Web Similarity

Andrew R. Cohen and Paul M.B. Vitányi

Abstract

Normalized web distance (NWD) is a similarity or normalized semantic distance based on the World Wide Web or any other large electronic database, for instance Wikipedia, and a search engine that returns reliable aggregate page counts. For sets of search terms the NWD gives a similarity on a scale from 0 (identical) to 1 (completely different). The NWD approximates the similarity according to all (upper semi)computable properties. We develop the theory and give applications. The derivation of the NWD method is based on Kolmogorov complexity.

Index Terms— Normalized web distance, pattern recognition, data mining, similarity, classification, Kolmogorov complexity,

I. Introduction

Commonly objects are computer files that carry all their properties in themselves. However, there are also objects that are given by name, such as 'red,' 'three,' 'Einstein,' or 'chair.' Such objects acquire their meaning from the common knowledge of mankind. We can give objects either as the object itself or as the name of that object, such as the literal text of the work "Macbeth by Shakespeare" or the name "Macbeth by Shakespeare." We focus on the name case using the background information provided by the World Wide Web, or another data base such as Wikipedia, and a search engine that produces reliable aggregate page counts. The frequencies involved enable us to compute a distance for each set of names. The normalized form of this distance expresses similarity, that is, the search engine discovers the "meaning" names have in common. Insofar as the meaning of names on the data base as discovered by this process approximates the meaning of those names in human society, the above distance expresses the

Andrew Cohen is with the Department of Electrical and Computer Engineering, Drexel University. Address: A.R. Cohen, 3120–40 Market Street, Suite 313, Philadelphia, PA 19104, USA. Email: acohen@coe.drexel.edu

Paul Vitányi is with the national research center for mathematics and computer science in the Netherlands (CWI), and the University of Amsterdam. Address: CWI, Science Park 123, 1098XG Amsterdam, The Netherlands. Email: Paul.Vitanyi@cwi.nl.

common semantics of the names. The term "name" is used here synonymously with "word" "search term" or "query." The normalized distance above is called the normalized web distance (NWD). We apply it in classification.

Example I.1. Although Google gives notoriously unreliable counts it serves well enough for an example. On our scale of similarity, if $NWD(X) = 0$ then the search terms in the set X are identical, and if $NWD(X) = 1$ then the search terms in X are as different as can be. On 19 August 2014 searching for "Shakespeare" gave 124,000,000 hits; searching for "Macbeth" gave 22,400,000 hits; searching for "Hamlet" gave 51,300,000 hits; searching for "Shakespeare Macbeth" gave 7,730,000 hits; searching for "Shakespeare Hamlet" gave 18,500,000 hits; and searching for "Shakespeare Macbeth Hamlet" gave 663,000 hits. The number of web pages returned by Google was estimated by searching for "the" as 25,270,000,000. By (II.3) we have $NWD(\{Shakespeare, Macbeth\}) \approx 0.395$, $NWD(\{Shakespeare, Hamlet\}) \approx 0.306$ and $NWD(\{Shakespeare, Macbeth, Hamlet\}) \approx 0.372$. We conclude that Shakespeare and Hamlet have a lot in common, Shakespeare and Macbeth have a lot in common, and the commonality of Shakespeare, Hamlet, and Macbeth is intermediate between the two. $\diamondsuit$

To develop the theory behind the NWD we consider the information in individual objects. These objects are finite and expressed as finite binary strings. The classic notion of Kolmogorov complexity [8] is an objective measure for the information in a *single* object, and information distance measures the information between a *pair* of objects [1]. There arises the question of the shared information between many objects instead of just a pair of objects.

A. Related Work

The similarity or relative semantics between *pairs* of search terms was defined in [5] and demonstrated in practice by using the World Wide Web as database and Google as search engine. The proposed normalized Google distance (NGD) works for any search engine that gives an aggregate page count for search terms. See for example [2], [7], [21], [20], [3] and the many references to [5] in Google scholar.

In [11] the notion is introduced of the information required to go from any object in a finite multiset (a set where a member can occur more than once) of objects to any other object in the set. Let X denote a finite multiset of n finite binary strings defined by $\{x_1, \ldots, x_n\}$, the constituting

elements ordered length-increasing lexicographic. The *information distance* in X is defined by $E_{\max}(X) = \min\{|p| : U(x_i, p, j) = x_j \text{ for all } x_i, x_j \in X\}$. For instance, with $X = \{x, y\}$ the quantity $E_{\max}(X)$ is the least number of bits in a program to transform x to y and y to x. In [18] the mathematical theory is developed further and the difficulty of normalization is shown.

