbb{L}, \mathbb{C})$ and $S(\mathbb{C}, \mathbb{L})$.

Existing measures are usually grouped into three categories: pair counting, set matching, and information theoretic measures. Pair counting measures are based on computing the number of matched pairs of objects between two clusterings. Inspired from this, we propose a new category of cluster counting measures which rely on computing the number of matched clusters between two clusterings. According to how clusters are matched, we further group cluster counting measures into two subcategories: exact matching and overlap matching measures. Similarly, we group set matching measures into local matching and global matching measures according to whether the matching is performed locally or globally. One reason for proposing the new categorization scheme is that we find each new category or subcategory has a clear strategy of how the measures are generated, which is quite different from the strategies used by others. The new categorization scheme also allows us to better study and compare measures from the category level. We will review existing measures according to the new categorization scheme. We find all existing measures to be discussed in the thesis are not conditionally normalized, which are derived in Appendix A.

4.1 Pair Counting Measures

Pair counting measures consider a clustering task as a binary classification task based on the fact that any two objects are either in the same cluster or in different clusters. Any two objects forms a pair, resulting in $\binom{n}{2} = n(n-1)/2$ pairs in total. Within a clustering, a pair is positive if its associated two objects are in the same cluster and negative otherwise. In a predicted clustering, a positive (resp. negative) pair is true if it is also positive (resp. negative) in the true clustering, otherwise it is false. The number of positive pairs within a clustering $\mathbb{X}$ is denoted by $P(\mathbb{X}) = \sum_{X \in \mathbb{X}} \binom{|X|}{2}$, and the number of negative pairs within $\mathbb{X}$ is $\binom{n}{2} - P(\mathbb{X})$. Table 4.1 shows the pair information between a true clustering and a predicted clustering. The true (resp. predicted) clustering contains $P(\mathbb{L})$ (resp. $P(\mathbb{C})$) positive pairs and $\binom{n}{2} - P(\mathbb{L})$ (resp. $\binom{n}{2} - P(\mathbb{C})$) negative pairs. The number of true positive pairs is $\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \binom{|L \cap C|}{2}$, which is actually the number of positive pairs $P(\mathbb{M})$ within their meet. The two trivial upper bounds of $P(\mathbb{M})$ are $P(\mathbb{L})$ and $P(\mathbb{C})$. Some commonly used symmetric upper bounds are included in the following inequalities

$$P(\mathbb{M}) \leq \sqrt{P(\mathbb{L})P(\mathbb{C})} \leq (P(\mathbb{L}) + P(\mathbb{C}))/2 \leq P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M}) \leq \binom{n}{2}, \quad (4.2)$$

where $P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M})$ is the number of pairs that are either positive or false. Most pair counting measures are obtained by normalizing $P(\mathbb{M})$ with different upper bounds. They are summarized in Table 4.2 and will be discussed in the rest paragraphs.

Rand Index Rand index $R(\mathbb{L}, \mathbb{C}) = [\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + 2P(\mathbb{M})]/\binom{n}{2}$ is the fraction of the number of true pairs to the total number of pairs [37]. It can be interpreted as the probability that two pairs are treated alike in both clusterings. It highly depends on the number of clusters [48]. The number of true negative pairs increases very quickly when the number of clusters within a predicted clustering increases, so it gives a high score

		The True Clustering $\mathbb{L}$		Sum
		Positive	Negative	
$\mathbb{C}$	Positive	$P(\mathbb{M})$	$P(\mathbb{C}) - P(\mathbb{M})$	$P(\mathbb{C})$
	Negative	$P(\mathbb{L}) - P(\mathbb{M})$	$\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + P(\mathbb{M})$	$\binom{n}{2} - P(\mathbb{C})$
	Sum	$P(\mathbb{L})$	$\binom{n}{2} - P(\mathbb{L})$	$\binom{n}{2}$

Table 4.1: The predicted clustering has $P(\mathbb{C})$ positive pairs and $\binom{n}{2} - P(\mathbb{C})$ negative pairs. With respect to a true clustering $\mathbb{L}$, the $\binom{n}{2}$ pairs can be further divided into $P(\mathbb{M})$ true positive pairs, $P(\mathbb{C}) - P(\mathbb{M})$ false positive pairs, $P(\mathbb{L}) - P(\mathbb{M})$ false negative pairs, and $\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + P(\mathbb{M})$ true negative pairs.

when the predicted clustering is close to the bottom. Assume a true clustering contains 20 clusters with 1000 objects in every cluster, a trivial predicted clustering bottom is misleadingly scored 95% [35]. [10] revealed another similar but undesirable behavior: the score between two unrelated clusterings quickly approaches one as the number of clusters in both clusterings increases, and the scores usually fall into a small interval near one.

Adjusted Rand Index The Rand index is assumed to be zero under an appropriate null model (i.e., the generalized hypergeometric distribution): the two clusterings are drawn randomly subject to the constraints that they have fixed number of clusters and fixed number of objects in each cluster [18]. The adjusted index can be expressed as $\frac{\text{Rand Index} - \text{Expected Rand Index}}{\text{Maximum Rand Index} - \text{Expected Rand Index}} = \frac{P(\mathbb{M}) - P(\mathbb{L})P(\mathbb{C})/\binom{n}{2}}{[P(\mathbb{L}) + P(\mathbb{C})]/2 - P(\mathbb{L})P(\mathbb{C})/\binom{n}{2}}$. The null model is unrealistic because it makes strong assumptions on the distribution [48]. Besides, it is unnormalized because the scores could be negative [28].

Jaccard Index The Jaccard index was original introduced into taxonomy by Jaccard (1908). It is a measure of the degree of overlap between two sets. Given two sets X and Y , the Jaccard index $\frac{|X \cap Y|}{|X \cup Y|}$ is the size of their intersection $|X \cap Y|$ divided by the size of their union $|X \cup Y|$. The Jaccard index between two clusterings is $P(\mathbb{M})/(P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M}))$, which ignores the true negative pairs on purpose [48].

Measure	Formula	N	CN	S
Jaccard Index	$P(\mathbb{M})/[P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M})]$	✓	✗	✓
Pairwise F_1 Measure	$P(\mathbb{M})/\frac{1}{2}[P(\mathbb{L}) + P(\mathbb{C})]$	✓	✗	✓
Fowlkes-Mallows Index	$P(\mathbb{M})/\sqrt{P(\mathbb{L})P(\mathbb{C})}$	✓	✗	✓
Wallace Indexes	$P(\mathbb{M})/P(\mathbb{C}), P(\mathbb{M})/P(\mathbb{L})$	✓	✗	✗
Rand Index	$[\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + 2P(\mathbb{M})]/\binom{n}{2}$	✓	✗	✓
Mirkin Metric	$2[P(\mathbb{L}) + P(\mathbb{C}) - 2P(\mathbb{M})]$	✗	✗	✓
Adjusted Rand Index	$\frac{P(\mathbb{M}) - P(\mathbb{L})P(\mathbb{C})/\binom{n}{2}}{[P(\mathbb{L}) + P(\mathbb{C})]/2 - P(\mathbb{L})P(\mathbb{C})/\binom{n}{2}}$	✗	✗	✓

Table 4.2: The formulae and properties of counting pair measures. N, CN, and S are short hand notations for normalized, conditionally normalized, and symmetric respectively.

Wallace Indexes [49] proposed two normalized but asymmetric measures $P(\mathbb{M})/P(\mathbb{C})$ and $P(\mathbb{M})/P(\mathbb{L})$. In the context of information retrieval, a pair is treated as a document [51, 26]. Given a query, all the positive pairs of $\mathbb{L}$ (resp. $\mathbb{C}$) are the relevant (resp. retrieved) documents. The pairwise precision $P(\mathbb{M})/P(\mathbb{C})$ is the fraction of retrieved documents that are relevant, and the pairwise recall $P(\mathbb{M})/P(\mathbb{L})$ is the fraction of relevant documents that are retrieved [30].

Fowlkes-Mallows Index [10] introduced a normalized measure $P(\mathbb{M})/\sqrt{P(\mathbb{L})P(\mathbb{C})}$ to compare two flat clusterings. It can also be interpreted as the geometric mean $\sqrt{\frac{P(\mathbb{M})}{P(\mathbb{C})} \cdot \frac{P(\mathbb{M})}{P(\mathbb{L})}}$ of the two asymmetric Wallace indexes.

Pairwise F_1 Measure [30] used a pairwise F_1 measure $2P(\mathbb{M})/[P(\mathbb{L}) + P(\mathbb{C})]$ that is the harmonic mean of the pairwise precision and pairwise recall.

Mirkin Metric A distance metric $\sum_{L \in \mathbb{L}} |L|^2 + \sum_{C \in \mathbb{C}} |C|^2 - 2 \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} |L \cap C|^2$ is proposed in [32], which is reformulated as $2[P(\mathbb{L}) + P(\mathbb{C}) - 2P(\mathbb{M})] = n(n-1)(1 - R(\mathbb{L}, \mathbb{C}))$. Divided by n^2 makes it bounded [9, 28]. However, we recommend to normalize it by $n(n-1)$, which makes it equivalent to the Rand index.

The Rand index, the Jaccard index and the Fowlkes-Mallows index are found to behavior consistently with each other in general when the number of clusters, the cluster size, and the dimensionality of the data are varied [31]. Therefore, the Rand index and Wallace index are chosen as the representative symmetric and asymmetric measures to be studied, respectively.

4.2 Cluster Counting Measures

Inspired from pair counting measures, we propose a new category of cluster counting measures that are based on computing the number of matched clusters between two clusterings. It is easy to determine whether two pairs of objects are the same or not, but it takes some efforts to judge whether two clusters are matched or not. Most cluster counting measures are based on normalizing a basic unnormalized symmetric measure $\Phi_b(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \phi_b(L, C)$, where $\phi_b(L, C)$ is one if L and C are matched and zero otherwise. According to how two clusters are matched, cluster counting measures are further divided into two subcategories: exact matching and overlap matching. The cluster counting measures are summarized in Table 4.3.

4.2.1 Extract Matching

The score between two clusters L and C is $[L = C]$, which is one if and only if the two clusters are the same, otherwise it is zero. This matching scheme gives no credit for a predicted cluster that is partially correct. The corresponding basic unnormalized measure $\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} [L = C] = |\mathbb{L} \cap \mathbb{C}|$ is just the size of the intersection between $\mathbb{L}$ and $\mathbb{C}$. It is obvious that it is lower bounded by zero and upper bounded by $|\mathbb{L}|$ and $|\mathbb{C}|$.

Cluster-level Precision/Recall/ F_1 Measure Two asymmetric measures cluster-level precision $|\mathbb{L} \cap \mathbb{C}|/|\mathbb{C}|$ and cluster-level recall $|\mathbb{L} \cap \mathbb{C}|/|\mathbb{L}|$ as well as a symmetric

Measure	Formula	N	CN	S
Cluster-level Precision/Recall	$ \mathbb{L} \cap \mathbb{C} / \mathbb{C} , \mathbb{L} \cap \mathbb{C} / \mathbb{L} $	✓	✗	✗
Cluster-level F_1 Measure	$2 \mathbb{L} \cap \mathbb{C} /(\mathbb{L} + \mathbb{C})$	✓	✗	✓
MUC Precision/Recall	$(n - \mathbb{M})/(n - \mathbb{C}), (n - \mathbb{M})/(n - \mathbb{L})$	✓	✗	✗
MUC F_1 Measure	$2(n - \mathbb{M})/(2n - \mathbb{L} - \mathbb{C})$	✓	✗	✓
Normalized Split Distance	$(\mathbb{M} - \mathbb{C})/(n - 1)$	✓	✗	✗
Normalized Merge Distance	$(\mathbb{M} - \mathbb{L})/(n - 1)$	✓	✗	✗
Basic Merge Distance	$2 \mathbb{M} - \mathbb{L} - \mathbb{C} $	✗	✗	✓

Table 4.3: The formulae and properties of cluster counting measures. N, CN, and S are short hand notations for normalized, conditionally normalized, and symmetric respectively.

cluster-level F_1 measure $2|\mathbb{L} \cap \mathbb{C}|/(|\mathbb{L}| + |\mathbb{C}|)$ are used to evaluate entity resolution tasks [50, 17, 41, 30]. The cluster-level precision (resp. recall) indicates the ratio of exactly matched clusters to the size of the predicted (resp. true) clustering, that is why their scores are coarse granular and very low. They ignore cluster sizes completely and treat each cluster instead of each object equally.

4.2.2 Overlap Matching

The score between two clusters L and C is $[L \cap C \neq \emptyset]$, which is one if the two clusters overlap and zero otherwise. Different from the exact matching scheme, the overlap matching scheme gives a full credit to a predicted cluster even it is partially correct. The corresponding basic unnormalized distance measure $\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} [L \cap C \neq \emptyset] = |\mathbb{M}|$ is the size of the meet between $\mathbb{L}$ and $\mathbb{C}$. It can be normalized using the following inequalities

$$1 \leq \min\{|\mathbb{L}|, |\mathbb{C}|\} \leq (|\mathbb{L}|, |\mathbb{C}|) \leq \max\{|\mathbb{L}|, |\mathbb{C}|\} \leq |\mathbb{M}| \leq n. \quad (4.3)$$

MUC Precision/Recall/ F_1 Measure If $|\mathbb{M}|$ is normalized using a lower bound $|\mathbb{C}|$ (resp. $|\mathbb{L}|$) and an upper bound n , it becomes the MUC precision $(n - |\mathbb{M}|)/(n - |\mathbb{C}|)$ (resp. recall $(n - |\mathbb{M}|)/(n - |\mathbb{L}|)$) [46]. In coreference resolution, an object is a noun

phrase which may refer to another noun phrase. Coreferent noun phrases are grouped together. A cluster X is weakly linked by $|X| - 1$ coreferent links to form a tree structure. $n - |\mathbb{M}|$, $n - |\mathbb{C}|$, and $n - |\mathbb{L}|$ are the numbers of coreferent links in the meet, the predicted clustering, and the true clustering, respectively. The symmetric MUC F_1 measure $2(n - |\mathbb{M}|)/(2n - |\mathbb{L}| - |\mathbb{C}|)$ is often used as the final evaluation measure. The MUC measures are said to ignore singletons and lack of the desired discriminativity.

Normalized Split/Merge Distance A predicted clustering can be transformed into a true clustering by applying a series of binary split operations followed by a series of binary merge operations, where a binary split operation splits one cluster into two disjoint non-empty clusters and a binary merge operation is its reverse operation [1]. A predicted cluster C overlaps with $\sum_{L \in \mathbb{L}} [L \cap C \neq \emptyset]$ true clusters, so $\sum_{L \in \mathbb{L}} [L \cap C \neq \emptyset] - 1$ binary split operations are required to split C . The number of binary split operations to split $\mathbb{C}$ into $\mathbb{M}$ is the split distance $\sum_{C \in \mathbb{C}} (\sum_{L \in \mathbb{L}} [L \cap C \neq \emptyset] - 1) = |\mathbb{M}| - |\mathbb{C}|$, the number of binary merge operations to merge $\mathbb{M}$ into $\mathbb{L}$ is the merge distance $|\mathbb{M}| - |\mathbb{L}|$, and the total number of binary operations is the basic merge distance $2|\mathbb{M}| - |\mathbb{L}| - |\mathbb{C}|$ [30]. The split and merge distances are bounded between 0 and $n - 1$, based on which [1] proposed the normalized split distance $(|\mathbb{M}| - |\mathbb{C}|)/(n - 1)$ and the normalized merge distance $(|\mathbb{M}| - |\mathbb{L}|)/(n - 1)$.

The MUC measures and the split/merge distance are closely related. One can be considered as the counterpart of the other. If the split distance $|\mathbb{M}| - |\mathbb{C}|$ is normalized using bounds 0 and $n - |\mathbb{C}|$, it becomes the distance counterpart of MUC precision because $(|\mathbb{M}| - |\mathbb{C}|)/(n - |\mathbb{C}|) = 1 - (n - |\mathbb{M}|)/(n - |\mathbb{C}|)$. The same conclusion can be obtained for merge distance and MUC recall. If the basic merge distance is normalized using bounds 0 and $2n - |\mathbb{L}| - |\mathbb{C}|$, it becomes the distance counterpart of MUC F_1 measure because $(2|\mathbb{M}| - |\mathbb{L}| - |\mathbb{C}|)/(2n - |\mathbb{L}| - |\mathbb{C}|) = 1 - (2n - 2|\mathbb{M}|)/(2n - |\mathbb{L}| - |\mathbb{C}|)$.

Measure	Formula	N	CN	S
Purity	$U(\mathbb{L}, \mathbb{C}) = \sum_{C \in \mathbb{C}} \frac{ C }{n} \max_{L \in \mathbb{L}} \frac{ L \cap C }{ C }$	✗	✗	✗
GK Association Index	$(U(\mathbb{L}, \mathbb{C}) - \max_{L \in \mathbb{L}} \frac{ L }{n}) / (1 - \max_{L \in \mathbb{L}} \frac{ L }{n})$	✓	✗	✗
Van Dongen Metric	$2n(1 - (U(\mathbb{L}, \mathbb{C}) + U(\mathbb{C}, \mathbb{L}))/2)$	✗	✗	✓
F-Purity	$\sum_{C \in \mathbb{C}} \frac{ C }{n} \max_{L \in \mathbb{L}} \frac{2 L \cap C }{ L + C }$	✗	✗	✗
Closest Cluster Precision	$\sum_{C \in \mathbb{C}} \frac{1}{ \mathbb{C} } \max_{L \in \mathbb{L}} \frac{ L \cap C }{ L \cup C }$	✗	✗	✗
Classification Accuracy	$\max_{\hat{w}} \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \hat{w}(L, C) \frac{ L \cap C }{n}$	✗	✗	✓
ϕ_4 -CEAF Precision	$\frac{1}{ \mathbb{C} } \max_{\hat{w}} \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \hat{w}(L, C) \frac{2 L \cap C }{ L + C }$	✗	✗	✗

Table 4.4: The formulae and properties of set matching measures. For a pair of asymmetric measures, we only list the measure on the predicted clustering due to space limitations. Goodman-Kruschal is called GK for short. N, CN, and S are short hand notations for normalized, conditionally normalized, and symmetric respectively.

4.3 Set Matching Measures

Within this Subsection, a predicted cluster is called a cluster for short and a true cluster is called a category. Set matching measures considers a clustering task as a multiclass classification problem, which is the problem of classifying clusters into more than two categories. We group set matching measures into two types: local matching and global matching. Every cluster is classified to exact one category under local matching, while every cluster is classified to at most one category under global matching. The set matching measures are summarized in Table 4.4. Most set matching measures are asymmetric and unnormalized. They also ignore unmatched clusters and categories [39]. In addition, most of them can be interpreted from the information retrieval point of view. That is why the F_1 measure is often used.

4.3.1 Local Matching

Local matching requires every cluster to be classified to the category which maximizes their similarity. They are usually asymmetric because the roles of a predicted clustering and a true clustering can be exchanged. Let $\Phi_c(\mathbb{L}, \mathbb{C})$ be the similarity of a predicted

clustering with respect to a true clustering and $\Phi_l(\mathbb{C}, \mathbb{L})$ be the similarity of the true clustering with respect to the predicted clustering. They are defined as follows

$$\Phi_c(\mathbb{L}, \mathbb{C}) = \sum_{C \in \mathbb{C}} w_c(C) \max_{L \in \mathbb{L}} \phi(C, L), \quad \Phi_l(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} w_c(L) \max_{C \in \mathbb{C}} \phi(L, C), \quad (4.4)$$

where $w_c : D \rightarrow \mathbb{R}_{\geq 0}$ is a weighting function that distributes non-negative weights to clusters, and ϕ is a cluster-cluster similarity measure. The two asymmetric measures result from applying a measure to two ordered clustering pairs $(\mathbb{L}, \mathbb{C})$ and $(\mathbb{C}, \mathbb{L})$, so they are related by $\Phi_l(\mathbb{L}, \mathbb{C}) = \Phi_c(\mathbb{C}, \mathbb{L})$ and $\Phi_c(\mathbb{L}, \mathbb{C}) = \Phi_l(\mathbb{C}, \mathbb{L})$. From now on, we focus on studying $\Phi_c(\mathbb{L}, \mathbb{C})$. $\max_{L \in \mathbb{L}} \phi(L, C)$ is the score of a predicted cluster C with respect to the true clustering, and $\Phi_c(\mathbb{L}, \mathbb{C})$ is just the weighted sum of the scores of predicted clusters. It is lower bounded by $\max_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} w_c(C) \phi(L, C)$ that classifies all clusters into the dominant category, and it is upper bounded by the self-similarity $\Phi_c(\mathbb{C}, \mathbb{C})$ of the predicted clustering. The lower bound is often positive and complicated. Most local matching measures use an unreachable zero as the lower bound when performing normalization, so most of them are unnormalized.

A simple weighting function w_c^c assigns a constant one to every cluster such that the scores of clusters are directed added. A weighting function is convex if the sum of the cluster weights is one. A convex weighting function w_c^s uses the reciprocal of a clustering's size (i.e., $1/|\mathbb{C}|$) as the weight for every cluster, which ignores the size of each cluster. Instead, another weighting function w_c^p uses the relative size of a cluster as its weight (i.e., $|C|/n$). A basic similarity between two clusters is the size of their intersection. Many cluster-cluster similarity measures can be obtained by normalizing $|L \cap C|$ using various upper bounds shown in the following inequalities

$$|L \cap C| \leq (|L|, |C|) \leq 2|L||C|/(|L| + |C|) \leq |L \cup C| \leq n. \quad (4.5)$$

Purity and Inverse Purity Let the similarity $\phi(C, L)$ be $|L \cap C|/|C|$ and the weighing function be w_c^p . The score $\max_{L \in \mathbb{L}} |L \cap C|/|C|$ of a cluster C is called the cluster purity, which is the fraction of the objects belonging to the dominant category. The purity $U(\mathbb{L}, \mathbb{C})$ is the weighted average $\sum_{C \in \mathbb{C}} \frac{|C|}{n} \max_{L \in \mathbb{L}} \frac{|L \cap C|}{|C|}$ of cluster purities [54, 6]. It is lower bounded by $\max_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|C|}{n} \frac{|L \cap C|}{|C|} = \max_{L \in \mathbb{L}} \frac{|L|}{n}$, which is always positive. So purity is unnormalized. It penalizes noises in clusters but fails to reward grouping objects from the same category together [52, 2]. If the predicted clustering refines the true clustering, the purity is always one. The purity of the true clustering with respect to the predicted clustering is called inverse purity. From information retrieval point of view, purity is precision and inverse purity is recall. Their harmonic mean is a symmetric measure [3].

Goodman-Kruskal Association Index If a cluster C is classified into the category that has the maximal probability, the probability of misclassification is $1 - \max_{L \in \mathbb{L}} |L \cap C|/|C|$. The expected probability of misclassification is $1 - U(\mathbb{L}, \mathbb{C})$ [40, 19], which is the distance counterpart of the purity. Without using the knowledge of the predicted clustering, the best strategy is to classify all clusters into the dominant category, resulting in a probability $1 - \max_{L \in \mathbb{L}} \frac{|L|}{n}$ of misclassification. The relative reduction in the expected probability of misclassification is the Goodman-Kruskal association index $(U(\mathbb{L}, \mathbb{C}) - \max_{L \in \mathbb{L}} \frac{|L|}{n}) / (1 - \max_{L \in \mathbb{L}} \frac{|L|}{n})$ [12, 13, 14, 19]. It is normalized version of the purity.

Van Dongen Metric [9] defined the distance between two clusterings as the minimal number of object moves, which can be accomplished by transforming the predicted clustering to the meet followed by transforming the meet to the true clustering. The minimal number of object moves for a cluster C is $|C| - \max_{L \in \mathbb{L}} |L \cap C|$. Therefore, the total number of object moves is a metric $2n - \sum_{L \in \mathbb{L}} \max_{C \in \mathbb{C}} |L \cap C| - \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} |L \cap C|$,

which can be rewritten as $2n(1 - (U(\mathbb{L}, \mathbb{C}) + U(\mathbb{C}, \mathbb{L}))/2)$. The metric is constrained instead of normalized to a right-open interval $[0, 1)$ when divided by $2n$ [28], which becomes the distance counterpart of the arithmetic mean of purity and inverse purity.

F-Purity and F-Inverse Purity If the F_1 measure $\frac{2|L \cap C|}{|L| + |C|}$ is used as the cluster-cluster similarity measure and w_c^p is selected as the weighing function, the resulting similarity measure $\sum_{C \in \mathbb{C}} \frac{|C|}{n} \max_{L \in \mathbb{L}} \frac{2|L \cap C|}{|L| + |C|}$ is the F-purity [22, 42, 11, 6]. It was originally designed to compare hierarchical clusterings before it was used to compare flat clusterings [48, 6]. The F-purity of the true clustering with respect to the predicted clustering is called F-inverse purity.

Closest Cluster Precision/Recall The closest cluster precision $\sum_{C \in \mathbb{C}} \frac{1}{|\mathbb{C}|} \max_{L \in \mathbb{L}} \frac{|L \cap C|}{|L \cup C|}$ uses the Jaccard index $\frac{|L \cap C|}{|L \cup C|}$ as the cluster-cluster similarity measure and w_c^s as the weighing function [5, 30]. The closest cluster precision of the true clustering with respect to the predicted clustering is the closest cluster recall.

4.3.2 Global Matching

Local matching permits many-to-one matching where different clusters can be classified to the same category because every cluster makes a local decision without being influenced by other clusters. Instead, global matching makes a global decision to classify a cluster to at most one category, resulting in some clusters or categories being unused in some situations. A basic global matching measure $\Phi_g(\mathbb{L}, \mathbb{C})$ is formulated as a maximum

weighted bipartite matching problem

$$\begin{aligned}
 & \max_{\hat{w}} \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \hat{w}(L, C) \phi(L, C) \\
 & \text{subject to } \sum_{C \in \mathbb{C}} \hat{w}(L, C) \leq 1 \quad \forall L \in \mathbb{L} \\
 & \sum_{L \in \mathbb{L}} \hat{w}(L, C) \leq 1 \quad \forall C \in \mathbb{C} \\
 & \hat{w}(L, C) \in \{0, 1\} \quad \forall L \in \mathbb{L}, C \in \mathbb{C}
 \end{aligned} \tag{4.6}$$

where the binary variable $\hat{w}(L, C)$ cluster C is classified to category L . The problem can be solved by the Kuhn-Munkres algorithm [20, 33, 27]. $\Phi_g(\mathbb{L}, \mathbb{C})$ is symmetric but could be unnormalized. It is upper bounded by both the self-similarity $\Phi_g(\mathbb{C}, \mathbb{C})$ of the predicted clustering and the self-similarity $\Phi_g(\mathbb{L}, \mathbb{L})$ of the true clustering. The lower bound is positive and complicated. Most global matching measures are often unnormalized because an unreachable zero is used as the lower bound when performing normalization for simplicity.

Classification Accuracy Using $|L \cap C|/n$ as the cluster-cluster similarity measure yields the classification accuracy, which computes the fraction of the objects that are correctly classified [29]. It is the same with the Φ_3 -CEAF precision/recall proposed by [25]. But is unnormalized because its lower bound is $1/n$.

ϕ_4 -CEAF Precision/Recall/ F_1 Measure An unnormalized symmetric measure can be obtained by using the F_1 cluster-cluster measure $2|L \cap C|/|L| + |C|$. Divided by its upper bounds $|\mathbb{C}|$ and $|\mathbb{L}|$ yields the ϕ_4 -CEAF precision and ϕ_4 -CEAF recall, respectively [25]. Their harmonic mean is the ϕ_4 -CEAF F_1 measure.

4.4 Information Theoretic Measures

Information theoretic measures are developed based on the concept of information entropy, some of which are summarized in Table 4.5. $\log$ represents the natural logarithm.

Measure	Formula	N	CN	S
Conditional Entropy	$H(\mathbb{C} \mathbb{L}), H(\mathbb{L} \mathbb{C})$	✗	✗	✗
Normalized Conditional Entropy	$H(\mathbb{C} \mathbb{L})/\log \mathbb{C} , H(\mathbb{L} \mathbb{C})/\log \mathbb{L} $	✓	✗	✗
Mutual Information	$I(\mathbb{L}, \mathbb{C})$	✗	✗	✓
Asymmetric Measures [39]	$I(\mathbb{L}, \mathbb{C})/H(\mathbb{C}), I(\mathbb{L}, \mathbb{C})/H(\mathbb{L})$	✓	✗	✗
[43]	$I(\mathbb{L}, \mathbb{C})/\frac{1}{2}[\log \mathbb{L} + \log \mathbb{C}]$	✓	✗	✓
Minimum [21]	$I(\mathbb{L}, \mathbb{C})/\min\{H(\mathbb{L}), H(\mathbb{C})\}$	✓	✗	✓
Geometric Mean [44]	$I(\mathbb{L}, \mathbb{C})/\sqrt{H(\mathbb{L})H(\mathbb{C})}$	✓	✗	✓
Arithmetic Mean [39]	$I(\mathbb{L}, \mathbb{C})/\frac{1}{2}[H(\mathbb{L}) + H(\mathbb{C})]$	✓	✗	✓
Maximum [21]	$I(\mathbb{L}, \mathbb{C})/\max\{H(\mathbb{L}), H(\mathbb{C})\}$	✓	✗	✓
Joint Entropy [53]	$I(\mathbb{L}, \mathbb{C})/H(\mathbb{M})$	✓	✗	✓
Variation of Information [28]	$H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C})$	✗	✗	✓
VI divided by $\log n$ [28]	$(H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}))/\log n$	✓	✗	✓
VI divided by $\log k^2$ [28]	$(H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}))/\log k^2$	✓	✗	✓

Table 4.5: The formulae and properties of information theoretic measures. VI is the short hand notation for variation of information. N, CN, and S are short hand notations for normalized, conditionally normalized, and symmetric respectively.

The entropy $H(\mathbb{C})$ of a predicted clustering is $-\sum_{C \in \mathbb{C}} \frac{|C|}{n} \log \frac{|C|}{n}$, which is the uncertainty in guessing which cluster a randomly drawn object belongs to [8]. $H(\mathbb{L})$ is the entropy of a true clustering. The joint entropy between $\mathbb{L}$ and $\mathbb{C}$ is $-\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \log \frac{|L \cap C|}{n}$, which is actually the entropy $H(\mathbb{M})$ of their meet. Their mutual information $I(\mathbb{L}, \mathbb{C}) = H(\mathbb{L}) + H(\mathbb{C}) - H(\mathbb{M})$ is the amount of information that one clustering contains about another clustering. It is upper bounded by two trivial asymmetric bounds $H(\mathbb{L})$ and $H(\mathbb{C})$. Some frequently used symmetric upper bounds are given in the following inequalities

$$\begin{aligned}
 I(\mathbb{L}, \mathbb{C}) &\leq \min\{H(\mathbb{L}), H(\mathbb{C})\} \leq \sqrt{H(\mathbb{L})H(\mathbb{C})} \\
 &\leq (H(\mathbb{L}) + H(\mathbb{C}))/2 \leq \max\{H(\mathbb{L}), H(\mathbb{C})\} \leq H(\mathbb{M}) \leq \log n.
 \end{aligned} \tag{4.7}$$

The entropy of $\mathbb{C}$ conditioned on $\mathbb{L}$ is the conditional entropy $H(\mathbb{C}|\mathbb{L}) = H(\mathbb{C}) - I(\mathbb{L}, \mathbb{C})$ that represents the uncertainty of $\mathbb{C}$ given that $\mathbb{L}$ is known. Similarly, $H(\mathbb{L}|\mathbb{C})$ is the entropy of $\mathbb{L}$ conditioned on $\mathbb{C}$.

Conditional Entropy [42] used $H(\mathbb{C}|\mathbb{L})$ as a distance measure to evaluate the performances of different document clusterings algorithms. A smaller conditional entropy implies better performance. A normalized variant $H(\mathbb{C}|\mathbb{L})/\log |\mathbb{C}|$ is used by [54, 55]. The conditional entropy and its normalized variant suffer from the same problem as purity [52]. The conditional entropy reflects the homogeneity of the predicted clustering and it favors a large number of clusters. If the predicted clustering refines the true clustering, the conditional entropy is always zero.

Mutual Information [4] used the mutual information as the performance evaluation measure. Unlike pair counting and set matching measures, it takes higher order dependencies into account. It measures the degree of statistical dependency between two clusterings, which is not always consistent with their similarity. For example, $I(\mathbb{T}, \mathbb{C}) = 0$ means there is no dependency between the top and any predicted clustering $\mathbb{C} \in \Omega$, but the actual similarity depends on the closeness of the top to the predicted clustering. Another example that the mutual information fails to rank two predicted clusterings correctly with respect to a true clustering was given by [47].

Normalized Mutual Information Two asymmetric measures can be obtained by normalizing the mutual information using the two trivial upper bounds $H(\mathbb{C})$ and $H(\mathbb{L})$, respectively [39]. Their harmonic mean $2I(\mathbb{L}, \mathbb{C})/(H(\mathbb{L}) + H(\mathbb{C}))$ is a special case of the V-measure introduced by [39]. The mutual information is divided by $(\log |\mathbb{L}| + \log |\mathbb{C}|)/2$ [45, 43]. It is normalized using $\sqrt{H(\mathbb{L})H(\mathbb{C})}$ because the analogy of geometric mean with a normalized inner product in Hilbert space [44]. The minimum and maximum of the two entropies can be used to normalize $I(\mathbb{L}, \mathbb{C})$ [21]. [53] normalized the mutual information using the joint entropy $H(\mathbb{M})$. After comparing various normalized variants, [47] advocated to use $I(\mathbb{L}, \mathbb{C})/\max\{H(\mathbb{L}), H(\mathbb{C})\}$ because it uses the range $[0, 1]$ better than others.

Variation of Information and Normalized Variants $H(\mathbb{L}|\mathbb{C})$ is the amount of information to be discarded and $H(\mathbb{C}|\mathbb{L})$ is the amount of information to be gained when transforming the true clustering to the predicted clustering through the meet. The total amount of information involved is the variation of information $H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C})$ [28]. Although the variation of information has many desirable properties, it is unnormalized. When divided by the upper bound $\log n$, the resulting measure is normalized but not suitable for comparing clusterings across data sets due to the dependence on n [28]. When both the two clusterings have at most $k \leq \sqrt{n}$ clusters, the variation of information is upper bounded by $2 \log k$ [28]. The normalized variant $(H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C})) / \log k^2$ is suitable for comparing clusterings across data sets, but applying this measure requires knowing k in advance [28]. [38] obtained two asymmetric measures based on the variation of information. However, its range is unbounded instead of normalized. The predicted clustering whose similarities lie in $[0, 1]$ are considered to be good.

Chapter 5

A Generative Model for Measures

Broadly speaking, a measure is a measurement on two objects (i.e., cluster, clustering). We propose a generative model for measures. It consists of two parts: aspect and operation. A measure has many properties, we select a few representative properties and organize them into four aspects in Section 5.1. Since normalized symmetric similarity measures are the focus of this thesis, some properties of each aspect are more preferred over other properties of the same aspect. In Section 5.2, we then design four operations corresponding to the four aspects so as to change most properties of a measure to more desirable properties. We apply the operations to generate most existing measures in Section 5.3. Note that the preference of properties is application specific, which further requires the operations to be designed to take the preference information into account. Our model gives an example of how to choose the desired properties and how to design the operations accordingly. Similar ideas can be employed to design distance measures.

5.1 Aspect

Combination Aspect The combination aspect reflects the complexity of a measure. Let 2^D be the power set of data set D , which is a set of all the subsets of D , including empty set and D itself. The size of the power set 2^D is $2^{|D|}$. Denote $X \times Y$ as the Cartesian

product of two sets X and Y that is the set of all ordered pairs $\{(x, y) \mid x \in X, y \in Y\}$. The size of $X \times Y$ is $|X| \times |Y|$. Let $\mathbb{R}$ refer to the set of all real numbers.

- A *cluster-cluster* measure α is a function $\alpha : 2^D \times 2^D \rightarrow \mathbb{R}$ that computes the similarity or distance between two clusters. Given two clusters L and C , $\alpha(L, C)$ is their similarity or distance from L to C , and $\alpha(C, L)$ is their similarity or distance from C to L .
- A *cluster-clustering* measure β is a function $\beta : 2^D \times \Omega \rightarrow \mathbb{R}$ that computes the similarity or distance between a cluster and a clustering. If the cluster is a true cluster $L \in \mathbb{L}$, the cluster-clustering similarity or distance is $\beta(L, \mathbb{C})$. If the cluster is a predicted cluster $C \in \mathbb{C}$, the cluster-clustering similarity or distance is $\beta(C, \mathbb{L})$.
- A *clustering-clustering* measure γ is a function $\gamma : \Omega \times \Omega \rightarrow \mathbb{R}$ computes the similarity or distance between two clusterings. $\gamma(\mathbb{L}, \mathbb{C})$ is the similarity or distance from $\mathbb{L}$ to $\mathbb{C}$, and $\gamma(\mathbb{C}, \mathbb{L})$ is the similarity or distance of from $\mathbb{C}$ to $\mathbb{L}$.

Similarization Aspect The similarization aspect indicates whether a measure is a similarity measure or a distance measure. A larger similarity or a smaller distance between two objects means they are more similar, and vice versa.

Normalization Aspect The normalization aspect contains information about whether a measure is normalized or not. A measure is normalized if the range of similarities or distances is the closed interval $[0, 1]$ and unnormalized otherwise.

Symmetrization Aspect The symmetrization aspect tells whether a measure is symmetric or asymmetric. A cluster-cluster (resp. clustering-clustering) measure is symmetric if the similarity or distance remains the same after exchanging the two clusters (resp. clusterings) and asymmetric otherwise. A cluster-clustering measure is considered to be asymmetric.

Aspect	Ordered Properties
Combination	cluster-cluster $\rightarrow$ cluster-clustering $\rightarrow$ clustering-clustering
Similarization	distance $\rightarrow$ similarity
Normalization	unnormalized $\rightarrow$ normalized
Symmetrization	asymmetric $\rightarrow$ symmetric

Table 5.1: The four aspects and their properties. The properties within each aspect are ordered from the least undesirable to the most desirable.

Ordered Properties Among the properties of an aspect, some properties are more favorable than others. Whether a property is more favorable than another property is application specific. Starting from a cluster-cluster measure, a cluster-clustering measure can be combined, and then a clustering-clustering measure can be combined. The sequence of the combination process provides a natural order of them. Since we focus on studying similarity measures, we favor similarity measures over distance measures. A normalized measure is preferred over an unnormalized measure because normalized measures facilitate comparisons across data sets. A symmetric measures provides a single criterion for evaluating clusterings and asymmetric measures are usually turned into a symmetric measure. As a result, a symmetric measure is more desirable than an asymmetric measure. The preference of the properties within each aspect are summarized in Table 5.1.

5.2 Operation

To convert a measure to any other measure, we need to perform a series of operations. Corresponding to the four aspects, we define four basic operations: combination operation, similarization operation, normalization operation, and symmetrization operation. The primary purpose of an operation is to change a property of its corresponding aspect to the next desirable property (see Table 5.1). For instance, the combination operation transforms a cluster-cluster measure to a cluster-clustering measure, or transforms a

Operation \ Aspect	Combination	Similarization	Normalization	Symmetrization
Combination	α, β	*	*	*
Similarization	*	distance	*	*
Normalization	*	similarity	unnormalized	*
Symmetrization	*	similarity	*	asymmetric

Table 5.2: The preconditions of operations in terms of properties. The properties that are not most desirable are in the diagonal direction. A star (*) indicates any property of an aspect.

cluster-clustering measure to a clustering-clustering measure. When to apply an operation (the precondition) and what kind of effects an operation is designed to achieve (the postcondition) are studied in Section 5.2.1. The question of how to perform an operation is solved by considering an operation as a function taking some extra parameters. The four operations are then studied in detail.

5.2.1 Precondition and Postcondition

Precondition What kind of properties should a measure satisfy before performing an operation on it? This is the same with asking what is the precondition of each operation in terms of properties. We try to put less restrictions on the preconditions of operations and list them in Table 5.2. A combination operation can be applied to all cluster-cluster and cluster-clustering measures. A similarization operation turns a distance measure into a similarity measure. To reduce the complexity of operations without compromising the flexibility, the rest two operations are restricted to similarity measures. Normalization operations and symmetrization operations are used on unnormalized similarity measures and asymmetric measures, respectively. If we want to turn an unnormalized (resp. asymmetric) distance measure to a normalized (resp. symmetric) similarity measure, a similarization operation can be applied followed by a normalization (resp. symmetrization) operation.

Postcondition What is the function of each operation? When designing the operations, we try to keep them simple, orthogonal yet flexible. An operation is designed to upgrade only one aspect, but it may bring side effects by changing other aspects. The side effects are flexible in generating many types of measures. They, however, can also be harmful. First, there are many ways to create a measure, making the design of operations complicated and making the functions of operations highly overlapped. Second, we may encounter loops when applying operations. Taking all the factors into account, we consider the combination operation as the primary operation and the rest as secondary operations. A combination operation is designed to be as flexible as possible. It is allowed to change the rest aspects as needed. The rest operations are mainly used to reduce the side effects caused by a combination operation. Hence, they should change as less aspects as possible to avoid introducing more adverse effects. Only a combination operation can create distance measures, and we can use a similarization operation to turn them into similarity measures without changing other properties. The combination operation is the only operation that creates unnormalized measures, and normalization operations can make them normalized. In some domains (i.e., coreference resolution), asymmetric measures can provide additional insight into the closeness of two clusterings. As a result, a normalization operation can change the symmetrization aspect. A symmetrization operation makes asymmetric measures symmetric. When it is applied to asymmetric cluster-cluster or clustering-clustering measures, the combination aspect is preserved. However, a symmetric cluster-cluster measure is obtained when it is performed on cluster-clustering measures, which is only allowed to use at most once since it may create loops. The postcondition of each operation in terms of properties are summarized in Table 5.3.

Operation \ Aspect	Combination	Similarization	Normalization	Symmetrization
Combination	β, γ	?	?	?
Similarization	✓	similarity	✓	✓
Normalization	✓	✓	normalized	?
Symmetrization	α, γ	✓	✓	symmetric

Table 5.3: The postconditions of operations in terms of properties. The properties that are not least desirable are in the diagonal direction. A check mark (✓) means the property is preserved. A question mark (?) indicates the property may be changed or preserved.

5.2.2 Combination Operation

A *combination operation* is an aggregation function $f_c : \mathbb{R}^{|\mathbb{X}|} \rightarrow \mathbb{R}$ over a clustering $\mathbb{X}$. A simple aggregation function is a weighted-sum function $\mathbf{f}_c^s(\mathbb{X}, \mathbf{m}, \mathbf{w}) = \sum_{\mathbf{X} \in \mathbb{X}} \mathbf{w}(\mathbf{X}) \mathbf{m}(\mathbf{X})$, where $w : \mathbb{X} \rightarrow \mathbb{R}^+$ is a weighting function that assigns weight $w(X)$ to cluster X , and $m(X)$ is the score associated with cluster X . A binary weighting function assigns binary weight $w(X) \in \{0, 1\}$ to every cluster X . A conic weighting function uses nonnegative weight $w(X) \geq 0$ for every cluster X . A convex weighting function associates every cluster X with a nonnegative weight $w(X) \geq 0$ under the constraint that the sum of their weights $\sum_{X \in \mathbb{X}} w(X)$ is one.

To create a clustering-clustering measure from a cluster-cluster measure, we have to apply two combination operations. Between the two combination operations, other valid operations can be used. It is obvious that the weighting function is the essential part of a combination operation. If the weighting function used in combining cluster-cluster measures has nothing to do with the weighting function used in combining cluster-clustering measures, the two combinations are said to be independent. Otherwise, they are called joint combinations. Most combinations are independent.

When determining the weight for a cluster, some contextual information ($\mathbb{X}$ and m) are often used. An example of the conic weighting is a simple weighting function $w_s(X) = 1$. No contextual information is used since all clusters have equal weights.

The scores of each cluster are simply added. An example of the convex weighting is a probabilistic weighting function $w_p(X) = |X \cap Y|/|Y|$, where Y is a reference cluster or the data set D . It uses the cluster information (Y). An example of the binary weighting is a maximum weighting function $w_m(X) = [X = \arg \max_{X \in \mathbb{X}} m(X, Y)]$, where Y is a reference cluster or a clustering. It uses the clustering information ($\mathbb{X}$). The combination function can be simplified as $\max_{X \in \mathbb{X}} m(X, Y)$. Table 5.4 summarizes their specific weights under different measures m and clusterings $\mathbb{X}$.

A cluster-cluster measure α can be combined over both clusterings, a cluster-clustering measure $\beta(L, \mathbb{C})$ can only be combined over the true clustering, and a cluster-clustering measure $\beta(C, \mathbb{L})$ can only be combined over the predicted clustering. The four possible independent combinations are shown below.

$$\begin{aligned} \beta(L, \mathbb{C}) = \sum_{C \in \mathbb{C}} w(C) \alpha(L, C) & \quad \rightarrow \quad \gamma(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} w(L) \beta(L, \mathbb{C}) \\ \beta(C, \mathbb{L}) = \sum_{L \in \mathbb{L}} w(L) \alpha(L, C) & \quad \rightarrow \quad \gamma(\mathbb{C}, \mathbb{L}) = \sum_{C \in \mathbb{C}} w(C) \beta(C, \mathbb{L}) \end{aligned}$$

The weights used in the cluster-cluster and cluster-clustering combination operations are determined separately. A joint combination determines the weights jointly. It actually uses all the available information (the true and predicted clustering).

$$\begin{aligned} \gamma(\mathbb{L}, \mathbb{C}) &= \max_w \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} w(L, C) \alpha(L, C) \\ \text{subject to } &\forall L \in \mathbb{L} \forall C \in \mathbb{C} : w(L, C) \in \{0, 1\}, \\ &\forall L \in \mathbb{L} : \sum_{C \in \mathbb{C}} w(L, C) \leq 1, \quad \forall C \in \mathbb{C} : \sum_{L \in \mathbb{L}} w(L, C) \leq 1. \end{aligned} \tag{5.1}$$

5.2.3 Similarization Operation

A *similarization operation* is a function $f_i : \mathbb{R}^2 \rightarrow \mathbb{R}$ such that $f_i(x, u)$ is a decreasing function on x under the constraint $x \leq u$. A simple similarization function is $\mathbf{f}_1^s(\mathbf{x}, \mathbf{u}) = \mathbf{u} - \mathbf{x}$,

Measure m	Clustering $\mathbb{X}$	Weighting Scheme for $w(X)$		
		Simple w_s	Probabilistic w_p	Maximum w_m
$\alpha(L, C)$	$\mathbb{L}$	1	$ L \cap C / C $	$[L = \arg \max_{L \in \mathbb{L}} \alpha(L, C)]$
$\alpha(C, L)$	$\mathbb{C}$	1	$ L \cap C / L $	$[C = \arg \max_{C \in \mathbb{C}} \alpha(C, L)]$
$\beta(L, \mathbb{C})$	$\mathbb{L}$	1	$ L /n$	$[L = \arg \max_{L \in \mathbb{L}} \beta(L, \mathbb{C})]$
$\beta(C, \mathbb{L})$	$\mathbb{C}$	1	$ C /n$	$[C = \arg \max_{C \in \mathbb{C}} \beta(C, \mathbb{L})]$

Table 5.4: The weights of some representative weighting functions under different measures m and clusterings $\mathbb{X}$.

where u is an upper bound of x . When it is applied to a measure, u cannot be an asymmetric function of x in order not to change its symmetry property. If a measure is already normalized, then set $u = 1$ to make sure it is still normalized. The symmetrization operation has only one free parameter: the selection of the upper bound. Since this operation is applied to distance measures, the upper bound of a distance measure cannot be trivial found, which instead is application specific.