B. Results

The NWD is a similarity (a common semantics) between all search terms in a *set*. (We use set rather than multiset since a set is more appropriate in the context of search terms.) It can be thought of as a diameter of the set. For sets of cardinality two this diameter reduces to a distance between the two elements of the set. The NWD can be used for the classification of an unseen item into one of several classes (sets of names or phrases). This is simpler and computationally much easier that constructing the classes from the pairwise distances. In the latter solution inevitably information gets lost.

The basic concepts like the web events, web distribution, and web code are given in Section II. We determine the length of a single shortest binary program to compute from any web event of a single member in a set to the web event associated with the whole set (Theorem II.5). The mentioned length is an absolute information distance associated with the set. It is incomputable (Lemma II.4). However, for different sets it can be large while a set has similar members and small when a (different) set has dissimilar members. Therefore we normalize on a scale from 0 to 1 to express the information distance or similarity between members of the set. We approximate the incomputable normalized version with the computable NWD (Definition II.6). In Section III we present properties of the NWD such as the range from 0 to 1 (Lemma III.1), whether and how it changes under adding members (Lemma III.2), and that it does not satisfy the triangle inequality and hence is not metric (Lemma III.5). Theorem III.7 and Corollary III.8 show that the NWD approximates the common similarity of the queries in a set of search terms (that is, a common semantics). We subsequently apply the NWD to various data sets based on search results from Amazon, Wikipedia and the National Center for Biotechnology Information (NCBI) website from the U.S. National Institutes of Health in Section IV. We treat strings and self-delimiting strings in Appendix A, computability notions in Appendix B, Kolmogorov complexity in Appendix C, and metric of sets in Appendix D. The proofs are deferred to Appendix E.

II. Web Distribution and Web Code

We give a derivation that holds for idealized search engines that return reliable aggregate page counts from their data bases (here called the web consisting of web pages). Subsequently we apply the idealized theory to real problems using real search engines on real data bases.

A. Web Event

The set of singleton *search terms* is denoted by $\mathcal{S}$, a *set of search terms* is $X = \{x_1, \ldots, x_n\}$ with $x_i \in \mathcal{S}$ for $1 \leq i \leq n < \infty$, and $\mathcal{X}$ denotes the set of such X. Let the set of web pages indexed (possible of being returned) by the search engine be Ω.

Definition II.1. We define the *web event* $e(X) \subseteq \Omega$ by the set of web pages returned by the search engine doing a search for X such that each web page in the set contains occurrences of all elements from X.

If $x, y \in S$ and $e(x) = e(y)$ then $x \sim y$ and the equivalence class $[x] = \{y \in S : y \sim x\}$. Unless otherwise stated, we consider all singleton search terms that define the same web event as the same term. Hence we deal actually with equivalence classes $[x]$ rather than x. However, for ease of notation we write x in the sequel and consider this to mean $[x]$.

If $X = \{x_1, \ldots, x_n\}$, then $e(X) = e(x_1) \bigcap \cdots \bigcap e(x_n)$ and the *frequency* $f(X) = |e(X)|$. The web event $e(X)$ embodies all direct context in which all elements from X simultaneously occur in these web pages. Therefore web events capture in the outlined sense all background knowledge about this combination of search terms on the web.

B. The Web Code

It is natural to consider code words for events. We base those code words on the probability of the event. Consider the set

$$T_{w,s} = \{(w, s) : w \in \Omega,\ s \in S, s \text{ occurs in } w\}.$$

Then $\alpha = \sum_{w \in \Omega,\ s \in S} |T_{w,s}|/|\Omega|$ is the average number of search terms per web page in Ω. Define the *probability* $g(X)$ of X as $g(X) = f(X)/N$ with $N = \alpha|\Omega|$. This probability may change over time, but let us imagine that the probability holds in the sense of an instantaneous snapshot.