5.2.4 Normalization Operation

A *normalization operation* is a scaling function $f_n : \mathbb{R}^3 \rightarrow [0, 1]$ such that $0 \leq f_n(x, l, u) \leq 1$ for any $x, l, u \in \mathbb{R}$ and $l \leq x \leq u$. A simple function $\mathbf{f}_n^s(\mathbf{x}, \mathbf{l}, \mathbf{u}) = (\mathbf{x} - \mathbf{l})/(\mathbf{u} - \mathbf{l})$ is often used, where l and u are a lower bound and an upper bound of x , respectively. It is an increasing function on x such that the similarization aspect is preserved. The symmetrization aspect can be changed according to the bounds used. In most situations, the lower bound can be fixed as zero even it is not reachable, and there are many ways to choose the upper bound.

We restrict the normalization operation to be applied to only similarity measures. One benefit of this restriction is that some upper bounds can be found trivially from a similarity measure itself. A cluster-cluster similarity measure α has two trivial upper bounds $\alpha(L, L)$ and $\alpha(C, C)$. The two trivial upper bounds of a clustering-clustering similarity measure γ are $\gamma(\mathbb{L}, \mathbb{L})$ and $\gamma(\mathbb{C}, \mathbb{C})$. For a cluster-clustering similarity measure

β , the only one trivial upper bound of $\beta(L, \mathbb{C})$ is $\beta(L, \{L\})$ while the only one trivial upper bound of $\beta(C, \mathbb{L})$ is $\beta(C, \{C\})$. Most upper bounds of a similarity measure can be expressed as a function of the measure itself and two trivial upper bounds. If a measure has only one trivial upper bound, the two trivial upper bounds are the same.

An *upper bound* function $f_u : \mathbb{R}^3 \rightarrow \mathbb{R}$ satisfies $x \leq f_u(x, u_1, u_2)$ for any $x, u_1, u_2 \in \mathbb{R}$ with $x \leq u_1$ and $x \leq u_2$. If $f_u(x, u_1, u_2) = f_u(x, u_2, u_1)$ for any $x, u_1, u_2 \in \mathbb{R}$, then it is symmetric and asymmetric otherwise. Some commonly used symmetric functions are the minimal $f_u^m(x, u_1, u_2) = \min\{u_1, u_2\}$, the harmonic mean $f_u^h(x, u_1, u_2) = 2u_1u_2/(u_1 + u_2)$, the geometric mean $f_u^g(x, u_1, u_2) = \sqrt{u_1u_2}$, the arithmetic mean $f_u^a(x, u_1, u_2) = (u_1 + u_2)/2$, the maximal $f_u^x(x, u_1, u_2) = \max\{u_1, u_2\}$, and the union¹ $f_u^u(x, u_1, u_2) = u_1 + u_2 - x$ of the two trivial upper bounds u_1 and u_2 . Two commonly used asymmetric functions are $f_u^l(x, u_1, u_2) = u_1$ and $f_u^r(x, u_1, u_2) = u_2$. They are summarized in Table 5.5. They obey the following inequality: $f_u^m \leq f_u^h \leq f_u^g \leq f_u^a \leq f_u^x \leq f_u^u$.

The upper bound function of a similarity measure can be considered as a measure that preserves the combination aspect and the similarization aspect. The upper bound of a cluster-cluster similarity measure α is a cluster-cluster measure $\alpha'(L, C) = f_u(\alpha(L, C), \alpha(L, L), \alpha(C, C))$. The upper bound of a clustering-clustering similarity measure γ is a clustering-clustering measure $\gamma'(\mathbb{L}, \mathbb{C}) = f_u(\gamma(\mathbb{L}, \mathbb{C}), \gamma(\mathbb{L}, \mathbb{L}), \gamma(\mathbb{C}, \mathbb{C}))$. The upper bound of a cluster-clustering similarity measure $\beta(L, \mathbb{C})$ is a cluster-clustering measure $\beta'(L, \mathbb{C}) = f_u(\beta(L, \mathbb{C}), \beta(L, \{L\}), \beta(L, \{L\}))$ and the upper bound of $\beta(C, \mathbb{L})$ is $\beta'(C, \mathbb{L}) = f_u(\beta(C, \mathbb{L}), \beta(C, \{C\}), \beta(C, \{C\}))$. In summary, given a similarity measure, its upper bound depends on the selection of the upper bound function f_u .

5.2.5 Symmetrization Operation

A *symmetrization operation* is a symmetric function $f_s : \mathbb{R}^2 \rightarrow \mathbb{R}$ such that $f_s(x, y) = f_s(y, x)$ for any $x, y \in \mathbb{R}$. Given an asymmetric cluster-cluster measure α , it cre-

¹We name it as a union function because of the set union $|L \cup C| = |L| + |C| - |L \cap C|$.

Function	Symmetric	Function	Symmetric
$f_u^m(x, u_1, u_2) = \min\{u_1, u_2\}$	✓	$f_u^h(x, u_1, u_2) = 2u_1u_2/(u_1 + u_2)$	✓
$f_u^g(x, u_1, u_2) = \sqrt{u_1u_2}$	✓	$f_u^a(x, u_1, u_2) = (u_1 + u_2)/2$	✓
$f_u^x(x, u_1, u_2) = \max\{u_1, u_2\}$	✓	$f_u^u(x, u_1, u_2) = u_1 + u_2 - x$	✓
$f_u^l(x, u_1, u_2) = u_1$	✗	$f_u^r(x, u_1, u_2) = u_2$	✗

Table 5.5: A few commonly used symmetric (✓) and asymmetric (✗) upper bound functions. They preserve the combination and the similarization aspects of a measure.

Function
$f_s^h(x, y) = 2xy/(x + y)$
$f_s^g(x, y) = \sqrt{xy}$
$f_s^a(x, y) = (x + y)/2$

Table 5.6: Some symmetrization functions.

ates a symmetric cluster-cluster measure $\alpha'(L, C) = f_s(\alpha(L, C), \alpha(C, L))$. Similarly, a symmetric clustering-clustering measure $\gamma'(\mathbb{L}, \mathbb{C}) = f_s(\gamma(\mathbb{L}, \mathbb{C}), \gamma(\mathbb{C}, \mathbb{L}))$ can be obtained from an asymmetric clustering-clustering measure γ . When the operator is applied to a cluster-clustering measure β , a symmetric cluster-cluster measure $\alpha'(L, C) = f_s(\beta(L, \mathbb{C}), \beta(C, \mathbb{L}))$. Therefore, the symmetrization operation has only one free parameter: the selection of the symmetric function f_s .

Some commonly used functions are the harmonic mean $f_s^h(x, y) = 2xy/(x + y)$, the geometric mean $f_s^g(x, y) = \sqrt{xy}$, and the arithmetic mean $f_s^a(x, y) = (x + y)/2$ of the two arguments x and y . They are listed in Table 5.6, and they satisfy the following inequality: $f_s^h(x, y) \leq f_s^g(x, y) \leq f_s^a(x, y)$ for any $x, y \in \mathbb{R}$.

5.3 Application

Most existing similarity measures can be derived using the proposed generative model, and we only need to determine the parameters associated with each operation at each step (see Table 5.7). The generative model provides a systematic way to generate similarity

Operation	Function	Parameters
Combination	$f_c^s(\mathbb{X}, m, w) = \sum_{X \in \mathbb{X}} w(X)m(X)$	$\mathbb{X}, w, m$
Similarization	$f_i^s(x, u) = u - x$	u
Normalization	$f_n^s(x, l, u) = (x - l)/(u - l)$	l, f_u
Symmetrization	$f_s(x, y)$	f_s

Table 5.7: The function and parameters of the four operations. Most parameters are actually functions. The lower bound l is usually set to zero.

Cluster-Cluster Measure	Formula	Normalized
Shared positive pairs	$\binom{ L \cap C }{2}$	✗
Exact Match	$[L = C]$	✓
Overlap Match	$[L \cap C \neq \emptyset]$	✓
Shared Data Points	$ L \cap C $	✗
Pointwise Mutual Information	$\log \frac{n L \cap C }{ L C }$	✗

Table 5.8: A few representative symmetric cluster-cluster similarity measures.

measure candidates, and each different application can choose the best candidate whose similarity scores are best aligned with the ground-truth (i.e. human judgement) scores.

Given a symmetric cluster-cluster similarity measure, using the four operations is capable of generating an existing normalized symmetric clustering-clustering similarity measure δ . Table 5.8 lists some representative symmetric cluster-cluster similarity measures, and Fig. 5.1 visualizes the possible operations that we can apply at each step. There are some restrictions: exactly two combination operations can be applied (one is combined on $\mathbb{L}$ while the other is combined on $\mathbb{C}$), and the symmetrization operation can be applied at most once on a cluster-clustering measure so as to avoid loops.

Most pair counting and cluster counting similarity measures follow this pattern: apply two successive combination operations with simple weighting, apply the similarization operation if the previous measure is a distance measure, and apply the normalization operation with different upper bound functions to create different specific measures. Most pair counting similarity measures are based on normalizaing $P(\mathbb{M}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \binom{|L \cap C|}{2}$

Figure 5.1: The possible ways to turn a symmetric cluster-cluster similarity measure (the root α of the tree at (a)) to a normalized symmetric clustering-clustering similarity measure δ . Combination, similarization, normalization, and symmetrization operations are called **comb**, **sim**, **norm**, and **sym** for short, respectively. The combination operation should be applied exactly twice, one is combined on $\mathbb{L}$ while the other is combined on $\mathbb{C}$. If α and β at the root of (a) and (b) are distance measures, we can only apply combination or similarization operations. The symmetrization on β can be applied at most once to avoid loops. The measure γ at the root of (c) must be a similarity measure.

with different bound functions (see the equalities in (4.2)). The exact matching similarity measures are based on normalizing $|\mathbb{L} \cap \mathbb{C}| = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} [L = C]$ with different upper bound functions. For the overlap matching similarity measures, a distance measure $|\mathbb{M}| = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} [L \cap C \neq \emptyset]$ is first obtained followed by turning it into a similarity measure $n - |\mathbb{M}|$, and the normalize it by utilizing the upper bounds (see (4.3)) of $|\mathbb{M}|$.

Most local set matching similarity measures are created as follows: apply a normalization operation on $|L \cap C|$, apply a combination operation with maximum weighting and a combination with simple weighting, apply a normalization operation to obtain normalized measures, and apply a similarization operation to obtain symmetric measures. Many global set matching measures are obtained in a way similar to the local matching measures except that the two combinations are replaced by a joint combination (5.1).

Most information theoretic similarity measures are obtained by first applying two combination operations with probabilistic weighing to obtain the mutual information

$I(\mathbb{M}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \log \frac{n|L \cap C|}{|L||C|}$ followed by applying a normalization operation with different upper bounds given in (4.7). The combination operation is more important than other operations. However, set matching measures are combined to ignore some clusters while other measures are simply combined to create unnormalized measures. The analysis of how existing measures are obtained inspires us to propose a new framework of similarity measures to simplify the design of a similarity measure as well as satisfying different desirable properties.

Chapter 6

A Graph-Based Model of Clusterings

We model the relation between two clusterings as a directed bipartite graph. One major reason of using a directed graph to model two clusterings is that a directed graph can be decomposed into several weakly connected components, which help us to study the intrinsic property of a measure in a divide-and-conquer manner. Using a directed bipartite graph instead of an undirected bipartite graph enables us to study both symmetric and asymmetric similarity measures. The graph is defined below.

Definition 6.0.1 (Graph). A directed bipartite graph $\mathcal{G} = (\mathbb{L}, \mathbb{C}, \mathbb{E})$ considers clusters as vertices and connects overlapping clusters with directed edges, where $\mathbb{L}$ is the set of true clusters, $\mathbb{C}$ is the set of predicted clusters, $\mathbb{E} = \{\langle L, C \rangle \mid L \in \mathbb{L}, C \in \mathbb{C}, L \cap C \neq \emptyset\}$ is the set of directed edges, and $\langle L, C \rangle$ is a directed edge from L to C with $L \cap C \neq \emptyset$.

6.1 Component

A directed bipartite graph can be decomposed into a set of weakly connected components that are called components for short.

Definition 6.1.1 (Component). A weakly connected component of a directed graph $\mathcal{G}$ is a maximal connected subgraph of $\mathcal{G}$ where vertices are mutually reachable by violating

the edge directions.

The decomposition is some kind of orthogonal since any two components are disjoint and the union of all components is the original graph. Decomposing a graph into components allows us to study measures using a divide-and-conquer approach. A directed graph is weakly connected if its corresponding undirected graph generated by ignoring edge directions is connected, that is, there is a path from every vertex to every other vertex. A weakly connected directed graph contains only one component.

When studying the relationship between a cluster and a clustering, we are often interested in how the objects of the cluster are separated in the clustering, the information of which is contained in an induced clustering.

Definition 6.1.2 (Induced Clustering). A clustering $\mathbb{C}$ gives an induced clustering $\mathbb{C}_A = \{C \cap A \mid C \in \mathbb{C}, C \cap A \neq \emptyset\}$ when acted upon by a cluster A [28]. $\mathbb{C}_A$ is a partition of cluster A and each cluster in $\mathbb{C}_A$ is a subset of a certain cluster of $\mathbb{C}$.

Based on two induced clusterings, an induced graph can be defined.

Definition 6.1.3 (Induced Graph). $\mathcal{G}_A = (\mathbb{L}_A, \mathbb{C}_A, \mathbb{E}_A)$ is an induced graph of graph $\mathcal{G}$ on a cluster A , where $\mathbb{L}_A$ and $\mathbb{C}_A$ are induced clusterings, and $\mathbb{E}_A = \{\langle L, C \rangle \mid L \in \mathbb{L}_A, C \in \mathbb{C}_A, L \cap C \neq \emptyset\}$ is the set of directed edges from $\mathbb{L}_A$ to $\mathbb{C}_A$.

Induced graph is not always the same with the induced subgraph defined in graph theory because the vertex set of $\mathcal{G}_A$ is not a subset of vertex set of $\mathcal{G}$. Instead, it is the set of nonempty intersections of vertices in $\mathcal{G}$ with cluster A . An induced graph $\mathcal{G}_A$ becomes an induced subgraph of $\mathcal{G}$ if it has the same edges that appear in $\mathcal{G}$ over the same vertex set. We study the relationship between components and clusters of the join.

Theorem 6.1.1 (Components and Join). *There is a bijection between the components of graph $\mathcal{G}$ and the clusters of the join $\mathbb{J}$: the union of all clusters within a component*

is a cluster of the join and the induced graph $\mathcal{G}_J$ of graph $\mathcal{G}$ on a cluster $J \in \mathbb{J}$ is a component.

Proof. First, we prove the 'one-to-one' part: every component corresponds to a cluster of the join. Every cluster in the join $\mathbb{J}$ is the smallest set that is exactly a union of clusters from both $\mathbb{L}$ and $\mathbb{C}$. The join can be computed based on graph $\mathcal{G}$ under two constraints: reachable vertices by violating the edge directions should be grouped together because the union of them should be a subset of one cluster in the join, and unreachable vertices by violating the edge directions should be separated to ensure each cluster in the join is the smallest. The constraints are the same with those imposed by the components of graph $\mathcal{G}$. As a result, the union of all clusters within a component becomes a cluster in the join.

We then prove the 'onto' part: every cluster of the join corresponds to a unique component. Assume two components correspond to the same cluster $J \in \mathbb{J}$. The set of the objects within the two components should be the same, which is a contradiction to the fact that components are disjoint. In summary, every component corresponds to a cluster of the join, and vice versa.

Given a cluster $J \in \mathbb{J}$, the corresponding component is the induced graph $\mathcal{G}_J$ that is computed using set intersection, the reverse operation of set union. Since components are disjoint, clusters of the join are also disjoint. The union of all components is $\mathcal{G}$, so the join $\mathbb{J}$ is a partition of data set D . $\square$

Within a graph $\mathcal{G}$, the set of components is $\{\mathcal{G}_J \mid J \in \mathbb{J}\}$ that is a partition of graph $\mathcal{G}$. Graph $\mathcal{G}$ is a union of its components $\bigcup_{J \in \mathbb{J}} \mathcal{G}_J$, and different components are disjoint $\mathcal{G}_J \cap \mathcal{G}_{J'} = \emptyset$ for $J \in \mathbb{J}$, $J' \in \mathbb{J}$ and $J \neq J'$. Figure 6.1 gives an example of a bipartite graph and its components.

Figure 6.1: The bipartite graph of two clusterings. A cluster is represented by a circle or an ellipse. Clusters on the same rectangle filled in gray belong to the same clustering. The top row is the true clustering while the bottom row is the predicted clustering. Clusters in the same hollow rectangle belong to the same component.

6.2 Decomposition on Components

Recall that S denotes a general similarity measure. The two clusterings within a component $\mathcal{G}_J$ are partitions of data set J . The similarity of a component $\mathcal{G}_J$ is defined as the similarity $S(\mathbb{L}_J, \mathbb{C}_J)$ of the two clusterings within the component. We define a property that is satisfied by most measures, which helps us to analyze the behavior of existing measures from the decomposition point of view.

Property 6.2.1 (Conical Decomposition on Components). *A similarity measure S is conical decomposable on components if it can be decomposed as*

$$S(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} w(J, \mathbb{L}, \mathbb{C}) S(\mathbb{L}_J, \mathbb{C}_J) + b(\mathbb{L}, \mathbb{C}),$$

where $w(J, \mathbb{L}, \mathbb{C}) \geq 0$ is a nonnegative weight assigned to component $\mathcal{G}_J$, and $b(\mathbb{L}, \mathbb{C}) \geq 0$ is a nonnegative bias term that is not related to any component score $S(\mathbb{L}_J, \mathbb{C}_J)$.

Proposition 6.2.1. *Assume two similarity measures S_1 and S_2 are conical decomposable on components and they have the same weighting function w and bias term b . The linear combination of S_1 and S_2 is also conical decomposable on components with weighting function w and bias term b .*

Assume a similarity measure S_1 is conical decomposable on components, that is, $S_1(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} w_1(J, \mathbb{L}, \mathbb{C}) S_1(\mathbb{L}_J, \mathbb{C}_J) + b_1(\mathbb{L}, \mathbb{C})$. Suppose we have another measure S_2

that is an upper bound of S_1 . A normalized measure $S_3(\mathbb{L}, \mathbb{C}) = S_1(\mathbb{L}, \mathbb{C})/S_2(\mathbb{L}, \mathbb{C})$ is also conical decomposable on components, and it can be decomposed as

$$S_3(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \left(w_1(J, \mathbb{L}, \mathbb{C}) \frac{S_2(\mathbb{L}_J, \mathbb{C}_J)}{S_2(\mathbb{L}, \mathbb{C})} \right) S_3(\mathbb{L}_J, \mathbb{C}_J) + \left(\frac{b_1(\mathbb{L}, \mathbb{C})}{S_2(\mathbb{L}, \mathbb{C})} \right).$$

We decompose most existing similarity measures in Appendix B and find they mainly differ in the weighting schemes. Since all components of a graph are disjoint and complete, we opine that a similarity score should simply be the weighted average of the similarity scores of all components. A refined property of Property 6.2.1 is described below.

Property 6.2.2 (Convex Decomposition on Components). *A similarity measure S is convex decomposable on components if it can be decomposed as*

$$S(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} w(J, \mathbb{L}, \mathbb{C}) S(\mathbb{L}_J, \mathbb{C}_J),$$

where $w(J, \mathbb{L}, \mathbb{C}) \geq 0$ is the positive weight assigned to component $\mathcal{G}_J$, and the summation of all component weights is one, that is, $\sum_{J \in \mathbb{J}} w(J, \mathbb{L}, \mathbb{C}) = 1$.

The weight of a component $\mathcal{G}_J$ is determined by the weights of all objects within J . In the absence of additional information such as the importance of each object, every object should be treated equally. A natural weighting scheme is to weight each component $\mathcal{G}_J$ by the relative size of the cluster J . Property 6.2.2 can be further refined.

Property 6.2.3 (Join-weighted Decomposition on Components). *A similarity measure S is join-weighted decomposable on components if it can be decomposed as*

$$S(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} S(\mathbb{L}_J, \mathbb{C}_J). \quad (6.1)$$

If $\mathbb{C}$ refines $\mathbb{L}$, the above property becomes the similarity version of the convex additivity axiom [28, Axiom A3]. Since the join is the least clustering that is refined by the two clusterings $\mathbb{L}$ and $\mathbb{C}$, $1 - S(\mathbb{L}, \mathbb{C})$ also satisfies the additivity of composition property [28, Property 8] if $S(\mathbb{L}, \mathbb{C})$ satisfies Property 6.2.3.

Chapter 7

The Split-Merge Framework

The decomposition of measures on components enable us to study and compare existing measures as well as designing new and better measures. A component is further decomposed into split and merge subcomponents in Section 7.1. The split and merge subcomponents can be combined into a derivation graph that transforms $\mathbb{L}$ into $\mathbb{C}$ via $\mathbb{M}$. In Section 7.2, we capture the essence of the derivation graph by pairing split subcomponents with merge subcomponents. The similarity of each pair, called a subcomponent pair, is discussed in Section 7.2.1. It is also there that the precise definition for the split-merge framework is given. We discuss its properties in Section 7.3 and compare it with some existing frameworks in Section 7.4. We give some examples of the split-merge framework in Section 7.5 and compare one selected example with other representative measures in Section 7.6.

7.1 Subcomponent

A component focuses on the clustering-clustering relation. It may be difficult to assign a score to such a relation. Hence, we further break a component into subcomponents, with the focus on the cluster-clustering relations. We define two kinds of subcomponents depending on whether cluster in the relation is from the true clustering or the predicted clustering.

Definition 7.1.1 (Split/Merge Subcomponent). An induced graph $\mathcal{G}_L$ of graph $\mathcal{G}$ on a truth cluster $L \in \mathbb{L}$ is a split subcomponent, and an induced graph $\mathcal{G}_C$ of graph $\mathcal{G}$ on a predicted cluster $C \in \mathbb{C}$ is a merge subcomponent.

A subcomponent is a subgraph of the graph $\mathcal{G}$. Even it contains two clusterings, one clustering contains only one cluster, and the other clustering is a partition of the cluster. For instance, a split subcomponent $\mathcal{G}_L$ contains a clustering $\mathbb{C}_L$ that is a partition of cluster L , and a merge subcomponent $\mathcal{G}_C$ contains a clustering $\mathbb{L}_C$ that is a partition of cluster C . Therefore, a subcomponent can be regarded as a clustering from a simplified point of view. Conceptually, a split subcomponent $\mathcal{G}_L$ splits a cluster L into one or more clusters of $\mathbb{C}$, while a merge subcomponent $\mathcal{G}_C$ merges one or more clusters of $\mathbb{L}$ into C . For a component $\mathcal{G}_J$, there can be one or more split/merge subcomponents.

Proposition 7.1.1 (Component and Subcomponents). *Within a component $\mathcal{G}_J$, $\{\mathcal{G}_L \mid L \in \mathbb{L}_J\}$ is the set of split subcomponents and $\{\mathcal{G}_C \mid C \in \mathbb{C}_J\}$ is the set of merge subcomponents.*

Proposition 7.1.2. *The set of sinks in the split subcomponent set of $\mathcal{G}_J$ is the same as the set of sources in the merge subcomponent set of $\mathcal{G}_J$. This set is the meet $\mathbb{M}_J$ of $\mathbb{L}_J$ and $\mathbb{C}_J$, and it is identical to the meet $\mathbb{M}$ of $\mathbb{L}$ and $\mathbb{C}$ induced by J .*

With the above proposition, we can transform $\mathbb{L}_J$ to $\mathbb{C}_J$ via $\mathbb{M}_J$. The transformation consists of splitting of clusters in $\mathbb{L}_J$ into clusters in $\mathbb{M}_J$ (if necessary), then merging into clusters in $\mathbb{C}_J$ (if necessary). The split and merge mappings are given by the split and merge subcomponent set of $\mathcal{G}_J$. Figure 7.1 gives an example. Formally, the transformation follows the derivation graph that combines the split and merge sets.

Definition 7.1.2 (Derivation Graph). The derivation graph is a directed tripartite graph with three parts: $\mathbb{L}$, $\mathbb{M}$ and $\mathbb{C}$. The set of subgraphs from $\mathbb{L}$ to $\mathbb{M}$ is the union of

Figure 7.1: Follow on from Figure 6.1. The top and bottom rows are $\mathbb{L}$ and $\mathbb{C}$, respectively. The middle two clusterings are the same: they are the meet $\mathbb{M}$. Each connected subgraph is a subcomponent. The top subgraphs are the split subcomponents while the bottom subgraphs are the merge subcomponents. Identifying the two copies of $\mathbb{M}$ together gives the derivation graph.

the split sets of the components of $\mathcal{G}$ (up to relabeling of the sinks in the union to the clusters in $\mathbb{M}$); and the set of subgraphs from $\mathbb{M}$ to $\mathbb{C}$ is the union of the merge sets (up to relabeling). There is no edge between vertices in $\mathbb{L}$ and vertices in $\mathbb{C}$.

7.1.1 Subcomponent Similarity Measure

A subcomponent can be simplified as a clustering based on a cluster, and it is more easily to score a clustering than scoring a component that contains two clusterings. A subcomponent similarity measure is actually a clustering compactness measure. It should correctly rank and assign scores to all the clusterings based on their compactness. We opine that a proper measure should consider two factors: the number of clusters and the relative size of each cluster [48]. A clustering is more compact if it contains very few clusters or has skewed distribution of cluster sizes. The following property checks whether a subcomponent similarity measure takes these factors into account.

Property 7.1.1 (Subcomponent Monotonically Decreasing). *A subcomponent similarity measure is monotonically decreasing if the similarity monotonically decreases as the number of clusters increases or the distribution of cluster sizes gets less skewed.*

Besides ranking all subcomponents based on the same cluster correctly, the following property states that a proper subcomponent similarity measure should also produce normalized similarities to facilitate fair comparison between subcomponents based on different clusters.

Property 7.1.2 (Subcomponent Normalization). *A subcomponent similarity measure or a clustering compactness measure is normalized if*

1. *the similarity is one if and only if the clustering contains only one cluster;*
2. *the similarity is zero if and only if the clustering contains only singletons; and*
3. *the similarity is within the open interval $(0, 1)$ otherwise.*

Note that if there is only one object within the clustering, its score should be one according to the first rule.

A subcomponent is a connected bipartite graph by definition. The similarity measure S can also be applied to compare the two clusterings within a subcomponent. We denote the similarity of a split subcomponent $\mathcal{G}_L$ as $s(\mathbb{C}_L)$ and the similarity of a merge subcomponent $\mathcal{G}_C$ as $m(\mathbb{L}_C)$. For consistency, the following property requires that $S(\{L\}, \mathbb{C}_L)$ should equal to $s(\mathbb{C}_L)$ and $S(\mathbb{L}_C, \{C\})$ should be the same with $m(\mathbb{L}_C)$.

Property 7.1.3 (Subcomponent Consistency). *A similarity measure S is subcomponent consistent if $S(\{L\}, \mathbb{C}_L) = s(\mathbb{C}_L)$ and $S(\mathbb{L}_C, \{C\}) = m(\mathbb{L}_C)$.*

7.2 Subcomponent Pair

A subcomponent is easy to be scored, but different subcomponents may be related. This prevents us from directly combining the similarities of subcomponents into a component similarity. Different subcomponents of the same type (i.e., split or merge) are disjoint because their associated clusters are disjoint, and different types of subcomponents are possibly related if their associated clusters overlap. We propose a subcomponent pair to capture the relatedness between a split subcomponent and a merge subcomponent.

Definition 7.2.1 (Subcomponent Pair). A subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ contains a split subcomponent $\mathcal{G}_L$ and a merge subcomponent $\mathcal{G}_C$, where $L \in \mathbb{L}$, $C \in \mathbb{C}$ and $L \cap C \neq \emptyset$.

This definition exploits that a split and a merge subcomponent are not disjoint in the derivation graph only if a sink in the split subcomponent and a source in the merge subcomponent is the same cluster in $\mathbb{M}_J$; see Proposition 7.1.2.

Theorem 7.2.1 (Subcomponent Pairs and Meet). *There is a one-to-one correspondence between the subcomponent pairs of graph $\mathcal{G}$ and the clusters of meet $\mathbb{M}$: a subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ corresponds to a cluster $L \cap C$ of the meet and a cluster $M \in \mathbb{M}$ corresponds to a subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ with $M = L \cap C$.*

Proof. The existence of the cluster in the meet is by definition of a subcomponent pair. A subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ requires $L \cap C \neq \emptyset$, and every nonempty $L \cap C$ is a member of $\mathbb{M}$ according to the definition of meet. For uniqueness, we need to prove that a cluster $M \in \mathbb{M}$ is mapped to a unique subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ such that $L \cap C = M$, which can be proved if we can show that a cluster $M \in \mathbb{M}$ is mapped to a unique true cluster $L \in \mathbb{L}$ and a unique predicted cluster $C \in \mathbb{C}$ such that $M \subseteq L$ and $M \subseteq C$.

We prove that a cluster of the meet is mapped to a unique true cluster and the other case can be proved in a similar manner. The definition of meet implies that there exists at least one true cluster L such that $M \subseteq L$. For uniqueness, assume there is another true cluster $L' \in \mathbb{L}$ such that $M \subseteq L'$, then L and L' must overlap, which is a contradiction to the fact that different clusters are disjoint $L \cap L' = \emptyset$. $\square$

The above conclusion indicates that subcomponent pairs are similar to components in that they are disjoint and complete. Since the true and predicted clusterings are refined by their meet, we introduce a split-merge interpretation of transforming the true clustering to the predicted clustering through their meet: the true clustering is transformed to

the meet by splitting no or some clusters followed by merging no or some clusters of the meet to obtain the predicted clustering.

Proposition 7.2.2 (Component and Subcomponent Pairs). *Within a component $\mathcal{G}_J$, $\{\langle \mathcal{G}_L, \mathcal{G}_C \rangle \mid L \cap C \in \mathbb{M}_J\}$ is the set of subcomponent pairs.*

With this proposition, we can exactly enumerate all the subcomponent pairs in a component $\mathcal{G}_J$ by enumerating the clusters in $\mathbb{M}_J$. Moreover, $\mathbb{M}_J$ is a partition on J . These two properties suggest the following decomposition of the component score $S(\mathbb{L}_J, \mathbb{C}_J)$ in the spirit of Property 6.2.3.

$$S(\mathbb{L}_J, \mathbb{C}_J) = \sum_{M \in \mathbb{M}_J} \frac{|M|}{|J|} \sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} [L \cap C = M] \sigma(\mathcal{G}_L, \mathcal{G}_C),$$

where $[L \cap C = M]$ is one if $L \cap C$ is M and zero otherwise, and $\sigma(\mathcal{G}_L, \mathcal{G}_C)$ is the similarity of a subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$. Using Theorem 7.2.1 and $|\emptyset| = 0$, the above decomposition can be simplified as

$$S(\mathbb{L}_J, \mathbb{C}_J) = \sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} \frac{|L \cap C|}{|J|} \sigma(\mathcal{G}_L, \mathcal{G}_C).$$

This can be directly substituted into (6.1). After subsuming the sum over the join into the sums over the true and predicted clusterings, we obtain the following convex combination.

Property 7.2.1 (Meet-weighted Decomposition on Subcomponent Pairs). *A similarity measure S is meet-weighted decomposable on subcomponent pairs if it can be decomposed as*

$$S(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \sigma(\mathcal{G}_L, \mathcal{G}_C). \quad (7.1)$$

The join contains the information about the components of two clusterings, while the meet contains the information about the subcomponent pairs of two clusterings. We have the following proposition regarding their sizes.

Proposition 7.2.3 (Components, Subcomponents, and Subcomponent Pairs).

A graph $\mathcal{G}$ has $|\mathbb{J}|$ components, $|\mathbb{L}|$ split subcomponents, $|\mathbb{C}|$ merge subcomponents, and $|\mathbb{M}|$ subcomponent pairs.

7.2.1 Subcomponent Pair Similarity Measure

We now discuss the choice of the similarity measure σ on a subcomponent pair. Given a subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$, the general idea is to let σ be a function of the split subcomponent measure s and the merge subcomponent measure m . We discuss two functions: the product and the arithmetic mean. Our preference is the product because it is subcomponent consistent (Property 7.1.3), while the arithmetic mean is not.

Product The proposed split-merge framework of similarity measures is defined as

$$S^*(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}_L) m(\mathbb{L}_C). \quad (7.2)$$

The similarity measures instantiated from the above split-merge framework can be symmetric/asymmetric and normalized/unnormalized depending on the subcomponent measures used. As we shall show in Section 7.3, $S^*(\mathbb{L}, \mathbb{C})$ is subcomponent consistent when the subcomponent similarity measures are normalized.

Arithmetic Mean One might also use other functions of $s(\mathbb{C}_L)$ and $m(\mathbb{L}_C)$ to define σ . A natural choice is the arithmetic mean, which combined with (7.1) gives

$$S'(\mathbb{L}, \mathbb{C}) = \frac{1}{2} \sum_{L \in \mathbb{L}} \frac{|L|}{n} s(\mathbb{C}_L) + \frac{1}{2} \sum_{C \in \mathbb{C}} \frac{|C|}{n} m(\mathbb{L}_C).$$

The measures instantiated from the above framework can be normalized/unnormalized symmetric/asymmetric similarity/distance measures depending on the subcomponent measures used. If the subcomponent measures s and m are distance measures, S' is also a distance measure. Some existing measures are instances of S' . If we define both s and m

to be the fraction of the largest cluster within a clustering, that is, $s(\mathbb{C}_L) = \max_{C \in \mathbb{C}} \frac{|L \cap C|}{|L|}$ and $m(\mathbb{L}_C) = \max_{L \in \mathbb{L}} \frac{|L \cap C|}{|C|}$, then $S'(\mathbb{L}, \mathbb{C})$ becomes the similarity counterpart of the constrained Van Dongen metric. If both s and m measure the entropy of a clustering, that is, $s(\mathbb{C}_L) = H(\mathbb{C}_L)$ and $m(\mathbb{L}_C) = H(\mathbb{L}_C)$, then $2S'(\mathbb{L}, \mathbb{C})$ becomes the variation of information. Similarly, the two normalized variants (by $\log n$ and $\log k^2$) of the variation of information are also instances of $S'(\mathbb{L}, \mathbb{C})$.

The similarity $S'(\mathbb{L}, \mathbb{C})$ directly combines the subcomponent similarities and ignores the dependencies between them. Hence, it defeats the very purpose of subcomponent pairs. In contrast, the similarity given by (7.2) cannot be further (linearly) decomposed into subcomponent similarities. Moreover, $S'(\mathbb{L}, \mathbb{C})$ is not subcomponent consistent: $S'(\{L\}, \mathbb{C}_L) = s(\mathbb{C}_L)/2 + 1/2 \geq s(\mathbb{C}_L)$, with equality only when $s(\mathbb{C}_L) = 1$.

7.3 Properties of the Framework

We prove some properties of the framework. Most of the properties are strongly related to the properties of its two subcomponent measures.

7.3.1 Time Complexity

Property 7.3.1. *The similarity S^* can be computed with time complexity $O(n)$ under the assumption that the time complexities of computing the similarities of a split subcomponent $\mathcal{G}_L$ and a merge subcomponent $\mathcal{G}_C$ are bounded by $O(|L|)$ and $O(|C|)$ respectively.*

Proof. Firstly, it takes $O(n)$ to find all the split subcomponents and $\sum_{L \in \mathbb{L}} O(|L|) = O(n)$ to compute their similarities. Secondly, it takes $O(n)$ to find all the merge subcomponents and $\sum_{C \in \mathbb{C}} O(|C|) = O(n)$ to compute their similarities. Thirdly, it takes $O(n)$ to find the $|\mathbb{M}|$ subcomponent pairs since the meet can be computed in $O(n)$ when an appropriate data structure is used [35], and we need $O(n)$ to compute the weighted average of their similarities as $S^*(\mathbb{L}, \mathbb{C})$. As a result, the overall time complexity is $O(n)$. $\square$

7.3.2 Symmetry

Property 7.3.2 (Symmetry). *The similarity measure S^* is symmetric if the subcomponent measures s and m are the same, and asymmetric otherwise.*

Proof. Exchanging the roles of the true clustering with the predicted clustering, the similarity $S^*(\mathbb{C}, \mathbb{L})$ is $\sum_{C \in \mathbb{C}} \sum_{L \in \mathbb{L}} \frac{|L \cap C|}{n} s(\mathbb{L}_C) m(\mathbb{C}_L)$. Letting $S^*(\mathbb{L}, \mathbb{C}) = S^*(\mathbb{C}, \mathbb{L})$ requires that $s(\mathbb{C}_L) m(\mathbb{L}_C) = m(\mathbb{C}_L) s(\mathbb{L}_C)$ for every $L \in \mathbb{L}$ and every $C \in \mathbb{C}$, which further demands the two subcomponent measures to be the same. $\square$

7.3.3 The Best Clustering and Worst Clusterings

Property 7.3.3 (The Best Clustering). *When the two subcomponent similarity measures s and m are normalized, $S^*(\mathbb{L}, \mathbb{C}) = 1$ if and only if $\mathbb{L} = \mathbb{C}$.*

Proof. It is easy to verify that the similarity between the true and predicted clusterings is one when they are the same. When the similarity is one, the similarity $s(\mathbb{C}_L) m(\mathbb{L}_C)$ of every subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ is one. It implies the similarity of every subcomponent is one. That $s(\mathbb{C}_L) = 1$ for every split subcomponent $\mathcal{G}_L$ suggests that the true clustering refines the predicted clustering, and the similarity $m(\mathbb{L}_C)$ of every merge subcomponent $\mathcal{G}_C$ being one suggests that the predicted clustering refines the true clustering. Therefore, the true clustering and the predicted clustering are the same. $\square$

Property 7.3.4 (The Worst Clusterings). *When the two subcomponent similarity measures s and m are normalized, $S^*(\mathbb{L}, \mathbb{C}) = 0$ if and only if $\mathbb{M} = \perp$ and $\mathbb{L} \cap \mathbb{C} = \emptyset$.*

Proof. When the similarity is zero, the similarity $s(\mathbb{C}_L) m(\mathbb{L}_C)$ of every subcomponent pair $\langle \mathcal{G}_L, \mathcal{G}_C \rangle$ is zero. This means each true cluster is split into singletons and/or each predicted cluster is merged from singletons. For a split subcomponent $\mathcal{G}_L$, its similarity is always one if L is a singleton since it cannot be further split; otherwise, a zero subcomponent similarity can be achieved by splitting L into singletons. For a merge subcomponent

$\mathcal{G}_C$, its similarity is zero if it contains at least two objects with no two objects belonging to the same true cluster; otherwise, its similarity is positive. Given a true clustering, a worst predicted clustering can be generated using two steps. First, each non-singleton true cluster is split into singletons, so the meet $\mathbb{M}$ equals to the bot $\perp$. Second, the singletons of the meet can be merged or untouched in any way while ensuring that the singletons from the true clustering are merged in some way, so the intersection $\mathbb{L} \cap \mathbb{C}$ between the true clustering and the predicted clustering is empty. $\square$

Property 7.3.5. *Assume the two subcomponent similarity measures s and m are normalized. Given any true clustering $\mathbb{L}$, the size of a worst predicted clustering $\mathbb{C}$ is lower bounded by $\max_{L \in \mathbb{L}} |L|$ and upper bounded by $n - \lceil |\mathbb{L} \cap \perp|/2 \rceil$.*

Proof. We prove the property by proposing methods to generate the worst clusterings whose sizes are the smallest and largest without breaking the two constraints: $\mathbb{M} = \perp$ and $\mathbb{L} \cap \mathbb{C} = \emptyset$. We use a simple method to construct the worst clustering with the smallest number of clusters: put the i -th object of the j -th true cluster into the i -th predicted cluster, resulting in $\max_{L \in \mathbb{L}} |L|$ true clusters generated. The construction of the two maximally separated clusterings mentioned in [28] is a special case of our method. If the number of predicted clusters is less than $\max_{L \in \mathbb{L}} |L|$, then there always exists a predicted cluster that contains at least two objects from the largest true cluster, which absolutely violates the constraint $\mathbb{M} = \perp$. As a result, the size of a worst predicted clustering is lower bounded by $\max_{L \in \mathbb{L}} |L|$.

We use three steps to construct the worst clustering with the largest number of clusters. Firstly, every object of the non-singleton true clusters forms a predicted cluster, resulting in $n - |\mathbb{L} \wedge \perp|$ predicted clusters. Secondly, we merge any two singletons of the true clustering to form a new predicted cluster, resulting in $\lfloor |\mathbb{L} \cap \perp|/2 \rfloor$ additional predicted clusters. Thirdly, if there is one more singleton in the true clustering, it is

merged with any predicted cluster. Therefore, the size of a worst predicted clustering is upper bounded by $n - |\mathbb{L} \wedge \perp| + \lfloor |\mathbb{L} \cap \perp|/2 \rfloor = n - \lceil |\mathbb{L} \cap \perp|/2 \rceil$. $\square$

7.3.4 Normalization

Property 7.3.6 (Normalization). *The similarity measure S^* is normalized if 1) the subcomponent measures s and m are lower bounded by zero and 2) the multiplication of their upper bounds equals to one, and unnormalized otherwise.*

Proof. If subcomponent measures s and m are lower bounded by zero, S^* is also lower bounded by zero because $S^*(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}_L) m(\mathbb{L}_C) \geq 0$. Let the upper bounds of s and m be u_s and u_m respectively, S^* is upper bounded by $u_s u_m$ because

$$S^*(\mathbb{L}, \mathbb{C}) \leq \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} u_s u_m = u_s u_m \left(\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \right) = u_s u_m.$$

Making S^* normalized requires that $u_s u_m$ equals to one. $\square$

A special case of the above property is that: the similarity measure S^* is normalized if the subcomponent measures s and m are normalized (see Property 7.1.2). Most of the remaining properties require the two subcomponent measures to be normalized.

Property 7.3.7 (Conditional Normalization). *When the two subcomponent similarity measures s and m are normalized, the similarity measure S^* is conditionally normalized.*

Proof. According to Property 7.3.3, there is only one best predicted clustering for any true clustering. The worst clustering condition ($\mathbb{M} = \perp$ and $\mathbb{L} \cap \mathbb{C} = \emptyset$) given by Property 7.3.4 implies $\Omega_{\mathbb{L}}^w \neq \emptyset$ when $n > 1$. The union of the best and worst clusterings is: ($\mathbb{M} = \perp$ and $\mathbb{L} \cap \mathbb{C} = \emptyset$) or $\mathbb{L} = \mathbb{C}$. It is a proper subset of Ω when $n > 2$, making $\Omega_{\mathbb{L}}^n$ being nonempty. Therefore, S^* is conditionally normalized. $\square$

7.3.5 Top and Bottom

Property 7.3.8 (Top). *When the two subcomponent similarity measures s and m are normalized, $S^*(\top, \mathbb{C}) = s(\mathbb{C})$ and $S^*(\mathbb{L}, \top) = m(\mathbb{L})$.*

Proof. When the true clustering is the top, the similarity of every merge subcomponent is one. The similarity between the top and a predicted clustering is the compactness $s(\mathbb{C})$ of the predicted clustering because

$$S^*(\top, \mathbb{C}) = \sum_{L \in \top} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}) = s(\mathbb{C}) \left(\sum_{L \in \top} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \right) = s(\mathbb{C}).$$

Similarly, we can prove that the similarity between the true clustering and the top is the compactness $m(\mathbb{L})$ of the true clustering. $\square$

Property 7.3.9 (Bottom). *When the two subcomponent similarity measures s and m are normalized, $S^*(\perp, \mathbb{C}) = |\perp \cap \mathbb{C}|/n$ and $S^*(\mathbb{L}, \perp) = |\mathbb{L} \cap \perp|/n$.*

Proof. When the true clustering is the bottom, the similarity of every split subcomponent $\mathcal{G}_L$ is one, and the similarity of a merge subcomponent $\mathcal{G}_C$ is $[|C| = 1]$ that takes value one when C is a singleton and zero otherwise. The similarity between the bottom and a predicted clustering is the fraction of the singletons in the predicted clustering because

$$\begin{aligned} S^*(\perp, \mathbb{C}) &= \sum_{L \in \perp} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} [|C| = 1] = \sum_{C \in \mathbb{C}} [|C| = 1] \left(\sum_{L \in \perp} \frac{|L \cap C|}{n} \right) \\ &= \sum_{C \in \mathbb{C}} [|C| = 1] \frac{|C|}{n} = \sum_{C \in \mathbb{C}} \frac{[|C| = 1]}{n} = \frac{|\perp \cap \mathbb{C}|}{n}. \end{aligned}$$

Similarly, we can prove that the similarity between the true clustering and the bottom is the fraction of the singletons in the true clustering, $\square$

7.3.6 Split and Merge

Property 7.3.10 (Split). *Assume the two subcomponent similarity measures s and m are normalized. If the predicted clustering is obtained by splitting only one cluster L' of*

the true clustering into a few clusters while keeping the rest true clusters untouched, then

$$S^*(\mathbb{L}, \mathbb{C}) = 1 - \frac{|L'|}{n}(1 - s(\mathbb{C}_{L'})).$$

Proof. It is obvious that the predicted clustering refines the true clustering. The similarity $s(\mathbb{C}_L)$ of a split subcomponent $\mathcal{G}_L$ is one if $L \neq L'$, and $s(\mathbb{C}_{L'})$ otherwise. The similarity of every merge subcomponent is one. The similarity $S^*(\mathbb{L}, \mathbb{C})$ is given by

$$S^*(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \frac{|L|}{n} s(\mathbb{C}_L) = \frac{|L'|}{n} s(\mathbb{C}_{L'}) + \sum_{L \in \mathbb{L}, L \neq L'} 1 = \frac{|L'|}{n} s(\mathbb{C}_{L'}) + 1 - \frac{|L'|}{n}.$$

It can be rewritten as $1 - \frac{|L'|}{n}(1 - s(\mathbb{C}_{L'}))$. $\square$

The distance $1 - S^*(\mathbb{L}, \mathbb{C})$ achieved by splitting a true cluster L' is proportional to its relative size $\frac{|L'|}{n}$ times the penalty $1 - s(\mathbb{C}_{L'})$ of the split.

Property 7.3.11 (Merge). *Assume the two subcomponent similarity measures s and m are normalized. If the predicted clustering is obtained by merging some clusters of the true clustering into a predicted cluster C' while keeping the rest true clusters untouched, then $S^*(\mathbb{L}, \mathbb{C}) = 1 - \frac{|C'|}{n}(1 - m(\mathbb{L}_{C'}))$.*

Proof. We prove it using a approach different from the above property. The join is the predicted clustering since it is refined by the true clustering. The two clusterings differ only in component $\mathcal{G}_{C'}$ whose similarity is $\sum_{L \in \mathbb{L}} \frac{|L|}{n} m(\mathbb{L}_{C'}) = m(\mathbb{L}_{C'})$. The distance $1 - S^*(\mathbb{L}, \mathbb{C})$ between them is the weighed distance $\frac{|C'|}{n}(1 - m(\mathbb{L}_{C'}))$ of component $\mathcal{G}_{C'}$. $\square$

The above two properties show that the distance caused by splitting a cluster or merging a few clusters into one cluster is independent of other clusters, which is a desirable property [28].