A probability mass function on a known set allows us to define the associated prefix-code word length (information content) equal to unique decodable code word length [9], [13]. Such a prefix code is a code such that no code word is a proper prefix of any other code word. By the ubiquitous Kraft inequality [9], if $l_1, l_2, \ldots$ is a sequence of positive integers satisfying

$$\sum_i 2^{-l_i} \leq 1, \tag{II.1}$$

then there is a set of prefix-code words of length $l_1, l_2, \ldots$. Conversely, if there is a set of prefix-code words of length $l_1, l_2, \ldots$ then these lengths satisfy the above displayed equation. By the fact that the probabilities of a discrete set sum to at most 1, every web event $e(X)$ having probability $g(X)$ can be encoded in a prefix-code word.

Definition II.2. The *length* $G(X)$ of the *web code word* for $X \in \mathcal{X}$ is

$$G(X) = \log 1/g(X), \tag{II.2}$$

or ∞ for $g(X) = 0$. The case $|X| = 1$ gives the length of the web code word for singleton search terms. The logarithms are throughout base 2.

The web code is a prefix code. The code word associated with X and therefore with the web event $e(X)$ can be viewed as a compressed version of the set of web pages constituting $e(X)$. That is, the search engine compresses the set of web pages that contain all elements from X into a code word of length $G(X)$.

Definition II.3. Let $p \in \{0,1\}^*$ and $X \in \mathcal{X} \setminus S$. The *information* $EG_{\max}(X)$ *to compute event* $e(X)$ *from event* $e(x)$ *for any* $x \in X$ is defined by $EG_{\max}(X) = \min_p\{|p| : \text{for all } x \in X \text{ we have } U(e(x), p) = e(X)\}$.

In this way $EG_{\max}(X)$ corresponds to the length of a single shortest self-delimiting program to compute output $e(X)$ from an input $e(x)$ for all $x \in X$. We use the notion of prefix Kolmogorov complexity K as in Appendix C.

Lemma II.4. *The function* $EG_{\max}$ *is upper semicomputable but not computable.*

Theorem II.5. $EG_{\max}(X) = \max_{x \in X}\{K(e(X)|e(x))\}$ *up to an additive logarithmic term*

$O(\log \max_{x\in X}\{K(e(X)|e(x))\})$ *which we ignore in the sequel.*

To obtain the NWD we must normalize $EG_{\max}$. Let us give some intuition first. Suppose $X, Y \in \mathcal{X} \setminus S$. If the web events $e(x)$'s are more or less the same for all $x \in X$ then we consider the members of X very similar to each other. If the web events $e(y)$'s are very different for different $y \in Y$ then we consider the members of Y to be very different from one another. Yet for certain X and Y depending on the cardinalities and the size of the web events of the members we can have $EG_{\max}(X) = EG_{\max}(Y)$. That is to say, the similarity is dependent on size. Therefore, to express similarity of the elements in a set X we need to normalize $EG_{\max}(X)$ using the cardinality of X and the events of its members. Expressing the normalized values on a scale of 0 to 1 allows us to express the degree in which all elements of a set are alike. Then we can compare truly different sets.

Use the symmetry of information law (A.1) to rewrite $EG_{\max}(X)$ according to Theorem II.5 as $K(e(X)) - \min_{x\in X}\{K(e(x))\}$ up to a logarithmic additive term which we ignore. Since $G(X)$ is computable prefix code for $e(X)$, while $K(e(X))$ is the shortest computable prefix code for $e(X)$, it follows that $K(e(X)) \leq G(X)$. Similarly $K(e(x)) \leq G(x)$ for $x \in X$. The search engine G returns frequency $f(X)$ on query X (respectively frequency $f(x)$ on query x). These frequencies are readily converted into $G(X)$ (respectively $G(x)$) using (II.2). Replace $K(e(X))$ by $G(X)$ and $\min_{x\in X}\{K(e(x))\}$ by $\min_{x\in X}\{G(x)\}$ in $EG_{\max}(X)$. Subsequently use as normalizing term $\max_{x\in X}\{G(x)\}(|X| - 1)$. This yields the following.

Definition II.6. The *normalized web distance* (NWD) of $X \in \mathcal{X}$ with $G(X) < \infty$ (equivalently $f(X) > 0$)) is

$$\begin{aligned} NWD(X) &= \frac{G(X) - \min_{x\in X}\{G(x)\}}{\max_{x\in X}\{G(x)\}(|X| - 1)} \\ &= \frac{\max_{x\in X}\{\log f(x)\} - \log f(X)}{(\log N - \min_{x\in X}\{\log f(x)\})(|X| - 1)}, \end{aligned} \tag{II.3}$$

otherwise $NWD(X)$ is undefined.