7.3.7 Cluster Homogeneity and Cluster Completeness

Property 7.3.12 (Cluster Homogeneity). *The similarity measure $S^*(\mathbb{L}, \mathbb{C})$ meets the cluster homogeneity criterion if the merge subcomponent measure m meets Property 7.1.1.*

Proof. The similarity of a merge subcomponent $\mathcal{G}_C$ is defined as the compactness of the induced clustering $\mathbb{L}_C$, which can also be interpreted as the homogeneity of the predicted cluster C . The more compact the induced clustering $\mathbb{L}_C$ is, the more homogeneous the predicted cluster C is. A clustering is more homogeneous if it merges less true clusters into a predicted cluster according to [2]. Based on the true clustering, assume a predicted clustering $\mathbb{C}_1$ is formed by only merging two true clusters L_1 and L_2 into a predicted cluster C_1 , and another predicted clustering $\mathbb{C}_2$ is formed by only merging three true clusters L_1 , L_2 and L_3 into a predicted cluster C_2 . The similarity between $\mathbb{L}$ and $\mathbb{C}_1$ is $S^*(\mathbb{L}, \mathbb{C}_1) = 1 - \frac{|\mathbb{C}_1|}{n}(1 - m(\mathbb{L}_{C_1}))$, where $\mathbb{L}_{C_1} = \{L_1, L_2\}$. The similarity between $\mathbb{L}$ and $\mathbb{C}_2$ is $S^*(\mathbb{L}, \mathbb{C}_2) = 1 - \frac{|\mathbb{C}_2|}{n}(1 - m(\mathbb{L}_{C_2}))$, where $\mathbb{L}_{C_2} = \{L_1, L_2, L_3\}$. If the merge subcomponent measure m satisfies Property 7.1.1, then $m(\mathbb{L}_{C_1}) > m(\mathbb{L}_{C_2})$. Considering that $\frac{|\mathbb{C}_1|}{n} < \frac{|\mathbb{C}_2|}{n}$, we can conclude $\mathbb{C}_1$ is more homogeneous than $\mathbb{C}_2$ because $S^*(\mathbb{L}, \mathbb{C}_1) > S^*(\mathbb{L}, \mathbb{C}_2)$. $\square$

Property 7.3.13 (Cluster Completeness). *The similarity measure $S^*(\mathbb{L}, \mathbb{C})$ satisfies the cluster completeness criterion if the split subcomponent measure s satisfies Property 7.1.1.*

Proof. It can be proved in a similar manner. $\square$

7.3.8 Decomposition and Consistency

Property 7.3.14. *The similarity measure S^* is join-weighted decomposable on components: $S^*(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} S^*(\mathbb{L}_J, \mathbb{C}_J)$.*

Proof. A true cluster $L \in J$ only overlaps with predicted clusters within $\mathbb{C}_J$, so

$$(\mathbb{C}_J)_L = \{C \cap L \mid C \in \mathbb{C}_J, C \cap L \neq \emptyset\} = \{C \cap L \mid C \in \mathbb{C}, C \cap L \neq \emptyset\} = \mathbb{C}_L.$$

Similarly, $(\mathbb{L}_J)_C = \mathbb{L}_C$ for $C \in J$. Based on the above conclusions, we can prove

$$\begin{aligned} S^*(\mathbb{L}, \mathbb{C}) &= \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}_L) m(\mathbb{L}_C) = \sum_{J \in \mathbb{J}} \left(\sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} \frac{|L \cap C|}{n} s(\mathbb{C}_L) m(\mathbb{L}_C) \right) \\ &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} \frac{|L \cap C|}{|J|} s((\mathbb{C}_J)_L) m((\mathbb{L}_J)_C) \right) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} S^*(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

□

Property 7.3.15. *When the two subcomponent similarity measures s and m are normalized, the similarity measure S^* is subcomponent consistent: $S^*(\{L\}, \mathbb{C}_L) = s(\mathbb{C}_L)$ and $S^*(\mathbb{L}_C, \{C\}) = m(\mathbb{L}_C)$.*

Proof. When the similarity measure S^* is applied to a subcomponent $\mathcal{G}_L$,

$$S^*(\{L\}, \mathbb{C}_L) = \sum_{L \in \{L\}} \sum_{C \in \mathbb{C}_L} \frac{|C|}{|L|} s(\mathbb{C}_L) \underbrace{m(\{C\})}_{=1} = s(\mathbb{C}_L) \left(\sum_{L \in \{L\}} \sum_{C \in \mathbb{C}_L} \frac{|C|}{|L|} \right) = s(\mathbb{C}_L).$$

$S^*(\mathbb{L}_C, \{C\}) = m(\mathbb{L}_C)$ can be proved similarly.

□

7.3.9 Bounds

Property 7.3.16 (Meet). *Assume the two subcomponent similarity measures s and m are normalized. $S^*(\mathbb{L}, \mathbb{C}) \leq S^*(\mathbb{L}, \mathbb{M})$ with equality holds if and only if $\mathbb{M} = \mathbb{C}$, and $S^*(\mathbb{L}, \mathbb{C}) \leq S^*(\mathbb{M}, \mathbb{C})$ with equality holds if and only if $\mathbb{M} = \mathbb{L}$.*

Proof. We first prove the two induced clusterings $\mathbb{M}_L$ and $\mathbb{C}_L$ are the same.

$$\begin{aligned} \mathbb{M}_L &= \{M \cap L \mid M \in \mathbb{M}, M \cap L \neq \emptyset\} \\ &= \{(L' \cap C) \cap L \mid L' \in \mathbb{L}, C \in \mathbb{C}, (L' \cap C) \cap L \neq \emptyset\} \\ &= \{C \cap (L' \cap L) \mid C \in \mathbb{C}, L' \in \mathbb{L}, C \cap (L' \cap L) \neq \emptyset\} \\ &= \{C \cap L \mid C \in \mathbb{C}, C \cap L \neq \emptyset\} = \mathbb{C}_L. \end{aligned}$$

The meet refines the true clustering, so the similarity $m(\mathbb{L}_M)$ of a merge subcomponent $\mathbb{L}_M$ is one. The similarity $S^*(\mathbb{L}, \mathbb{C})$ is upper bounded by $S^*(\mathbb{L}, \mathbb{M})$.

$$\begin{aligned} S^*(\mathbb{L}, \mathbb{M}) &= \sum_{L \in \mathbb{L}} \sum_{M \in \mathbb{M}} \frac{|L \cap M|}{n} s(\mathbb{M}_L) m(\mathbb{L}_M) = \sum_{L \in \mathbb{L}} \sum_{L' \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap (L' \cap C)|}{n} s(\mathbb{C}_L) \\ &= \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}_L) \geq \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s(\mathbb{C}_L) m(\mathbb{L}_C) = S^*(\mathbb{L}, \mathbb{C}). \end{aligned}$$

The equality holds only when the similarity $m(\mathbb{L}_C)$ of each merge subcomponent $\mathcal{G}_C$ is one, that is, the predicted clustering refines the true clustering, which is the same with $\mathbb{M} = \mathbb{C}$. Similarly, we can prove that the similarity $S^*(\mathbb{L}, \mathbb{C})$ is upper bounded by $S^*(\mathbb{M}, \mathbb{C})$, and they are the same only when the true clustering refines the predicted clustering, in notation, $\mathbb{M} = \mathbb{L}$. $\square$

The above property simply means the true clustering and the predicted clustering are more similar to their meet than to each other. From the split-merge point of view, the true clustering is split into the meet followed by merging the meet to the predicted clustering. The split and merge operations decrease the similarity. Therefore, the two clusterings are more similar to their intermediate clustering $\mathbb{M}$.

Property 7.3.17 (Bound Preserving). *A similarity measure S_1^* is based split subcomponent measure s_1 and merge subcomponent measure m_1 , and another similarity measure S_2^* is based split subcomponent measure s_2 and merge subcomponent measure m_2 . If $s_1 \leq s_2$ and $m_1 \leq m_2$, then $S_1^* \leq S_2^*$.*

Proof. $S_1^*(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s_1(\mathbb{C}_L) m_1(\mathbb{L}_C) \leq \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} s_2(\mathbb{C}_L) m_2(\mathbb{L}_C) = S_2^*(\mathbb{L}, \mathbb{C})$. $\square$

7.4 Comparisons with Other Frameworks

7.4.1 The Arithmetic Mean Framework

$S^*(\mathbb{L}, \mathbb{M})$ and $S^*(\mathbb{M}, \mathbb{C})$ can also be regarded as two trivial asymmetric measures generated from the split-merge framework by letting the split and the merge subcomponent

similarity measures to be one, respectively.

$$S^*(\mathbb{L}, \mathbb{M}) = \sum_{L \in \mathbb{L}} \frac{|L|}{n} s(\mathbb{C}_L) \times 1, \quad S^*(\mathbb{M}, \mathbb{C}) = \sum_{C \in \mathbb{C}} \frac{|C|}{n} 1 \times m(\mathbb{L}_C).$$

Assume the two subcomponent similarity measures s and m are normalized. Property 7.3.16 states that $S^*(\mathbb{L}, \mathbb{M})$ and $S^*(\mathbb{M}, \mathbb{C})$ are two trivial bounds of $S^*(\mathbb{L}, \mathbb{C})$. The definitions of the two asymmetric measures also imply that their scores are higher than $S^*(\mathbb{L}, \mathbb{C})$.

The arithmetic mean framework $S'(\mathbb{L}, \mathbb{C}) = (S^*(\mathbb{L}, \mathbb{M}) + S^*(\mathbb{M}, \mathbb{C}))/2$ can be considered as a symmetric measure generated from the two asymmetric measures. Consequently, the arithmetic mean framework is derived from the split-merge framework, and it is another upper bound of the split-merge framework. Similarly, it also gives higher scores than the split-merge framework.

The framework of the generalized information-theoretic distance proposed in [24] is $S^*(\mathbb{L}, \mathbb{M}) + S^*(\mathbb{M}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \frac{|L|}{n} s(\mathbb{C}_L) + \sum_{C \in \mathbb{C}} \frac{|C|}{n} m(\mathbb{L}_C)$, which restricts both s and m to be distance measures. It is equivalent to the arithmetic mean framework. Unnormalized distance measures s and m are used in their article, resulting in unnormalized distance measures. To make them normalized, they proposed complicated computational solutions.

7.4.2 The Direct-sum Framework

We introduce a direct-sum framework

$$S^+(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} s(\mathbb{C}_L) + \sum_{C \in \mathbb{C}} m(\mathbb{L}_C) = s(\mathbb{M}) + m(\mathbb{M}).$$

Compared to the arithmetic mean framework, it uses a simple weighting function, resulting in a measure with simple formula. This framework only models unnormalized symmetric measures and offers no solutions about how to perform the normalization.

Even the split and merge subcomponents are normalized, the $S^+(\mathbb{L}, \mathbb{C})$ is still unnormalized. If both s and m are similarity (resp. distance) measures, $s(\mathbb{M})$, $m(\mathbb{M})$, and $S^+(\mathbb{L}, \mathbb{C})$ are upper (resp. lower) bounded by $s(\mathbb{L})$, $m(\mathbb{C})$, and $s(\mathbb{L}) + m(\mathbb{C})$, respectively.

The direct-sum framework is used by the generalized merge distance, which computes the shortest edit distance from a predicted clustering to the true clustering by performing binary splits and binary merges on clusters [30]. For the consistency of this thesis, we present and discuss their work by focusing on transforming the true clustering to the predicted clustering. Therefore, some terms related to split and merge are reversed compared to the original article, but this does not affect the understanding of their work. The generalized merge distance relies on the provision of a binary split cost function $g_s : \mathbb{R}^2 \rightarrow \mathbb{R}$ and a binary merge cost function $g_m : \mathbb{R}^2 \rightarrow \mathbb{R}$. Unfortunately, the distance between two clusterings is computed via a slice algorithm and no formula for the generalized merge distance is derived, which makes it hard to theoretically study the influence of the split/merge cost functions and compare it with other frameworks. We first show the connection between a binary split/merge cost function and a split/merge subcomponent measure. Based on the relation, we derive an analytical formula for the generalized merge distance.

Split/Merge Cost Function Given a split/merge subcomponent measure, it can be converted to a binary split/merge cost function. We show how to obtain the corresponding split cost function from a split subcomponent measure. Obtaining the binary merge cost function from a merge subcomponent measure can be done in a similar manner. Assume a true cluster L is split into two nonempty clusters L_1 and L_2 . If the split subcomponent measure s is a similarity measure, the split cost function is $g_s(|L_1|, |L_2|) = s(\{L\}) - s(\{L_1\}) - s(\{L_2\})$. If s is a distance measure, the split cost

function is $g_s(|L_1|, |L_2|) = s(\{L_1\}) + s(\{L_2\}) - s(\{L\})$.

$$g_s(|L_1|, |L_2|) = \begin{cases} s(\{L\}) - s(\{L_1\}) - s(\{L_2\}) & \text{if } s \text{ is a similarity measure,} \\ s(\{L_1\}) + s(\{L_2\}) - s(\{L\}) & \text{if } s \text{ is a distance measure.} \end{cases} \quad (7.3)$$

The split cost function g_s is a zero if s is one. If the split subcomponent measure $s(\mathbb{C}_L) = P(\mathbb{C}_L) = \sum_{C \in \mathbb{C}_L} \binom{|C|}{2}$ computes the number of positive pairs within a clustering, the split cost function is $\binom{|L|}{2} - \binom{|L_1|}{2} - \binom{|L_2|}{2} = (|L|^2 - |L_1|^2 - |L_2|^2)/2 = |L_1||L_2|$, which is defined in [30, Theorem 4.4]. If $s(\mathbb{C}_L) = \sum_{C \in \mathbb{C}_L} (|C| - 1)$ is defined as the number of links within a clustering to keep the clustering weakly connected, the split cost function is $(|L| - 1) - (|L_1| - 1) - (|L_2| - 1) = 1$, which is mentioned at the beginning of [30, Section 4.2]. If the split subcomponent measure $s(\mathbb{C}_L)$ is defined as $-\sum_{C \in \mathbb{C}_L} \frac{|C|}{n} \log \frac{|C|}{n}$, the split cost function becomes $\frac{|L|}{n} \log \frac{|L|}{n} - \frac{|L_1|}{n} \log \frac{|L_1|}{n} - \frac{|L_2|}{n} \log \frac{|L_2|}{n}$, which is defined in [30, Theorem 4.3].

Split/Merge Subcomponent Cost The cost of a split/merge subcomponent in terms of a split/merge subcomponent measure can be derived. Take a split subcomponent $\mathcal{G}_L$ as an example. There are two ways to transform $\{L\}$ to the induced clustering $\mathbb{C}_L$: a multiway split on L and a series of binary splits on L . The i -th cluster of $\mathbb{C}_L$ is denoted as $C_i \in \mathbb{C}_L$ with constraint that $\mathbb{C}_L = \bigcup_{i=1}^{|\mathbb{C}_L|} C_i$. The true cluster L is first split into two clusters C_1 and $L \setminus C_1$. The cluster $L \setminus C_1$ is further split into two clusters C_2 and $L \setminus C_1 \setminus C_2$, and so on. Therefore, $\{L\}$ can be transformed to the induced clustering $\mathbb{C}_L$ using $|\mathbb{C}_L| - 1$ binary splits. The i -th ($1 \leq i \leq |\mathbb{C}_L| - 2$) binary split generates the i -th cluster C_i , and the last binary split generates the last two clusters $C_{|\mathbb{C}_L|-1}$ and $C_{|\mathbb{C}_L|}$. The cost of the i -th binary split is $g_s(|C_i|, |L| - \sum_{j=1}^i |C_j|)$, and the cost of the split subcomponent $\mathcal{G}_L$ is $\sum_{i=1}^{|\mathbb{C}_L|-1} g_s(|C_i|, |L| - \sum_{j=1}^i |C_j|)$. If the split subcomponent measure satisfies $s(\mathbb{C}_L) = \sum_{C \in \mathbb{C}} s(\{C\})$, it can be simplified as follows according to (7.3).

$$\sum_{i=1}^{|\mathbb{C}_L|-1} g_s(|C_i|, |L| - \sum_{j=1}^i |C_j|) = \begin{cases} s(\{L\}) - s(\mathbb{C}_L) & \text{if } s \text{ is a similarity measure,} \\ s(\mathbb{C}_L) - s(\{L\}) & \text{if } s \text{ is a distance measure.} \end{cases}$$

Generalized Merge Distance When s is a similarity measure, the cost of the split subcomponent $\mathcal{G}_L$ is $s(\{L\}) - s(\mathbb{C}_L)$. The total cost of binary splits is the split distance $\sum_{L \in \mathbb{L}} (s(\{L\}) - s(\mathbb{C}_L)) = s(\mathbb{L}) - s(\mathbb{M})$, and the total cost of binary merges is the merge distance $\sum_{C \in \mathbb{C}} (m(\{C\}) - m(\mathbb{L}_C)) = m(\mathbb{C}) - m(\mathbb{M})$. The total cost of binary splits and binary merges is the generalized merge distance $s(\mathbb{L}) - s(\mathbb{M}) + m(\mathbb{C}) - m(\mathbb{M})$. If s and m are the same, then the generalized merge distance can be simplified as $s(\mathbb{L}) + s(\mathbb{C}) - 2s(\mathbb{M})$. When s is a distance measure, the split distance is $s(\mathbb{M}) - s(\mathbb{L})$, the merge distance is $m(\mathbb{M}) - m(\mathbb{C})$, and generalized merge distance is $s(\mathbb{M}) - s(\mathbb{L}) + m(\mathbb{M}) - m(\mathbb{C})$. If s and m are the same, the generalized merge distance can be simplified as $2s(\mathbb{M}) - s(\mathbb{L}) - s(\mathbb{C})$. As a result, the generalized merge distance is based on the direct-sum framework. The following are some examples of the generalized merge distances under the assumption that the split and the merge subcomponent measures are the same.

- The split cost function $g_s(x, y)$ is xy , and the split subcomponent measure is $P(\mathbb{C}_L)$. The split distance $P(\mathbb{L}) - P(\mathbb{M})$ is the number of false negative pairs, and the merge distance $P(\mathbb{C}) - P(\mathbb{M})$ is the number of false positive pairs. The generalized merge distance $P(\mathbb{L}) + P(\mathbb{C}) - 2P(\mathbb{M})$ is half of the Mirkin Metric [32].
- The split cost function $g_s(x, y)$ is 1, and the split subcomponent measure is $|\mathbb{C}_L|$. The split and merge distances are $|\mathbb{M}| - |\mathbb{L}|$ and $|\mathbb{M}| - |\mathbb{C}|$, respectively. The generalized merge distance $2|\mathbb{M}| - |\mathbb{L}| - |\mathbb{C}|$ is the basic merge distance [1, 30].
- The split cost function $g_s(x, y)$ is $h(x) + h(y) - h(x + y)$, where $h(x) = -\frac{x}{n} \log \frac{x}{n}$. The split distance $H(\mathbb{M}) - H(\mathbb{L})$ is the conditional entropy $H(\mathbb{C}|\mathbb{L})$, and the merge distance $H(\mathbb{M}) - H(\mathbb{C})$ is the conditional entropy $H(\mathbb{L}|\mathbb{C})$. The generalized merge distance $2H(\mathbb{M}) - H(\mathbb{L}) - H(\mathbb{C})$ is the variation of information [28].

7.5 Examples of the Framework

The split-merge framework (7.2) is flexible because the subcomponent similarity measures can be application specific. The application of the framework requires the provision of the split and merge subcomponent measures, which can be obtained by reusing existing external or internal measures. Property 7.1.3 lays the foundation for how to reuse an existing measure. A subcomponent similarity measure can be obtained by applying an existing measure to a subcomponent. However, not all directly derived subcomponent measures are normalized (Property 7.1.2), so we have to modify them accordingly.

7.5.1 Reusing Existing External Measures

The Rand index or the pairwise recall on a split subcomponent $\mathcal{G}_L$ is $\sum_{C \in \mathbb{C}_L} \binom{|C|}{2} / \binom{|L|}{2} = \sum_{C \in \mathbb{C}_L} (|C|(|C| - 1)) / (|L|(|L| - 1))$, which is the fraction of the positive pairs within the induced clustering $\mathbb{C}_L$. It is normalized so there is no need to further modify it. The classification accuracy or inverse purity of a split subcomponent $\mathcal{G}_L$ is $\max_{C \in \mathbb{C}_L} |C| / |L|$. It is unnormalized because it is lower bounded by $1/|L|$ and upper bounded by one. There are two approaches to make it normalized. The first method is to decrease the numbers by one, that is, $\max_{C \in \mathbb{C}_L} (|C| - 1) / (|L| - 1)$. The second solution is to scale the numbers by a logarithmic function, that is, $\max_{C \in \mathbb{C}_L} \log |C| / \log |L|$. The normalized variation of information by $\log n$ on a split subcomponent $\mathcal{G}_L$ is $(H(\{L\}) + H(\mathbb{C}_L) - 2I(\{L\}, \mathbb{C}_L)) / \log |L| = H(\mathbb{C}_L) / \log |L|$, which is the relative entropy of the induced clustering $\mathbb{C}_L$. It is normalized but it is a distance measure. We use its similarity counterpart $1 - H(\mathbb{C}_L) / \log |L| = \sum_{C \in \mathbb{C}_L} (|C| \log |C|) / (n \log n)$. In summary, we can obtain four different normalized split subcomponent measures

$$\begin{aligned} s_R(\mathbb{C}_L) &= \sum_{C \in \mathbb{C}_L} \frac{|C|(|C| - 1)}{|L|(|L| - 1)}, & s_H(\mathbb{C}_L) &= \sum_{C \in \mathbb{C}_L} \frac{|C| \log |C|}{|L| \log |L|}, \\ s_D(\mathbb{C}_L) &= \max_{C \in \mathbb{C}_L} \frac{|C| - 1}{|L| - 1}, & s_A(\mathbb{C}_L) &= \max_{C \in \mathbb{C}_L} \frac{\log |C|}{\log |L|}. \end{aligned}$$

s_R and s_H take both the number of clusters and the relative size of each cluster into account. S_A and S_D only uses the largest cluster of $\mathbb{C}_L$ and ignores the rest clusters. When the true cluster L is a singleton, a division-by-zero error occurs when evaluating the above measures. According to the first rule of Property 7.1.2, their values should be set to one.

Property 7.5.1. $s_R(\mathbb{C}_L) \leq s_H(\mathbb{C}_L) \leq s_A(\mathbb{C}_L)$ and $s_R(\mathbb{C}_L) \leq s_D(\mathbb{C}_L) \leq s_A(\mathbb{C}_L)$.

Proof. The convex combination of a set of positive numbers is upper bounded by the maximum of them, therefore $s_R(\mathbb{C}_L) \leq s_D(\mathbb{C}_L)$ and $s_H(\mathbb{C}_L) \leq s_A(\mathbb{C}_L)$.

When L is a singleton, their values are all ones. When L is not a singleton and $\mathbb{C}_L$ contains only singletons, their values are all zeros. The rest cases imply $2 \leq |C| \leq |L|$. Define a function $f(x) = (x - 1)/\log x$ with domain $x \in [2, \infty)$. Its derivative is

$$f'(x) = \frac{\log x - 1 + \frac{1}{x}}{(\log x)^2} \geq \frac{x - 1 - 1 + \frac{1}{x}}{(\log x)^2} \geq \frac{x - 2}{(\log x)^2} \geq 0.$$

It is a monotonically increasing function. When $2 \leq |C| \leq |L|$,

$$f(2) = 1 \leq f(|C|) = \frac{|C| - 1}{\log |C|} \leq f(|L|) = \frac{|L| - 1}{\log |L|} \implies \frac{|C| - 1}{|L| - 1} \leq \frac{\log |C|}{\log |L|}.$$

As a result, $s_R(\mathbb{C}_L) \leq s_H(\mathbb{C}_L)$ and $s_D(\mathbb{C}_L) \leq s_A(\mathbb{C}_L)$. □

A symmetric measure of the split-merge framework 7.2 can be obtained by letting the merge subcomponent measure be the same with the split subcomponent measure. Using subcomponent measures s_R , s_H , s_D , and s_A to score both split and merge subcomponents, four similarity measures S_R , S_H , S_D , and S_A can be obtained, respectively. With Property 7.5.1 and Property 7.3.17, it is obvious that $S_R(\mathbb{L}, \mathbb{C}) \leq S_H(\mathbb{L}, \mathbb{C}) \leq S_A(\mathbb{L}, \mathbb{C})$ and $S_R(\mathbb{L}, \mathbb{C}) \leq S_D(\mathbb{L}, \mathbb{C}) \leq S_A(\mathbb{L}, \mathbb{C})$. They offer different granularity and can be used according to application needs. We suggest to use S_H due to the already identified advantages of information theoretic measures. S_D and S_A are not discriminative enough.