The second equality in (II.3), expressing the NWD in terms of frequencies, is seen as follows. We use (II.2). The numerator is rewritten by $G(X) = \log 1/g(X) = \log(N/f(X)) = \log N - \log f(X)$ and $\min_{x\in X}\{G(x)\} = \min_{x\in X}\{\log 1/g(x)\} = \log N - \max_{x\in X}\{\log f(x)\}$.

The denominator is rewritten as $\max_{x\in X}\{G(x)\}(|X|-1) = \max_{x\in X}\{\log 1/g(x)\}(|X|-1) = (\log N - \min_{x\in X}\{\log f(x)\})(|X|-1)$.

Remark II.7. By assumption $f(X) > 0$ which, since it has integer values, means $f(X) \geq 1$. The case $f(X) = 0$ means that there is an $x \in X$ such that $e(x) \bigcap e(X \setminus \{x\}) = \varnothing$. That is, query x is independent of the set of queries $X \setminus \{x\}$, that is, x has nothing in common with $X \setminus \{x\}$ since there is no common web page. Hence the NWD is undefined. The other extreme is that $e(x) = e(y)$ ($x \sim y$) for all $x, y \in X$. In this case the $NWD(X) = 0$. $\diamondsuit$

III. Theory

Let $X = \{x, y\} \in \mathcal{X}$. We can rewrite [5, Section 3.4 formula (6)] for the NGD distance between x and y as $NWD(X)$ up to a constant. Hence the NGD and NWD coincide for pairs up to a constant. For arbitrary sets the following holds.

Lemma III.1. *Let $X \in \mathcal{X} \setminus S$. Then $NWD(X) \in [0, 1]$.*

We determine bounds on how the NWD may change under addition of members to its argument. These bounds are necessary loose since the added members may be similar to existing ones or very different. In Lemma III.2 below we shall distinguish two cases for the relation between the minimum frequencies of members of X and Y with $X \subset Y$ and the overall frequencies of X and Y. In the first case

$$\frac{f(y_1)f(X)}{f(x_1)f(Y)} \geq \left(\frac{f(x_0)}{f(y_0)}\right)^{(|X|-1)NWD(X)}, \tag{III.1}$$

where $x_0 = \arg\min_{x\in X}\{\log f(x)\}$, $y_0 = \arg\min_{y\in Y}\{\log f(y)\}$, $x_1 = \arg\max_{x\in X}\{\log f(x)\}$, and $y_1 = \arg\max_{y\in Y}\{\log f(y)\}$.

We give an example. Let $|X| = 5$, $f(x_0) = 1,100,000$, $f(y_0) = 1,000,000$, $f(x_1) = f(y_1) = 2,000,000$, $f(X) = 500$, $f(Y) = 100$, and $NWD(X) = 0.5$. The righthand side of the inequality (III.1) is $1.1^2 = 1.21$ while the lefthand side is 5. In the second case inequality (III.1) does not hold, that is, it holds with the $\geq$ sign replaced by the $<$ sign. We give an example. Let $|X| = 5$, $f(x_0) = 1,100,000$, $f(y_0) = 1,000,000$, $f(x_1) = f(y_1) = 2,000,000$, $f(X) = 110$, $f(Y) = 100$, and $NWD(X) = 0.5$. The righthand side of the inequality (III.1) with $\geq$ replaced by $<$ is $1.1^2 = 1.21$ while the lefthand side is 1.1.

Lemma III.2. *Let* $X, Z \subseteq Y$, $X, Y, Z \in \mathcal{X} \setminus S$, *and* $\min_{z \in Z}\{f(z)\} = \min_{y \in Y}\{f(y)\}$.

(i) *If* $f(y) \geq \min_{x \in X}\{f(x)\}$ *for all* $y \in Y$ *then* $(|X| - 1)NWD(X) \leq (|Y| - 1)NWD(Y)$.
(ii) *Let* $f(y) < \min_{x \in X}\{f(x)\}$ *for some* $y \in Y$. *If* (III.1) *holds then* $(|X| - 1)NWD(X) \leq (|Y| - 1)NWD(Y)$. *If* (III.1) *does not hold then* $(|X| - 1)NWD(X) > (|Y| - 1)NWD(Y) \geq (|Z| - 1)NWD(Z)$.