Entropy The similarities of a split subcomponent $\mathcal{G}_L$ and a merge subcomponent $\mathcal{G}_C$ are $s_H(\mathbb{C}_L) = 1 - H(\mathbb{C}_L)/\log |L|$ and $s_H(\mathbb{L}_C) = 1 - H(\mathbb{L}_C)/\log |C|$, respectively. Substituting them into the split-merge framework (7.2) gives the similarity measure S_H

$$S_H(\mathbb{L}, \mathbb{C}) = \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \left(1 - \frac{H(\mathbb{C}_L)}{\log |L|} \right) \left(1 - \frac{H(\mathbb{L}_C)}{\log |C|} \right).$$

7.5.2 Reusing Existing Internal Measures

The split-merge framework (7.2) can make use of the feature vectors of objects or the distances between objects when they are available. In contrast, such information is ignored by previous external measures [7]. Many internal compactness and separation measures can be used as subcomponent measures [23].

Mean Squared Error We give an example using the normalized mean squared error to score a subcomponent. For a split subcomponent $\mathcal{G}_L$, the mean squared error of $\mathbb{C}_L$ is $mse(\mathbb{C}_L) = \sum_{C \in \mathbb{C}_L} \sum_{i \in C} (v_i - v_C)^2 / n$, where v_i is the feature vector of the i -th object, and $v_C = \sum_{i \in C} v_i / |C|$ is the center of a cluster $C \in \mathbb{C}_L$. It is lower bounded by zero and upper bounded by $mse(\{L\})$. The similarity of a split subcomponent $\mathcal{G}_L$ is defined as the normalized mean squared error

$$nmse(\mathbb{C}_L) = \frac{mse(\mathbb{C}_L)}{mse(\{L\})} = \frac{\sum_{C \in \mathbb{C}_L} \sum_{i \in C} (v_i - v_C)^2}{\sum_{i \in L} (v_i - v_L)^2},$$

where $v_L = \sum_{i \in L} v_i / |L|$ is the center of cluster L . Similarly, the similarity of a merge subcomponent is $nmse(\mathbb{L}_C)$. Using these subcomponent similarity measures, we can obtain a similarity measure based on the split-merge framework (7.2). It is efficient to compute since the time complexity is $O(n)$. In contrast, the hybrid measure proposed in [7] requires $O(n^{2.6})$ time in average case and $O(n^3 \log n)$ time in the worst case.

Table 7.1: The formulae and properties of the selected similarity measures to be compared. N, CN, JW, and SC refer to normalized, conditional normalization (Definition 3.0.1), joint-weighted decomposition on component (Property 6.2.3), and subcomponent consistency (Property 7.1.3), respectively.

Measure	Formula	N	CN	JW	SC
R	$[\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + 2P(\mathbb{M})]/\binom{n}{2}$	✓	×	×	×
N	$\frac{1}{2} \sum_{L \in \mathbb{L}} \max_{C \in \mathbb{C}} \frac{ L \cap C }{n} + \frac{1}{2} \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} \frac{ L \cap C }{n}$	×	×	✓	×
A	$\max_{\hat{w}} \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \hat{w}(L, C) \frac{ L \cap C }{n}$	×	×	✓	✓
NMI	$I(\mathbb{L}, \mathbb{C}) / \max\{H(\mathbb{L}), H(\mathbb{C})\}$	✓	×	×	×
V	$[\log n - H(\mathbb{M}) + I(\mathbb{L}, \mathbb{C})] / \log n$	✓	×	×	×
K	$[\log k^2 - H(\mathbb{M}) + I(\mathbb{L}, \mathbb{C})] / \log k^2$	✓	×	✓	×
S_H	$\sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{ L \cap C }{n} s_H(\mathbb{C}_L) s_H(\mathbb{L}_C)$	✓	✓	✓	✓

7.6 Comparisons with Existing Measures

We compare the representative similarity measure S_H of the split-merge framework with a few representative symmetric normalized measures from different categories. For a distance measure, we study its counterpart similarity measure by subtracting it from one. We choose the Rand index (R), the similarity counterpart of the constrained Van Dongen metric (N), the classification accuracy (A), the normalized mutual information by the maximum entropy (NMI), the similarity counterpart of the variation of information normalized by $\log n$ (V), and the similarity counterpart of the variation of information normalized by $\log k^2$ (K).

7.6.1 Theoretical Comparisons

The properties of the measures in terms of normalization, conditional normalization, joint-weighted decomposition on component, and subcomponent consistency are summarized in Table 7.1. The first two properties have been studied in Chapter 4. We will study the remaining two properties. Table 7.2 lists the decomposition of a few similarity measures, which are derived in Appendix B. From the table, we find their w and b do

Table 7.2: The component-based decomposition of the selected similarity measures. The decomposition of S_H can be inferred from Property 7.3.14 since S_H is an instance of S^* .

Measure(s)	Component Weight $w(J, n)$	Bias Term $b(\mathbb{J}, n)$
R	$(J (J - 1))/(n(n - 1))$	$1 - \sum_{J \in \mathbb{J}} (J (J - 1))/(n(n - 1))$
I	$ J /n$	$\log n - \sum_{J \in \mathbb{J}} J \log J /n$
NMI	$\frac{ J }{n} \frac{\max\{H(\mathbb{L}_J), H(\mathbb{C}_J)\}}{\max\{H(\mathbb{L}), H(\mathbb{C})\}}$	$\log n - \sum_{J \in \mathbb{J}} \frac{ J }{n} \log J / \max\{H(\mathbb{L}), H(\mathbb{C})\}$
V	$(J \log J)/(n \log n)$	$1 - \sum_{J \in \mathbb{J}} (J \log J)/(n \log n)$
N, A, K, S_H	$ J /n$	0

not explicitly depend on the two clusterings $\mathbb{L}$ and $\mathbb{C}$. Instead, they rely more closely on the join.

$$S(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} w(J, n) S(\mathbb{L}_J, \mathbb{C}_J) + b(\mathbb{J}, n).$$

We study whether some existing similarity measures satisfy Property 6.2.3 and Property 7.1.3.

Proposition 7.6.1. *Only N , A , K , and S_H are join-weighted decomposable on components. Among them, only A and S_H are subcomponent consistent.*

Proof. It is easy to verify that only N , A , K , and S^* are join-weighted decomposable from Table 7.2. N and K are subcomponent inconsistent because they are instances of S' (see Section 7.2.1). Property 7.3.15 implies S_H is subcomponent consistent. The classification accuracy A is subcomponent consistent because $A(\{L\}, \mathbb{C}_L) = \max_{C \in \mathbb{C}_L} |C|/|L| = a(\mathbb{C}_L)$ and $A(\mathbb{L}_C, \{C\}) = a(\mathbb{L}_C)$, where a is the relevant subcomponent measure. $\square$

7.6.2 Empirical Comparisons

We compare the conditional normalization and monotonically decreasing properties of various measures using the coreference resolution task. This task is to group noun phrases (objects) that refer into the same real-world entity (clusters) [34]. It is an important

Figure 7.2: Measures on an ACE05 document for the coreference task as the clustering worsens from the best to a worst. The bottom clustering $\perp$ is obtained at the 33rd operation.

problem in natural language processing (NLP) and has been extensively studied in recent years [36]. The ϕ_3 -CEAF measure [25] is frequently used to evaluate the performance of coreference algorithms, and it is the same as the classification accuracy A . Those selected similarity measures given in Section 7.6 except the K measure will be compared since it is unknown how to choose parameter k . Our experiments use a randomly selected document in the ACE-2005 English data set [36].

We construct a series of predicted clusterings from the best to a worst with respect to a given clustering. This series serves to evaluate the selected similarity measures with respect to this desideratum: a reasonable similarity score should decrease strictly from one to zero as the clustering “worsens” from the best to a worst. We use two operations to construct the series: (a) a binary split operation that splits the largest non-singleton cluster into two near equal-sized clusters; and (b) a binary merge operation that either merges two true singletons into one cluster or merges a true singleton with a randomly selected cluster if there is only one singleton, where a true singleton is one that is in the true clustering. Given a true clustering, we first apply the binary split operation

repeatedly to transform the true clustering to $\perp$. Then we apply the binary merge operation repeatedly to transform $\perp$ to a worst-clustering of the true clustering.

Figure 7.2 plots their similarities as the number of operations increases. Although all the measures decreases as the generated clustering worsens, only S_H decreases from one to zero. This is because, among these measures, only S_H is conditionally normalized. In addition, the rest measures are rather far from zero at the worst-clustering. The figure also shows that S_H is strictly decreasing. This is also satisfied by R , NMI and V but not by set matching measures A and N . The experiment confirmed again that the Rand index gives very high values when there are many predicted clusters. Both N and A ignore those unmatched clusters and their scores remain the same when the predicted clustering become worse in some situations. Besides, the running time of the classification accuracy A is very expensive.

Chapter 8

Conclusions

We proposed a conditional normalization property to better suit the task of external clustering evaluation than the traditional normalization property. A method is proposed to determine the normalization property of a measure, and it was applied to existing measures. We gave an extensive survey of existing measures from the normalization point of view, and we extended the three traditional categories by proposing a new category and refining an existing category. A generative model is proposed to study how measures within different categories are generated in a systematic manner, which helps us to study existing measures as well as producing new measures according to different applications. We proposed a graph-based model of clusterings to study the intrinsic relation between two clusterings. The graph contains many independent components, which allows us to decompose almost any measure as a conical combination of component scores plus an additional term. We decomposed existing measures and found they actually differ in their weighting schemes. Based on the graph-based model, we proposed a split-merge framework of similarity measures and proved many useful properties of it. We compared it with other existing frameworks. It can create similarity measures according to different application requirements by providing a measure to score a subcomponent. We presented some examples of the framework and compared a selected example with a few representative measures theoretically and empirically to reveal its advanced properties.

Appendix A

Conditional Normalization

The conditional normalization property of normalized measures are discussed. Two operators are used in the best/worst clustering condition. AND is denoted by $x \wedge y$, whose value is one if $x = y = 1$ and zero otherwise. OR is denoted by $x \vee y$, whose value is zero if $x = y = 0$ and one otherwise. The best clustering conditions are derived for normalized asymmetric measures to show that they are not conditionally normalized. The worst clustering conditions are derived for all normalized symmetric measures. For a normalized similarity measure, the worst clustering condition can be obtained by setting the numerator to zero and the denominator to nonzero. Unfortunately, all normalized measures discussed in the thesis are not conditionally normalized.

A.1 Pair Counting Measures

The worst clustering condition of $P(\mathbb{M})$ is $\mathbb{M} = \perp$. The set $\Omega_{\perp}^w$ is $\{\mathbb{C} \mid \mathbb{C} \in \Omega, \mathbb{M} = \perp, \mathbb{C} \neq \mathbb{L}\}$. The meet between the bottom and any clustering is still the bottom. The set $\Omega_{\perp}^w$ of the worst clusterings associated with the bottom is Ω , which violates the requirements of conditionally normalized measures. It is hard to be rectified by its normalized variants.

The pairwise F_1 measure $2P(\mathbb{M})/(P(\mathbb{L})+P(\mathbb{C}))$ and the Jaccard index $P(\mathbb{M})/(P(\mathbb{L})+P(\mathbb{C}) - P(\mathbb{M}))$ share the same worst clustering condition $\mathbb{M} = \perp \wedge (\mathbb{L} \neq \perp \vee \mathbb{C} \neq \perp)$.

The additional condition is derived as follows

$$P(\mathbb{L}) + P(\mathbb{C}) \neq 0 \implies P(\mathbb{L}) \neq 0 \vee P(\mathbb{C}) \neq 0 \implies \mathbb{L} \neq \perp \vee \mathbb{C} \neq \perp.$$

When both $\mathbb{L}$ and $\mathbb{C}$ are $\perp$, a divide-by-zero error raises because $P(\mathbb{L}) + P(\mathbb{C}) = 0$. A simple solution is to set the score to one since the two clusterings are the same. As a result, they are not conditionally normalized because $\Omega_{\perp}^b = \{\perp\}$, $\Omega_{\perp}^w = \Omega \setminus \{\perp\}$, and $\Omega_{\perp}^n = \emptyset$.

The worst clustering condition of the Fowlkes-Mallows index $P(\mathbb{M})/\sqrt{P(\mathbb{L})P(\mathbb{C})}$ is $\mathbb{M} = \perp \wedge (\mathbb{L} \neq \perp \wedge \mathbb{C} \neq \perp)$. The additional condition is derived as follows

$$P(\mathbb{L})P(\mathbb{C}) \neq 0 \implies P(\mathbb{L}) \neq 0 \wedge P(\mathbb{C}) \neq 0 \implies \mathbb{L} \neq \perp \wedge \mathbb{C} \neq \perp.$$

A divide-by-zero error raises if either $\mathbb{L}$ or $\mathbb{C}$ is $\perp$. When both of them are $\perp$, the score can be set to one. However, no quick fix is given when $\mathbb{L}$ is $\perp$ and $\mathbb{C}$ is not $\perp$. This causes the three categories of clusterings to be undetermined and the Fowlkes-Mallows index to be not conditionally normalized.

The best clustering conditions of the pairwise precision $P(\mathbb{M})/P(\mathbb{C})$ and the pairwise recall $P(\mathbb{M})/P(\mathbb{L})$ are $\mathbb{M} = \mathbb{C}$ and $\mathbb{M} = \mathbb{L}$, respectively. They are not conditionally normalized because the best clustering condition is not $\mathbb{L} = \mathbb{C}$. The worst clustering condition of the pairwise precision is $\mathbb{M} = \perp \wedge \mathbb{C} \neq \perp$, and that of the pairwise recall is $\mathbb{M} = \perp \wedge \mathbb{L} \neq \perp$. They suffer from similar problems as the Fowlkes-Mallows index.

The worst clustering condition of the Rand index $[(\binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + 2P(\mathbb{M}))]/\binom{n}{2}$ is $\mathbb{M} = \perp \wedge (\mathbb{L} = \top \vee \mathbb{C} = \top)$, which is derived based on the following inference

$$\begin{aligned} \binom{n}{2} - P(\mathbb{L}) - P(\mathbb{C}) + 2P(\mathbb{M}) = 0 &\implies P(\mathbb{M}) = 0 \wedge P(\mathbb{L}) + P(\mathbb{C}) = \binom{n}{2} \\ \implies P(\mathbb{M}) = 0 \wedge P(\mathbb{L}) + P(\mathbb{C}) = P(\top) &\implies \mathbb{M} = \perp \wedge (\mathbb{L} = \top \vee \mathbb{C} = \top). \end{aligned}$$

The worst clustering condition is true only when $\mathbb{L}$ is $\top$ or $\perp$, so the Rand index is not conditionally normalized.

A.2 Cluster Counting Measures

The worst clustering condition of $|\mathbb{L} \cap \mathbb{C}|$ is $\mathbb{L} \cap \mathbb{C} = \emptyset$. The set $\Omega_{\mathbb{L}}^w$ is $\{\mathbb{C} \mid \mathbb{C} \in \Omega, \mathbb{L} \cap \mathbb{C} = \emptyset, \mathbb{C} \neq \mathbb{L}\}$. The intersection between the top and any other clustering is always empty, so the set $\Omega_{\top}^w$ of the worst clusterings associated with the top is $\Omega \setminus \{\top\}$, which violates the requirements of a conditionally normalized measure. The cluster-level precision $|\mathbb{L} \cap \mathbb{C}|/|\mathbb{C}|$, cluster-level recall $|\mathbb{L} \cap \mathbb{C}|/|\mathbb{L}|$, and cluster-level F_1 measure $2|\mathbb{L} \cap \mathbb{C}|/(|\mathbb{L}| + |\mathbb{C}|)$ are not conditionally normalized because they have the same worst clustering condition as $|\mathbb{L} \cap \mathbb{C}|$.

The worst clustering condition of $|\mathbb{M}|$ is $\mathbb{M} = \perp$, which is the same with that of $P(\mathbb{M})$. The worst clustering conditions of MUC precision $(n - |\mathbb{M}|)/(n - |\mathbb{C}|)$, MUC recall $(n - |\mathbb{M}|)/(n - |\mathbb{L}|)$, and MUC F_1 measure $2(n - |\mathbb{M}|)/(2n - |\mathbb{L}| - |\mathbb{C}|)$ are $\mathbb{M} = \perp \wedge \mathbb{C} \neq \perp$, $\mathbb{M} = \perp \wedge \mathbb{L} \neq \perp$, and $\mathbb{M} = \perp \wedge (\mathbb{L} \neq \perp \vee \mathbb{C} \neq \perp)$ respectively. The worst clustering condition of the normalized split distance $(|\mathbb{M}| - |\mathbb{C}|)/(n - 1)$ is $\mathbb{M} = \perp \wedge \mathbb{C} = \top$, and that of the normalized merge distance $(|\mathbb{M}| - |\mathbb{L}|)/(n - 1)$ is $\mathbb{M} = \perp \wedge \mathbb{L} = \top$. The above mentioned measures are not conditionally normalized based on the analysis of the normalization property of pair counting measures.

A.3 Set Matching Measures

Among all set matching measures mentioned in the thesis, only Goodman-Kruskal association index $(U(\mathbb{L}, \mathbb{C}) - \max_{L \in \mathbb{L}} \frac{|L|}{n}) / (1 - \max_{L \in \mathbb{L}} \frac{|L|}{n})$ is normalized. But it is not conditionally normalized because its best clustering condition is $\mathbb{M} = \mathbb{C}$.

A.4 Information Theoretic Measures

The best clustering condition of an asymmetric distance measure $H(\mathbb{C}|\mathbb{L})$ is given by

$$H(\mathbb{C}|\mathbb{L}) = 0 \implies H(\mathbb{M}) - H(\mathbb{L}) = 0 \implies \mathbb{M} = \mathbb{L}.$$

Therefore, $H(\mathbb{C}|\mathbb{L})/\log |\mathbb{C}|$ and $H(\mathbb{L}|\mathbb{C})/\log |\mathbb{L}|$ are not conditionally normalized because their best clustering conditions are not $\mathbb{L} = \mathbb{C}$.

The worst clustering condition of the mutual information is $I(\mathbb{L}, \mathbb{C}) = 0$. The set $\Omega_{\mathbb{L}}^w$ is $\{\mathbb{C} \mid \mathbb{C} \in \Omega, I(\mathbb{L}, \mathbb{C}) = 0\}$. The set $\Omega_{\mathbb{C}}^w$ of the worst clusterings associated with the top is Ω , which violates the requirements of conditionally normalized measures. It makes the normalized variants to be not conditionally normalized.

The best clustering conditions of two normalized asymmetric measures $I(\mathbb{L}, \mathbb{C})/H(\mathbb{C})$ and $I(\mathbb{L}, \mathbb{C})/H(\mathbb{L})$ are $\mathbb{M} = \mathbb{L} \wedge \mathbb{C} \neq \top$ and $\mathbb{M} = \mathbb{C} \wedge \mathbb{L} \neq \top$, respectively. They are not conditionally normalized because their best clustering conditions are not $\mathbb{L} = \mathbb{C}$. Their worst clustering conditions are $I(\mathbb{L}, \mathbb{C}) = 0 \wedge \mathbb{C} \neq \top$ and $I(\mathbb{L}, \mathbb{C}) = 0 \wedge \mathbb{L} \neq \top$, respectively.

$I(\mathbb{L}, \mathbb{C})/\min\{H(\mathbb{L}), H(\mathbb{C})\}$ and $I(\mathbb{L}, \mathbb{C})/\sqrt{H(\mathbb{L})H(\mathbb{C})}$ have the same worst clustering condition $I(\mathbb{L}, \mathbb{C}) = 0 \wedge (\mathbb{L} \neq \top \wedge \mathbb{C} \neq \top)$. The extra condition is derived below

$$\min\{H(\mathbb{L}), H(\mathbb{C})\} \neq 0 \implies H(\mathbb{L})H(\mathbb{C}) \neq 0 \implies \mathbb{L} \neq \top \wedge \mathbb{C} \neq \top.$$

$I(\mathbb{L}, \mathbb{C})/\max\{H(\mathbb{L}), H(\mathbb{C})\}$, $2I(\mathbb{L}, \mathbb{C})/(H(\mathbb{L})+H(\mathbb{C}))$, $2I(\mathbb{L}, \mathbb{C})/(\log |\mathbb{L}|+\log |\mathbb{C}|)$, and $I(\mathbb{L}, \mathbb{C})/H(\mathbb{M})$ have the same worst clustering condition $I(\mathbb{L}, \mathbb{C}) = 0 \wedge (\mathbb{L} \neq \top \vee \mathbb{C} \neq \top)$. The extra condition is derived below

$$\max\{H(\mathbb{L}), H(\mathbb{C})\} \neq 0 \implies H(\mathbb{L}) + H(\mathbb{C}) \neq 0,$$

$$\log |\mathbb{L}| + \log |\mathbb{C}| \neq 0 \implies H(\mathbb{L}) + H(\mathbb{C}) \neq 0,$$

$$I(\mathbb{L}, \mathbb{C}) = 0 \wedge H(\mathbb{M}) \neq 0 \implies H(\mathbb{L}) + H(\mathbb{C}) \neq 0,$$

$$H(\mathbb{L}) + H(\mathbb{C}) \neq 0 \implies \mathbb{L} \neq \top \vee \mathbb{C} \neq \top.$$

The worst clustering condition of $(H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}))/\log n$ is

$$H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}) = \log n \implies I(\mathbb{L}, \mathbb{C}) = 0 \wedge H(\mathbb{M}) = \log n \implies I(\mathbb{L}, \mathbb{C}) = 0 \wedge \mathbb{M} = \perp.$$

It is not conditionally normalized because the set of worst clusterings is empty when $\mathbb{L} = \{\{1, 2\}, \{3\}\}$. Replacing $\log n$ by $\log k^2$, the worst clustering condition of $(H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}))/\log k^2$ is $I(\mathbb{L}, \mathbb{C}) = 0 \wedge H(\mathbb{M}) = \log k^2$. Similarly, it is not conditionally normalized. This measure implicitly requires $K(\mathbb{L}, \mathbb{C}) \leq \log k^2$ to make sure it is normalized.

Appendix B

Component-Based Decomposition

The decomposition of a measure is based on the fact that components are disjoint.

B.1 Pair Counting Measures

The number of pairs $P(\mathbb{M})$ in the meet can be decomposed as

$$P(\mathbb{M}) = \sum_{M \in \mathbb{M}} \binom{|M|}{2} = \sum_{J \in \mathbb{J}} \sum_{M \in \mathbb{M}_J} \binom{|M|}{2} = \sum_{J \in \mathbb{J}} P(\mathbb{M}_J).$$

Similarly, $P(\mathbb{L})$ and $P(\mathbb{C})$ can be decomposed as $P(\mathbb{L}) = \sum_{J \in \mathbb{J}} P(\mathbb{L}_J)$ and $P(\mathbb{C}) = \sum_{J \in \mathbb{J}} P(\mathbb{C}_J)$, respectively. For a one-to-one component J , the component score $P(\mathbb{M}_J)$ is $\binom{|J|}{2}$. If the size of the one-to-one component is one, the component score is zero, which indicates the correctly matched singletons are ignored.

B.1.1 Asymmetric Measures Precision/Recall - Wallace Index

The two asymmetric measures $P_l(\mathbb{L}, \mathbb{C}) = P(\mathbb{M})/P(\mathbb{L})$ and $P_c(\mathbb{L}, \mathbb{C}) = P(\mathbb{M})/P(\mathbb{C})$ can be decomposed as

$$\begin{aligned} P_l(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{P(\mathbb{L}_J)}{P(\mathbb{L})} P_l(\mathbb{L}_J, \mathbb{C}_J) = \sum_{J \in \mathbb{J}} \frac{\sum_{L \in \mathbb{L}_J} |L|^2 - |J|}{\sum_{L \in \mathbb{L}} |L|^2 - n} P_l(\mathbb{L}_J, \mathbb{C}_J), \\ P_c(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{P(\mathbb{C}_J)}{P(\mathbb{C})} P_c(\mathbb{L}_J, \mathbb{C}_J) = \sum_{J \in \mathbb{J}} \frac{\sum_{C \in \mathbb{C}_J} |C|^2 - |J|}{\sum_{C \in \mathbb{C}} |C|^2 - n} P_c(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.1.2 Geometric Mean - Fowlkes-Mallows Index

$P_g(\mathbb{L}, \mathbb{C}) = P(\mathbb{M}) / \sqrt{P(\mathbb{L})P(\mathbb{C})}$ can be decomposed as $\sum_{J \in \mathbb{J}} \frac{\sqrt{P(\mathbb{L}_J)P(\mathbb{C}_J)}}{\sqrt{P(\mathbb{L})P(\mathbb{C})}} P_g(\mathbb{L}_J, \mathbb{C}_J)$. The difference between $P(\mathbb{L})P(\mathbb{C})$ and $\left(\sum_{J \in \mathbb{J}} \sqrt{P(\mathbb{L}_J)P(\mathbb{C}_J)}\right)^2$ is

$$\begin{aligned} & P(\mathbb{L})P(\mathbb{C}) - \left(\sum_{J \in \mathbb{J}} \sqrt{P(\mathbb{L}_J)P(\mathbb{C}_J)}\right)^2 \\ &= \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} P(\mathbb{L}_J)P(\mathbb{C}_{J'}) - \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} \sqrt{P(\mathbb{L}_J)P(\mathbb{C}_J)} \sqrt{P(\mathbb{L}_{J'})P(\mathbb{C}_{J'})} \\ &= \frac{1}{2} \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} \left(\sqrt{P(\mathbb{L}_J)P(\mathbb{C}_{J'})} - \sqrt{P(\mathbb{L}_{J'})P(\mathbb{C}_J)}\right)^2. \end{aligned}$$

$\sum_{J \in \mathbb{J}} \frac{\sqrt{P(\mathbb{L}_J)P(\mathbb{C}_J)}}{\sqrt{P(\mathbb{L})P(\mathbb{C})}} \leq 1$ with equality holds only when $P(\mathbb{L}_J)/P(\mathbb{C}_J) = c$ for all components, where c is a positive constant or infinity. The constant c is one if the true clustering and the predicted clustering are the same. If the true clustering has k_1 equal-sized clusters and the predicted clustering is constructed based on the true clustering by splitting each true cluster into k_2 equal-sized clusters, the constant c is $(nk_2 - k_1k_2)/(n - k_1k_2)$, where $n \geq k_1k_2$. If $n = k_1k_2$, the constant c is positive infinity.

B.1.3 Arithmetic Mean - Pairwise F_1 Measure

The arithmetic mean $P_a(\mathbb{L}, \mathbb{C}) = 2P(\mathbb{M})/(P(\mathbb{L}) + P(\mathbb{C}))$ can be decomposed as

$$\begin{aligned} P_a(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{P(\mathbb{L}_J) + P(\mathbb{C}_J)}{P(\mathbb{L}) + P(\mathbb{C})} P_a(\mathbb{L}_J, \mathbb{C}_J) \\ &= \sum_{J \in \mathbb{J}} \frac{(\sum_{L \in \mathbb{L}_J} |L|^2 + \sum_{C \in \mathbb{C}_J} |C|^2)/2 - |J|}{(\sum_{L \in \mathbb{L}} |L|^2 + \sum_{C \in \mathbb{C}} |C|^2)/2 - n} P_a(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.1.4 Union - Jaccard Index

$P_u(\mathbb{L}, \mathbb{C}) = P(\mathbb{M}) / (P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M}))$ can be decomposed as

$$\begin{aligned} P_u(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{P(\mathbb{L}_J) + P(\mathbb{C}_J) - P(\mathbb{M}_J)}{P(\mathbb{L}) + P(\mathbb{C}) - P(\mathbb{M})} P_u(\mathbb{L}_J, \mathbb{C}_J) \\ &= \sum_{J \in \mathbb{J}} \frac{(\sum_{L \in \mathbb{L}_J} |L|^2 + \sum_{C \in \mathbb{C}_J} |C|^2 - \sum_{M \in \mathbb{M}_J} |M|^2) - |J|}{(\sum_{L \in \mathbb{L}} |L|^2 + \sum_{C \in \mathbb{C}} |C|^2 - \sum_{M \in \mathbb{M}} |M|^2) - n} P_u(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.1.5 Largest Achievable Upper Bound $\binom{n}{2}$

$P_n(\mathbb{L}, \mathbb{C}) = P(\mathbb{M}) / \binom{n}{2}$ can be decomposed as

$$P_n(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{\binom{|J|}{2}}{\binom{n}{2}} P_n(\mathbb{L}_J, \mathbb{C}_J) = \sum_{J \in \mathbb{J}} \frac{|J|^2 - |J|}{n^2 - n} P_n(\mathbb{L}_J, \mathbb{C}_J).$$

Within the join, $\sum_{J \in \mathbb{J}} \binom{|J|}{2}$ is the total number of within-cluster pairs and $\binom{n}{2}$ is the total number of pairs. They are equal when the join has only one cluster. If the join contains at least two clusters, $\sum_{J \in \mathbb{J}} \binom{|J|}{2} < \binom{n}{2}$ since there is at least one between-cluster pair. Therefore, $\sum_{J \in \mathbb{J}} \binom{|J|}{2} \leq \binom{n}{2}$ with equality holds only when the join $\mathbb{J}$ is the top $\top$.

B.1.6 Rand Index

Based on the above decomposition, we can decompose Rand index as

$$\begin{aligned} R(\mathbb{L}, \mathbb{C}) &= 1 - (P(\mathbb{L}) + P(\mathbb{C}) - 2P(\mathbb{M})) / \binom{n}{2} \\ &= 1 - \sum_{J \in \mathbb{J}} \frac{|J|^2 - |J|}{n^2 - n} (P(\mathbb{L}_J) + P(\mathbb{C}_J) - 2P(\mathbb{M}_J)) / \binom{|J|}{2} \\ &= \sum_{J \in \mathbb{J}} \frac{|J|^2 - |J|}{n^2 - n} R(\mathbb{L}_J, \mathbb{C}_J) + 1 - \sum_{J \in \mathbb{J}} \frac{|J|^2 - |J|}{n^2 - n}. \end{aligned}$$

B.2 Cluster Counting Measures

Cluster counting measures are grouped into exact matching and overlap matching measures.

B.2.1 Exact Matching

B.2.1.1 Cluster-level Precision/Recall

The cluster-level measures are based on the following decompositions.

$$|\mathbb{L} \cap \mathbb{C}| = \sum_{J \in \mathbb{J}} |\mathbb{L}_J \cap \mathbb{C}_J|, \quad |\mathbb{L}| = \sum_{J \in \mathbb{J}} |\mathbb{L}_J|, \quad |\mathbb{C}| = \sum_{J \in \mathbb{J}} |\mathbb{C}_J|.$$

The cluster-level precision $CL_p(\mathbb{L}, \mathbb{C})$, the cluster-level recall $CL_r(\mathbb{L}, \mathbb{C})$ and the cluster-level F_1 measure $CL_f(\mathbb{L}, \mathbb{C})$ are decomposed as

$$\begin{aligned} CL_p(\mathbb{L}, \mathbb{C}) &= \frac{|\mathbb{L} \cap \mathbb{C}|}{|\mathbb{C}|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{C}_J|}{|\mathbb{C}|} \frac{|\mathbb{L}_J \cap \mathbb{C}_J|}{|\mathbb{C}_J|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{C}_J|}{|\mathbb{C}|} CL_p(\mathbb{L}_J, \mathbb{C}_J), \\ CL_r(\mathbb{L}, \mathbb{C}) &= \frac{|\mathbb{L} \cap \mathbb{C}|}{|\mathbb{L}|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J|}{|\mathbb{L}|} \frac{|\mathbb{L}_J \cap \mathbb{C}_J|}{|\mathbb{L}_J|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J|}{|\mathbb{L}|} CL_r(\mathbb{L}_J, \mathbb{C}_J), \\ CL_f(\mathbb{L}, \mathbb{C}) &= \frac{2|\mathbb{L} \cap \mathbb{C}|}{|\mathbb{L}| + |\mathbb{C}|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J| + |\mathbb{C}_J|}{|\mathbb{L}| + |\mathbb{C}|} CL_f(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.2.2 Overlap Matching

The overlap matching measures are based on the following decompositions.

$$|\mathbb{M}| = \sum_{J \in \mathbb{J}} |\mathbb{M}_J|, \quad |\mathbb{L}| = \sum_{J \in \mathbb{J}} |\mathbb{L}_J|, \quad |\mathbb{C}| = \sum_{J \in \mathbb{J}} |\mathbb{C}_J|, \quad n = \sum_{J \in \mathbb{J}} |J|.$$

B.2.2.1 MUC Precision/Recall

MUC precision MUC_p and MUC recall MUC_r can be decomposed as

$$\begin{aligned} MUC_p(\mathbb{L}, \mathbb{C}) &= \frac{n - |\mathbb{M}|}{n - |\mathbb{C}|} = \sum_{J \in \mathbb{J}} \frac{|J| - |\mathbb{C}_J|}{n - |\mathbb{C}|} \frac{|J| - |\mathbb{M}_J|}{|J| - |\mathbb{C}_J|} = \sum_{J \in \mathbb{J}} \frac{|J| - |\mathbb{C}_J|}{n - |\mathbb{C}|} MUC_p(\mathbb{L}_J, \mathbb{C}_J), \\ MUC_r(\mathbb{L}, \mathbb{C}) &= \frac{n - |\mathbb{M}|}{n - |\mathbb{L}|} = \sum_{J \in \mathbb{J}} \frac{|J| - |\mathbb{L}_J|}{n - |\mathbb{L}|} \frac{|J| - |\mathbb{M}_J|}{|J| - |\mathbb{L}_J|} = \sum_{J \in \mathbb{J}} \frac{|J| - |\mathbb{L}_J|}{n - |\mathbb{L}|} MUC_r(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

MUC measures ignore correctly matched singletons. If a component J is a singleton, the component weights used in computing the precision and the recall are both zeros.

B.2.2.2 Split/Merge Distance

The normalized split distance $SM_s(\mathbb{L}, \mathbb{C})$ can be decomposed as

$$SM_s(\mathbb{L}, \mathbb{C}) = \frac{|\mathbb{M}| - |\mathbb{C}|}{n - 1} = \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} \frac{|\mathbb{M}_J| - |\mathbb{C}_J|}{|J| - 1} = \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} SM_s(\mathbb{L}_J, \mathbb{C}_J).$$

Its corresponding similarity measure $1 - SM_s(\mathbb{L}, \mathbb{C})$ can be decomposed as

$$\begin{aligned} 1 - SM_s(\mathbb{L}, \mathbb{C}) &= 1 - \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} SM_s(\mathbb{L}_J, \mathbb{C}_J) \\ &= \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} (1 - SM_s(\mathbb{L}_J, \mathbb{C}_J)) + 1 - \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1}. \end{aligned}$$

Similarly, the normalized merge distance $SM_m(\mathbb{L}, \mathbb{C})$ and its corresponding similarity variant $1 - SM_m(\mathbb{L}, \mathbb{C})$ can be decomposed as

$$\begin{aligned} SM_m(\mathbb{L}, \mathbb{C}) &= \frac{|\mathbb{M}| - |\mathbb{L}|}{n - 1} = \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} SM_m(\mathbb{L}_J, \mathbb{C}_J). \\ 1 - SM_m(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} (1 - SM_m(\mathbb{L}_J, \mathbb{C}_J)) + 1 - \sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1}. \end{aligned}$$

The total weight assigned to all components is $\sum_{J \in \mathbb{J}} \frac{|J| - 1}{n - 1} = (n - |\mathbb{J}|)/(n - 1) \leq 1$, that's why the similarity variant has an additional bias term.

B.3 Set Matching Measures

Set matching measures are grouped into local matching and global matching measures.

B.3.1 Local Matching

B.3.1.1 Purity and Inverse Purity

The purity PT_p and the inverse purity PT_r can be decomposed as

$$\begin{aligned} PT_p(\mathbb{L}, \mathbb{C}) &= \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} \frac{|L \cap C|}{n} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{C \in \mathbb{C}_J} \max_{L \in \mathbb{L}_J} \frac{|L \cap C|}{|J|} \right) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} PT_p(\mathbb{L}_J, \mathbb{C}_J), \\ PT_r(\mathbb{L}, \mathbb{C}) &= \sum_{L \in \mathbb{L}} \max_{C \in \mathbb{C}} \frac{|L \cap C|}{n} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{L \in \mathbb{L}_J} \max_{C \in \mathbb{C}_J} \frac{|L \cap C|}{|J|} \right) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} PT_r(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.3.1.2 Van Dongen Metric

The similarity variant of the normalized Van Dongen metric can be decomposed as

$$N(\mathbb{L}, \mathbb{C}) = \frac{1}{2}(PT_r(\mathbb{L}, \mathbb{C}) + PT_p(\mathbb{L}, \mathbb{C})) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} N(\mathbb{L}_J, \mathbb{C}_J).$$

B.3.1.3 Goodman-Kruskal Association Index

It is not easy to decompose the Goodman-Kruskal association index $GK(\mathbb{L}, \mathbb{C})$ directly.

Instead, we first decompose its distance variant as follows

$$1 - GK(\mathbb{L}, \mathbb{C}) = \frac{1 - \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} \frac{|L \cap C|}{n}}{1 - \max_{L \in \mathbb{L}} \frac{|L|}{n}} = \sum_{J \in \mathbb{J}} \frac{|J| - \max_{L \in \mathbb{L}_J} |L|}{n - \max_{L \in \mathbb{L}} |L|} (1 - GK(\mathbb{L}_J, \mathbb{C}_J)).$$

We can infer the decomposition of $GK(\mathbb{L}, \mathbb{C})$ as

$$GK(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J| - \max_{L \in \mathbb{L}_J} |L|}{n - \max_{L \in \mathbb{L}} |L|} GK(\mathbb{L}_J, \mathbb{C}_J) + 1 - \sum_{J \in \mathbb{J}} \frac{|J| - \max_{L \in \mathbb{L}_J} |L|}{n - \max_{L \in \mathbb{L}} |L|}.$$

$\sum_{J \in \mathbb{J}} (|J| - \max_{L \in \mathbb{L}_J} |L|) = n - \sum_{J \in \mathbb{J}} \max_{L \in \mathbb{L}_J} |L| \leq n - \max_{L \in \mathbb{L}} |L|$ with equality holds only when the join $\mathbb{J}$ is the top $\top$.

B.3.1.4 F-Purity and F-Inverse Purity

The F-purity FP_p and the F-inverse purity FP_r can be decomposed as

$$\begin{aligned} FP_p(\mathbb{L}, \mathbb{C}) &= \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} \frac{|C|}{n} \frac{2|L \cap C|}{|L| + |C|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{C \in \mathbb{C}_J} \max_{L \in \mathbb{L}_J} \frac{|C|}{|J|} \frac{2|L \cap C|}{|L| + |C|} \right) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} FP_p(\mathbb{L}_J, \mathbb{C}_J), \\ FP_r(\mathbb{L}, \mathbb{C}) &= \sum_{L \in \mathbb{L}} \max_{C \in \mathbb{C}} \frac{|L|}{n} \frac{2|L \cap C|}{|L| + |C|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{L \in \mathbb{L}_J} \max_{C \in \mathbb{C}_J} \frac{|L|}{|J|} \frac{2|L \cap C|}{|L| + |C|} \right) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} FP_r(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.3.1.5 J-Purity and J-Inverse Purity

The J-purity JP_p and the J-inverse purity JP_r can be decomposed as

$$\begin{aligned} JP_p(\mathbb{L}, \mathbb{C}) &= \sum_{C \in \mathbb{C}} \max_{L \in \mathbb{L}} \frac{1}{|C|} \frac{|L \cap C|}{|L \cup C|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{C}_J|}{|\mathbb{C}|} \left(\sum_{C \in \mathbb{C}_J} \max_{L \in \mathbb{L}_J} \frac{1}{|C_J|} \frac{|L \cap C|}{|L \cup C|} \right) = \sum_{J \in \mathbb{J}} \frac{|\mathbb{C}_J|}{|\mathbb{C}|} JP_p(\mathbb{L}_J, \mathbb{C}_J), \\ JP_r(\mathbb{L}, \mathbb{C}) &= \sum_{L \in \mathbb{L}} \max_{C \in \mathbb{C}} \frac{1}{|\mathbb{L}|} \frac{|L \cap C|}{|L \cup C|} = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J|}{|\mathbb{L}|} \left(\sum_{L \in \mathbb{L}_J} \max_{C \in \mathbb{C}_J} \frac{1}{|\mathbb{L}_J|} \frac{|L \cap C|}{|L \cup C|} \right) = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J|}{|\mathbb{L}|} JP_r(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

B.3.1.6 Global Matching

The decomposition of a global matching measure is based on the decomposition of the bipartite matching problem in (4.6). Since different components are disjoint, the weights between different clusters of different components are always zeros. If we arrange the objects component-by-component, the weight matrix $\hat{w}$ becomes a block diagonal matrix. As a result, optimizing $\hat{w}$ is equivalent to optimizing the weight matrix $\hat{w}_J$ of each component $\mathcal{G}_J$ independently. We decompose the optimal similarity $\Phi_g(\mathbb{L}, \mathbb{C})$ in (4.6) as

$$\Phi_g(\mathbb{L}, \mathbb{C}) = \max_{\hat{w}} \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \hat{w}(L, C) \phi(L, C) = \sum_{J \in \mathbb{J}} \max_{\hat{w}_J} \sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} \hat{w}(L, C) \phi(L, C) = \sum_{J \in \mathbb{J}} \Phi_g(\mathbb{L}_J, \mathbb{C}_J).$$

The possibly unnormalized similarity measure $\Phi_g(\mathbb{L}, \mathbb{C})$ has two trivial upper bounds $\Phi_g(\mathbb{L}, \mathbb{L})$ and $\Phi_g(\mathbb{C}, \mathbb{C})$. The following normalized measures can be obtained

$$\begin{aligned} \Phi_g^c(\mathbb{L}, \mathbb{C}) &= \frac{\Phi_g(\mathbb{L}, \mathbb{C})}{\Phi_g(\mathbb{C}, \mathbb{C})} = \sum_{J \in \mathbb{J}} \frac{\Phi_g(\mathbb{C}_J, \mathbb{C}_J)}{\Phi_g(\mathbb{C}, \mathbb{C})} \frac{\Phi_g(\mathbb{L}_J, \mathbb{C}_J)}{\Phi_g(\mathbb{C}_J, \mathbb{C}_J)} = \sum_{J \in \mathbb{J}} \frac{\Phi_g(\mathbb{C}_J, \mathbb{C}_J)}{\Phi_g(\mathbb{C}, \mathbb{C})} \Phi_g^c(\mathbb{L}_J, \mathbb{C}_J), \\ \Phi_g^l(\mathbb{L}, \mathbb{C}) &= \frac{\Phi_g(\mathbb{L}, \mathbb{C})}{\Phi_g(\mathbb{L}, \mathbb{L})} = \sum_{J \in \mathbb{J}} \frac{\Phi_g(\mathbb{L}_J, \mathbb{L}_J)}{\Phi_g(\mathbb{L}, \mathbb{L})} \frac{\Phi_g(\mathbb{L}_J, \mathbb{C}_J)}{\Phi_g(\mathbb{L}_J, \mathbb{L}_J)} = \sum_{J \in \mathbb{J}} \frac{\Phi_g(\mathbb{L}_J, \mathbb{L}_J)}{\Phi_g(\mathbb{L}, \mathbb{L})} \Phi_g^l(\mathbb{L}_J, \mathbb{C}_J). \end{aligned}$$

If $\Phi_g(\mathbb{L}, \mathbb{L})$ and $\Phi_g(\mathbb{C}, \mathbb{C})$ are the same, the two normalized measures are the same and they are symmetric. Otherwise, the two normalized measures are asymmetric.

Classification Accuracy The classification accuracy uses $\phi(L, C) = |L \cap C|$, resulting $\Phi_g(\mathbb{L}, \mathbb{L}) = \Phi_g(\mathbb{C}, \mathbb{C}) = n$ and $\Phi_g(\mathbb{L}_J, \mathbb{L}_J) = \Phi_g(\mathbb{C}_J, \mathbb{C}_J) = |J|$. Therefore, the classification accuracy can be decomposed as $A(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} A(\mathbb{L}_J, \mathbb{C}_J)$.

ϕ_4 -CEAF Precision/Recall ϕ_4 -CEAF uses $\phi(L, C) = \frac{2|L \cap C|}{|L| + |C|}$, resulting $\Phi_g(\mathbb{L}, \mathbb{L}) = |\mathbb{L}|$, $\Phi_g(\mathbb{C}, \mathbb{C}) = |\mathbb{C}|$, $\Phi_g(\mathbb{L}_J, \mathbb{L}_J) = |\mathbb{L}_J|$, and $\Phi_g(\mathbb{C}_J, \mathbb{C}_J) = |\mathbb{C}_J|$. As a result, ϕ_4 -CEAF precision $\Phi_4^p(\mathbb{L}, \mathbb{C})$ and ϕ_4 -CEAF recall $\Phi_4^r(\mathbb{L}, \mathbb{C})$ can be decomposed as

$$\Phi_4^p(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|\mathbb{C}_J|}{|\mathbb{C}|} \Phi_4^p(\mathbb{L}_J, \mathbb{C}_J), \quad \Phi_4^r(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|\mathbb{L}_J|}{|\mathbb{L}|} \Phi_4^r(\mathbb{L}_J, \mathbb{C}_J).$$

B.4 Information Theoretic Measures

The decompositions of most information theoretic measures are based on the decompositions of the joint entropy $H(\mathbb{M})$, the mutual information $I(\mathbb{L}, \mathbb{C})$, the entropy $H(\mathbb{L})$ of the true clustering, and the entropy $H(\mathbb{C})$ of the predicted clustering.