Example III.3. Consider the Shakespeare–Macbeth–Hamlet Example I.1. Let $X = \{Shakespeare, Macbeth\}$, $Y = \{Shakespeare, Macbeth, Hamlet\}$, and $Z = \{Shakespeare, Hamlet\}$. Then inequality (III.1) for X versus Y gives $(124,000,000 \times 7,730,000/(124,000,000 \times 663,000) \geq (22,400,000/22,400,000)^{0.395}$ (that is $11.659 \geq 1$), and for Z versus Y gives $18,500,000/663,000 \geq (51,300,000/22,400,000)^{0.306}$ (that is $27.903 \geq 1.289$). In the first case Lemma III.2 item (i) is applicable since the frequency minima of X and Y are the same. (In this case inequality (III.1) is not needed.) Therefore $NWD(X)(|X| - 1)/(|Y| - 1) \leq NWD(Y)$ which works out as $0.395/2 \leq 0.372$. In the second case Lemma III.2 item (ii) is applicable since the frequency minima of Z and Y are not the same. Since inequality (III.1) holds this gives $NWD(Z)(|Z| - 1)/(|Y| - 1) \leq NWD(Y)$ which works out as $0.306/2 \leq 0.372$. $\diamondsuit$

Remark III.4. To interpret Lemma III.2 we give the following intuition. Under addition of a member to a set there are two opposing tendencies on the NWD concerned. First, the range of the NWD stays fixed at a unit and (II.3) shows that addition of a member tends to decrease the NWD, that is, it moves closer to 0. Second, the common similarity of queries in a given set as measured by the NWD is based on the number of properties all members of a set have in common. By adding a member to the set clearly the number of common properties does not increase and generally decreases. This diminishing tends to cause the NWD to increase—move closer to 1. The first effect is visible when $(|X| - 1)NWD(X) > (|Y| - 1)NWD(Y)$, which happens in the case of Lemma III.2 item (ii) for the case when the frequencies do not satisfy (A.2). The second effect is visible when $(|X| - 1)NWD(X) \leq (|Y| - 1)NWD(Y)$, which happens in Lemma III.2 item (i), and item (ii) with the frequencies satisfying (A.2). (Note that to keep $NWD(X) \in [0, 1]$ for all X we have the factor $(|X| - 1)$ in the denominator of $NWD(X)$. Without this factor the resulting function of X has range $[0, |X| - 1]$ and in the inequalities in this remark and in the NWD formula (II.3) and all the previous theory properties

the factors $|X|-1$ and $|Y|-1$ are replaced by 1.) $\diamondsuit$

For every set X we have that the $NWD(X)$ is invariant under permutation of X: it is symmetric. The NWD is also positive definite as in Appendix D (where equal members should be interpreted as saying that the set has only one member). However the NWD does not satisfy the triangle inequality and hence is not a metric. This is natural for a common similarity or semantics: The members of a set XY can be less similar (have greater NWD) then the similarity of the members of XZ plus the similarity of the members of ZY for some set Z.

Lemma III.5. *The NWD violates the triangle inequality.*

It remains to formally prove that the NWD expresses in the similarity of the search terms in the set. We define the notion of a distance on these sets using the web as side-information. We consider only distances that are upper semicomputable, that is, the distance can be computably approximated from above (Appendix B). A priori we allow asymmetric distances, but we exclude degenerate distances such as $d(X) = 1/2$ for all $X \in \mathcal{X}$ containing a fixed element x. That is, for every d we want only finitely many sets $X \ni x$ such that $d(X) \leq d$. Exactly how fast we want the number of sets we admit to go to ∞ is not important; it is only a matter of scaling.