The joint entropy $H(\mathbb{M})$ can be decomposed as

$$\begin{aligned} H(\mathbb{M}) &= \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \log \frac{n}{|L \cap C|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sum_{L \in \mathbb{L}_J} \sum_{C \in \mathbb{C}_J} \frac{|L \cap C|}{|J|} \log \frac{1}{|L \cap C|} \right) + \log n \\ &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} (H(\mathbb{M}_J) - \log |J|) + \log n = \sum_{J \in \mathbb{J}} \frac{|J|}{n} H(\mathbb{M}_J) + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|. \end{aligned}$$

Similarly, the mutual information and the two entropies can be decomposed as

$$\begin{aligned} I(\mathbb{L}, \mathbb{C}) &= \sum_{L \in \mathbb{L}} \sum_{C \in \mathbb{C}} \frac{|L \cap C|}{n} \log \frac{n|L \cap C|}{|L||C|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} I(\mathbb{L}_J, \mathbb{C}_J) + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|, \\ H(\mathbb{L}) &= \sum_{L \in \mathbb{L}} \frac{|L|}{n} \log \frac{n}{|L|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} H(\mathbb{L}_J) + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|, \\ H(\mathbb{C}) &= \sum_{C \in \mathbb{C}} \frac{|C|}{n} \log \frac{n}{|C|} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} H(\mathbb{C}_J) + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|. \end{aligned}$$

B.4.1 Conditional Entropy

The conditional entropies can be decomposed as

$$\begin{aligned} H(\mathbb{L}|\mathbb{C}) &= H(\mathbb{M}) - H(\mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} (H(\mathbb{M}_J) - H(\mathbb{C}_J)) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} H(\mathbb{L}_J|\mathbb{C}_J), \\ H(\mathbb{C}|\mathbb{L}) &= H(\mathbb{M}) - H(\mathbb{L}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} (H(\mathbb{M}_J) - H(\mathbb{L}_J)) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} H(\mathbb{C}_J|\mathbb{L}_J). \end{aligned}$$

B.4.2 Asymmetric Measures

The two asymmetric measures $I_l(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C})/H(\mathbb{L})$ and $I_c(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C})/H(\mathbb{C})$ can be decomposed as

$$\begin{aligned} I_l(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{H(\mathbb{L}_J)}{H(\mathbb{L})} I_l(\mathbb{L}_J, \mathbb{C}_J) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{H(\mathbb{L})} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - \sum_{L \in \mathbb{L}_J} |L| \log |L|}{n \log n - \sum_{L \in \mathbb{L}} |L| \log |L|} I_l(\mathbb{L}_J, \mathbb{C}_J) + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - \sum_{L \in \mathbb{L}} |L| \log |L|}, \\ I_c(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - \sum_{C \in \mathbb{C}_J} |C| \log |C|}{n \log n - \sum_{C \in \mathbb{C}} |C| \log |C|} I_c(\mathbb{L}_J, \mathbb{C}_J) + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - \sum_{C \in \mathbb{C}} |C| \log |C|}. \end{aligned}$$

B.4.3 Minimum

$I_m(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C}) / \min\{H(\mathbb{L}), H(\mathbb{C})\}$ can be decomposed as

$$\begin{aligned} I_m(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{\min\{H(\mathbb{L}_J), H(\mathbb{C}_J)\}}{\min\{H(\mathbb{L}), H(\mathbb{C})\}} I_m(\mathbb{L}_J, \mathbb{C}_J) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{\min\{H(\mathbb{L}), H(\mathbb{C})\}} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - \max\{\sum_{L \in \mathbb{L}_J} |L| \log |L|, \sum_{C \in \mathbb{C}_J} |C| \log |C|\}}{n \log n - \max\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}} I_m(\mathbb{L}_J, \mathbb{C}_J) \\ &\quad + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - \max\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}}. \end{aligned}$$

The sum of the component weights and the bias term is

$$\frac{n \log n - \sum_{J \in \mathbb{J}} \max\{\sum_{L \in \mathbb{L}_J} |L| \log |L|, \sum_{C \in \mathbb{C}_J} |C| \log |C|\}}{n \log n - \max\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}} \leq 1.$$

The equality holds when the two clusterings are the same. In some situations, the measure $I_m(\mathbb{L}, \mathbb{C})$ behaves like asymmetric measures. If $\mathbb{L}$ is refined by $\mathbb{C}$, then $H(\mathbb{M}) = H(\mathbb{C})$ and $H(\mathbb{L}) \leq H(\mathbb{C})$, resulting $I_m(\mathbb{L}, \mathbb{C}) = 1$. If $\mathbb{L}$ refines $\mathbb{C}$, then $H(\mathbb{M}) = H(\mathbb{L})$ and $H(\mathbb{C}) \leq H(\mathbb{L})$, making $I_m(\mathbb{L}, \mathbb{C}) = 1$. Therefore, normalizing the mutual information by the minimum entropy ignores the larger entropy.

B.4.4 Geometric Mean

$I_g(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C}) / \sqrt{H(\mathbb{L})H(\mathbb{C})}$ can be decomposed as

$$I_g(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{\sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)}}{\sqrt{H(\mathbb{L})H(\mathbb{C})}} I_g(\mathbb{L}_J, \mathbb{C}_J) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{\sqrt{H(\mathbb{L})H(\mathbb{C})}}.$$

The square of the sum of the weights and bias term multiplied by $H(\mathbb{L})H(\mathbb{C})$ is

$$\begin{aligned} & \left(\sum_{J \in \mathbb{J}} \frac{|J|}{n} \sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)} + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J| \right)^2 \\ &= \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} \frac{|J|}{n} \frac{|J'|}{n} \sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)} \sqrt{H(\mathbb{L}_{J'})H(\mathbb{C}_{J'})} + (\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|)^2 \\ &+ (\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|) \sum_{J \in \mathbb{J}} \frac{|J|}{n} 2\sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)}. \end{aligned}$$

The multiplication of the two entropies is

$$\begin{aligned} H(\mathbb{L})H(\mathbb{C}) &= \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} \frac{|J|}{n} \frac{|J'|}{n} H(\mathbb{L}_J)H(\mathbb{C}_{J'}) + (\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|)^2 \\ &+ (\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|) \sum_{J \in \mathbb{J}} \frac{|J|}{n} (H(\mathbb{L}_J) + H(\mathbb{C}_J)). \end{aligned}$$

The difference between them is

$$\begin{aligned} & H(\mathbb{L})H(\mathbb{C}) - \left(\sum_{J \in \mathbb{J}} \frac{|J|}{n} \sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)} + \log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J| \right)^2 \\ &= \frac{1}{2} \sum_{J \in \mathbb{J}} \sum_{J' \in \mathbb{J}} \frac{|J|}{n} \frac{|J'|}{n} \left(\sqrt{H(\mathbb{L}_J)H(\mathbb{C}_{J'})} - \sqrt{H(\mathbb{L}_{J'})H(\mathbb{C}_J)} \right)^2 \\ &+ (\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|) \sum_{J \in \mathbb{J}} \frac{|J|}{n} \left(\sqrt{H(\mathbb{L}_J)} - \sqrt{H(\mathbb{C}_J)} \right)^2. \end{aligned}$$

It reaches zero only when $H(\mathbb{L}_J) = H(\mathbb{C}_J)$ is satisfied for every component, which further requires the true clustering to be the same with the predicted clustering. Hence, $\sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{\sqrt{H(\mathbb{L}_J)H(\mathbb{C}_J)}}{\sqrt{H(\mathbb{L})H(\mathbb{C})}} + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{\sqrt{H(\mathbb{L})H(\mathbb{C})}} \leq 1$ with equality holds only when the two clusterings are the same.

B.4.5 Arithmetic Mean

$I_a(\mathbb{L}, \mathbb{C}) = 2I(\mathbb{L}, \mathbb{C})/(H(\mathbb{L}) + H(\mathbb{C}))$ can be decomposed as

$$\begin{aligned} I_a(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{H(\mathbb{L}_J) + H(\mathbb{C}_J)}{H(\mathbb{L}) + H(\mathbb{C})} I_a(\mathbb{L}_J, \mathbb{C}_J) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{(H(\mathbb{L}) + H(\mathbb{C}))/2} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - (\sum_{L \in \mathbb{L}_J} |L| \log |L| + \sum_{C \in \mathbb{C}_J} |C| \log |C|)/2}{n \log n - (\sum_{L \in \mathbb{L}} |L| \log |L| + \sum_{C \in \mathbb{C}} |C| \log |C|)/2} I_a(\mathbb{L}_J, \mathbb{C}_J) \\ &\quad + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - (\sum_{L \in \mathbb{L}} |L| \log |L| + \sum_{C \in \mathbb{C}} |C| \log |C|)/2}. \end{aligned}$$

The arithmetic mean $(H(\mathbb{L}) + H(\mathbb{C}))/2$ has the same weighting function and bias term as the mutual information, thus $I_a(\mathbb{L}, \mathbb{C})$ is convex decomposable on components.

B.4.6 Maximum

$I_x(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C})/\max\{H(\mathbb{L}), H(\mathbb{C})\}$ can be decomposed as

$$\begin{aligned} I_x(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{\max\{H(\mathbb{L}_J), H(\mathbb{C}_J)\}}{\max\{H(\mathbb{L}), H(\mathbb{C})\}} I_x(\mathbb{L}_J, \mathbb{C}_J) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{\max\{H(\mathbb{L}), H(\mathbb{C})\}} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - \min\{\sum_{L \in \mathbb{L}_J} |L| \log |L|, \sum_{C \in \mathbb{C}_J} |C| \log |C|\}}{n \log n - \min\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}} I_x(\mathbb{L}_J, \mathbb{C}_J) \\ &\quad + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - \min\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}}. \end{aligned}$$

The sum of the component weights and the bias term is

$$\frac{n \log n - \sum_{J \in \mathbb{J}} \min\{\sum_{L \in \mathbb{L}_J} |L| \log |L|, \sum_{C \in \mathbb{C}_J} |C| \log |C|\}}{n \log n - \min\{\sum_{L \in \mathbb{L}} |L| \log |L|, \sum_{C \in \mathbb{C}} |C| \log |C|\}} \geq 1.$$

. The equality holds when the two clusterings are the same. If $\mathbb{L}$ is refined by $\mathbb{C}$, then $H(\mathbb{M}) = H(\mathbb{C})$ and $H(\mathbb{L}) \leq H(\mathbb{C})$, making $I_x(\mathbb{L}, \mathbb{C}) = H(\mathbb{L})/H(\mathbb{C})$. If $\mathbb{L}$ refines $\mathbb{C}$, then $H(\mathbb{M}) = H(\mathbb{L})$ and $H(\mathbb{C}) \leq H(\mathbb{L})$, making $I_x(\mathbb{L}, \mathbb{C}) = H(\mathbb{C})/H(\mathbb{L})$.

B.4.7 Union

$I_u(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C})/H(\mathbb{M})$ can be decomposed as

$$\begin{aligned} I_u(\mathbb{L}, \mathbb{C}) &= \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{H(\mathbb{M}_J)}{H(\mathbb{M})} I_u(\mathbb{L}, \mathbb{C}) + \frac{\log n - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \log |J|}{H(\mathbb{M})} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J| - \sum_{M \in \mathbb{M}_J} |M| \log |M|}{n \log n - \sum_{M \in \mathbb{M}} |M| \log |M|} I_u(\mathbb{L}_J, \mathbb{C}_J) + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n - \sum_{M \in \mathbb{M}} |M| \log |M|}. \end{aligned}$$

B.4.8 Largest Achievable Upper Bound $\log n$

$I_n(\mathbb{L}, \mathbb{C}) = I(\mathbb{L}, \mathbb{C})/\log n$ can be decomposed as

$$I_n(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J| \log |J|}{n \log n} I_n(\mathbb{L}, \mathbb{C}) + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n}.$$

$\sum_{J \in \mathbb{J}} |J| \log |J| \leq n \log n$ with equality holds only when the join is the top. If a cluster J of the join is a singleton, the weight assigned to the corresponding component is zero. This weighting scheme ignores singletons and it relies the bias term to adjust the score in some situations.

B.4.9 Variation of Information and Its Normalized Variants

The variation of information (VI) can be decomposed as

$$VI(\mathbb{L}, \mathbb{C}) = H(\mathbb{M}) - I(\mathbb{L}, \mathbb{C}) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} (H(\mathbb{M}_J) - I(\mathbb{L}_J, \mathbb{C}_J)) = \sum_{J \in \mathbb{J}} \frac{|J|}{n} VI(\mathbb{L}_J, \mathbb{C}_J).$$

The similarity variant V of the normalized variation of information by $\log n$ can be decomposed as

$$\begin{aligned} V(\mathbb{L}, \mathbb{C}) &= 1 - \frac{VI(\mathbb{L}, \mathbb{C})}{\log n} = 1 - \sum_{J \in \mathbb{J}} \frac{|J|}{n \log n} VI(\mathbb{L}_J, \mathbb{C}_J) = 1 - \sum_{J \in \mathbb{J}} \frac{|J| \log |J|}{n \log n} (1 - V(\mathbb{L}_J, \mathbb{C}_J)) \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J|}{n \log n} V(\mathbb{L}_J, \mathbb{C}_J) + 1 - \sum_{J \in \mathbb{J}} \frac{|J| \log |J|}{n \log n} \\ &= \sum_{J \in \mathbb{J}} \frac{|J| \log |J|}{n \log n} V(\mathbb{L}_J, \mathbb{C}_J) + \frac{n \log n - \sum_{J \in \mathbb{J}} |J| \log |J|}{n \log n}. \end{aligned}$$

Since k is a constant, the similarity variant K of the normalized variation of information by $\log K$ can be decomposed as

$$K(\mathbb{L}, \mathbb{C}) = 1 - \frac{VI(\mathbb{L}, \mathbb{C})}{2 \log k} = 1 - \sum_{J \in \mathbb{J}} \frac{|J|}{n} \frac{VI(\mathbb{L}_J, \mathbb{C}_J)}{2 \log k} = \sum_{J \in \mathbb{J}} \frac{|J|}{n} K(\mathbb{L}_J, \mathbb{C}_J).$$

References

- [1] Reema Al-Kamha and David W. Embley. Grouping search-engine returned citations for person-name queries. In *Proceedings of the 6th annual ACM international workshop on Web information and data management*, WIDM '04, pages 96–103, New York, NY, USA, 2004. ACM.
- [2] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *IR*, 12:461–486, 2009.
- [3] Javier Artiles, Julio Gonzalo, and Satoshi Sekine. The semeval-2007 weps evaluation: establishing a benchmark for the web people search task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 64–69, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.
- [4] Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the unit hypersphere using von mises-fisher distributions. *J. Mach. Learn. Res.*, 6:1345–1382, December 2005.
- [5] Omar Benjelloun, Hector Garcia-Molina, David Menestrina, Qi Su, Steven Euijong Whang, and Jennifer Widom. Swoosh: a generic approach to entity resolution. *The VLDB Journal*, 18(1):255–276, January 2009.

REFERENCES

- [6] M. Blume and P. Kalmar. Methods for evaluating entity disambiguation. In *Proceedings of Workshop on Resources and Evaluation for Identity Matching, Entity Resolution and Entity Management*, Paris, France, 2008. European Language Resources Association.
- [7] Michael H. Coen, M. Hidayath Ansari, and Nathanael Fillmore. Comparing clusterings in space. In *ICML*, pages 231–238, 2010.
- [8] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.
- [9] Stijn Dongen. Performance criteria for graph clustering and markov cluster experiments. Technical report, National Research Institute for Mathematics and Computer Science, Amsterdam, The Netherlands, 2000.
- [10] E. B. Fowlkes and C. L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):553–569, 1983.
- [11] Benjamin C.M. Fung, Ke Wang, and Martin Ester. Hierarchical document clustering using frequent itemsets. In *Proc. SIAM International Conference on Data Mining 2003 (SDM’03)*, 2003.
- [12] Leo A. Goodman and William H. Kruskal. Measures of association for cross classifications. *Journal of the American Statistical Association*, 49(268):732–764, 1954.
- [13] Leo A. Goodman and William H. Kruskal. Measures of association for cross classifications. ii: Further discussion and references. *Journal of the American Statistical Association*, 54(285):123–163, 1959.
- [14] Leo A. Goodman and William H. Kruskal. Measures of association for cross classifications iii: Approximate sampling theory. *Journal of the American Statistical Association*, 58(302):310–364, 1963.

REFERENCES

- [15] George Grätzer. *Lattice Theory: Foundation*. Springer, 1st edition, 2011.
- [16] Isabelle Guyon, Ulrike Von Luxburg, and Robert C. Williamson. Clustering: Science or art. In *NIPS Workshop on Clustering Theory*, 2009.
- [17] Jian Huang, Seyda Ertekin, and C. Lee Giles. Efficient name disambiguation for large-scale databases. In *Proceedings of The 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'06)*, pages 536–544. Springer-Verlag, 2006.
- [18] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.
- [19] S. Jaroszewicz, D.A. Simovici, W.P. Kuo, and L. Ohno-Machado. The goodman-kruskal coefficient and its applications in genetic diagnosis of cancer. *IEEE Transactions on Biomedical Engineering*, 51(7):1095 –1102, july 2004.
- [20] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.
- [21] Tarald O. Kvalseth. Entropy and correlation: Some comments. *Systems, Man and Cybernetics, IEEE Transactions on*, 17(3):517–519, 1987.
- [22] Bjornar Larsen and Chinatsu Aone. Fast and effective text mining using linear-time document clustering. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD'99, pages 16–22, New York, NY, USA, 1999. ACM.
- [23] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, and Junjie Wu. Understanding of internal clustering validation measures. In *ICDM*, pages 911–916, 2010.

REFERENCES

- [24] Ping Luo, Hui Xiong, Guoxing Zhan, Junjie Wu, and Zhongzhi Shi. Information-theoretic distance measures for clustering validation: Generalization and normalization. *IEEE Trans. on Knowl. and Data Eng.*, 21:1249–1262, 2009.
- [25] Xiaoqiang Luo. On coreference resolution performance metrics. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 25–32, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.
- [26] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.
- [27] Marina Meilă. Comparing clusterings: an axiomatic view. In *Proceedings of the 22nd international conference on Machine learning*, pages 577–584, New York, NY, USA, 2005.
- [28] Marina Meilă. Comparing clusterings — an information based distance. *J. Multivar. Anal.*, 98:873–895, May 2007.
- [29] Marina Meilă and David Heckerman. An experimental comparison of model-based clustering methods. *Mach. Learn.*, 42:9–29, 2001.
- [30] David Menestrina, Steven Euijong Whang, and Hector Garcia-Molina. Evaluating entity resolution results. *Proc. VLDB Endow.*, 3:208–219, September 2010.
- [31] Glenn W. Milligan, S. C. Soon, and Lisa M. Sokol. The effect of cluster size, dimensionality, and the number of clusters on recovery of true cluster structure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-5(1):40–47, Jan.
- [32] B.G. Mirkin. *Mathematical Classification and Clustering*. Kluwer Academic Press, Dordrecht, 1996.

- [33] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.
- [34] Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 104–111, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
- [35] Patrick Pantel and Dekang Lin. Efficiently clustering documents with committees. In *Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence*, pages 424–433, London, UK, UK, 2002. Springer-Verlag.
- [36] Altaf Rahman and Vincent Ng. Narrowing the modeling gap: a cluster-ranking approach to coreference resolution. *JAIR*, 40:469–521, 2011.
- [37] William M. Rand. Objective criteria for the evaluation of clustering methods. *JASA*, 66(336):846–850, 1971.
- [38] Roi Reichart and Ari Rappoport. The nvi clustering evaluation measure. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, pages 165–173, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
- [39] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, pages 410–420, 2007.
- [40] Dan A. Simovici and Szymon Jaroszewicz. A metric approach to building decision trees based on goodman-kruskal association index. In Honghua Dai, Ramakrishnan Srikant, and Chengqi Zhang, editors, *PAKDD*, volume 3056 of *Lecture Notes in Computer Science*, pages 181–190. Springer, 2004.

REFERENCES

- [41] Yang Song, Jian Huang, Isaac G. Councill, Jia Li, and C. Lee Giles. Efficient topic-based unsupervised name disambiguation. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, pages 342–351, New York, NY, USA, 2007. ACM.
- [42] Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. In Marko Grobelnik, Dunja Mladenic, and Natasa Milic-Frayling, editors, *KDD-2000 Workshop on Text Mining, August 20*, pages 109–111, Boston, MA, 2000.
- [43] Alexander Strehl and Joydeep Ghosh. Cluster ensembles: a knowledge reuse framework for combining partitionings. In *Eighteenth national conference on Artificial intelligence*, pages 93–98, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence.
- [44] Alexander Strehl and Joydeep Ghosh. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, 3:583–617, 2003.
- [45] Alexander Strehl, Er Strehl, Joydeep Ghosh, and Raymond Mooney. Impact of similarity measures on web-page clustering. In *In Workshop on Artificial Intelligence for Web Search (AAAI 2000)*, pages 58–64. AAAI, 2000.
- [46] Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, MUC6 '95, pages 45–52, Stroudsburg, PA, USA, 1995. Association for Computational Linguistics.
- [47] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.*, 11:2837–2854, 2010.

REFERENCES

- [48] Silke Wagner and Dorothea Wagner. Comparing clusterings — an overview. *Analysis*, 4769(001907):1–19, 2007.
- [49] D.L. Wallace. Comment: A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):569579, 1983.
- [50] Ben Wellner, Andrew McCallum, Fuchun Peng, and Michael Hay. An integrated, conditional model of information extraction and coreference with application to citation matching. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, UAI '04, pages 593–601, Arlington, Virginia, United States, 2004. AUAI Press.
- [51] Junjie Wu, Hua Yuan, Hui Xiong, and Guoqing Chen. Validation of overlapping clustering: A random clustering perspective. *Inf. Sci.*, 180:4353–4369, November 2010.
- [52] Hui Xiong, Junjie Wu, and Jian Chen. K-means clustering versus validation measures: a data distribution perspective. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 779–784, New York, NY, USA, 2006. ACM.
- [53] Yiyu Yao. Information-theoretic measures for knowledge discovery and data mining. In Karmeshu, editor, *Entropy Measures, Maximum Entropy Principle and Emerging Applications*, pages 115–136. Springer, Berlin, 2003.
- [54] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis. Technical Report TR #01–40, Department of Computer Science, University of Minnesota, Minneapolis, MN, 2001.
- [55] Ying Zhao and George Karypis. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Mach. Learn.*, 55:311–331, June 2004.

Index

Clustering

- Bottom $\perp$, 7
- Cartesian product $X \times Y$, 7
- Cluster, 7
- Clustering, 7
- Empty Cluster $\emptyset$, 7
- Join $\mathbb{J}$, 8
- Meet $\mathbb{M}$, 8
- Predicted Clustering $\mathbb{C}$, 7
- Refinement, 8
- Set of All Clusterings Ω , 8
- Top $\top$, 7
- True Clustering $\mathbb{L}$, 7

Conditional Normalization, 9

Existing Measures

- Cluster Counting Measures, 17
 - Extract Matching, 17
 - Overlap Matching, 18
- Information Theoretic Measures, 24
- Pair Counting Measures, 14
- Set Matching Measures, 20

Global Matching, 23

Local Matching, 20

Generative Model

- Aspect, 28
 - Combination Aspect, 28
 - Normalization Aspect, 29
 - Ordered Properties, 30
 - Similarization Aspect, 29
 - Symmetrization Aspect, 29
- Operation, 30
 - Combination Operation, 33
 - Normalization Operation, 35
 - Precondition and Postcondition, 31
 - Similarization Operation, 34
 - Symmetrization Operation, 36

Graph-based Model

- Component, 41
- Components and Join, 42
- Decomposition
 - Conical Decomposition, 44
 - Convex Decomposition, 45

- Join-weighted Decomposition, 45
- Graph, 41
- Induced Clustering, 42
- Induced Graph, 42
- Other Frameworks
 - The Arithmetic Mean Framework, 61
 - The Direct-sum Framework, 62
- Properties of the Framework
 - Bottom, 57
 - Bound Preserving, 61
 - Bounds of the Meet, 60
 - Cluster Completeness, 59
 - Cluster Homogeneity, 59
 - Conditional Normalization, 56
 - Join-weighted Decomposable, 59
 - Merge, 58
 - Normalization, 56
 - Split, 57
 - Subcomponent Consistency, 60
 - Symmetry, 54
 - The Best Clustering, 54
 - The Worst Clusterings, 54
 - Time Complexity, 53
 - Top, 57
- Subcomponent
 - Component and Subcomponents, 47
 - Derivation Graph, 47
 - Split/Merge Subcomponent, 47
 - Subcomponent Consistency, 49
 - Subcomponent Monotonically Decreasing, 48
 - Subcomponent Normalization, 49
 - Subcomponent Pair
 - Component and Subcomponent Pairs, 51
 - Meet-weighted Decomposition, 51
 - Similarity Measure
 - Arithmetic Mean, 52
 - Product, 52
 - Subcomponent Pair, 50
 - Subcomponent Pair and Meet, 50