Definition III.6. A *web distance function* (quantifying the common properties or common features) $d : \mathcal{X} \to \mathcal{R}^+$ is *admissible* if $d(X)$ is (i) a nonnegative total real function and is 0 iff $X \in S$; (ii) it is upper semicomputable from the $e(x)$'s with $x \in X$ and $e(X)$; and (iii) it satisfies the density requirement: for every $x \in S$

$$\sum_{X \ni x,\, |X| \geq 2} 2^{-d(X)} \leq 1.$$

We give the gist of what we are about to prove. Let $X = \{x_1, x_2, \ldots, x_n\}$. A feature of a query is a property of the web event of that query. For example, the frequency in the web event of web pages containing an occurrence of the word "red." We can compute this frequency for each $e(x_i)$ ($1 \leq i \leq n$). The minimum of those frequencies is the maximum of the number of web pages containing the word "red" which surely is contained in each web event $e(x_1), \ldots, e(x_n)$. One can identify this maximum with the inverse of a distance in X. There are many such distances in X. The shorter a web distance is, the more dominant is the feature it represents. We show

that the minimum admissible distance is $EG_{\max}(X)$. It is the least admissible web distance and represents the shortest of all admissible web distances in members of X. Hence the closer the numerator of $NWD(X)$ is to $EG_{\max}(X)$ the better it represents the dominant feature all members of X have in common.

Theorem III.7. *Let $X \in \mathcal{X}$. The function $G(X) - \min_{x \in X}\{G(x)\}$ is a computable upper bound on $EG_{\max}(X)$. The closer it is to $EG_{\max}(X)$, the better it approximates the shortest admissible distance in X. The normalized form of $EG_{\max}(X)$ is $NWD(X)$.*

The normalized least admissible distance in a set is the least admissible distance between its members which we call the common admissible similarity. Therefore we have:

Corollary III.8. The function $NWD(X)$ is the common admissible similarity among all search terms in X. This admissible similarity can be viewed as semantics that all search terms in X have in common.

IV. Applications

The application of the approach presented here requires the ability to query a database for the number of occurrences and co-occurrences of the elements in the set that we wish to analyze. One challenge is to find a database that has sufficient breadth as to contain a meaningful numbers of co-occurrences for related terms. As discussed previously, an example of one such database is the World Wide Web, with the page counts returned by Google search queries used as an estimate of co-occurrence frequency. There are two issues with using Google search page counts. The first issue is that Google limits the number of programmatic searches in a single day to a maximum of 100 queries, and charges for queries in excess of 100 at a rate of up to $50 per thousand. The second issue with using Google web search page counts is that the numbers are not exact, but are generated using an approximate algorithm that Google has not disclosed. For the questions considered previously [5] we found that these approximate measures were sufficient at that time to generate useful answers, especially in the absence of any a priori domain knowledge. It is possible to implement internet based searches without using search engine API's, and therefore not subject to daily limit. This can be accomplished by parsing the HTML returned by the search engine directly. The issue with google page counts in this study being approximate counts based

on a non-public algorithm was more concerning as changes in the approximation algorithm can influence page count results in a way that may not reflect true changes to the underlying distributions. Since any internet search that returns a results count can be used in computing the NWD, we adopt the approach of using web sites that return exact rather than approximate page counts for a given query.

Here we describe a comparison of the NWD using the set formulation based on web-site search result counts with the pairwise NWD formulation. The examples are based on search results from Amazon, Wikipedia and the National Center for Biotechnology Information (NCBI) website from the U.S. National Institutes of Health. The NCBI website exposes all of the NIH databases searchable from a single web portal. We consider example classification questions that involve partitioning a set of words into underlying categories. For the NCBI applications we compare various diseases using the loci identified by large genome wide association studies (GWAS). For the NWD set classification, we determine whether to assign element x to class A or class B by computing $NWD(Ax) - NWD(A)$ and $NWD(Bx) - NWD(B)$ and assigning element x to whichever class achieves the minimum.

For the pairwise formulation, we use the gap spectral clustering unsupervised approach developed in [4]. Gap spectral clustering uses the gap statistic as first proposed in [17] to estimate the number of clusters in a data set from an arbitrary clustering algorithm. In [4], it was shown that the gap statistic in conjunction with a spectral clustering [15] of the distance matrix obtained from pairwise NWD measurements is an estimate of randomness deficiency for clustering models. Randomness deficiency is a measure of the meaningful information that a model, here a clustering algorithm, captures in a particular set of data [12]. The approach is to select the number of clusters that minimizes the randomness deficiency as approximated by the gap value. In practice, this is achieved by picking the first number of clusters where the gap value achieves a maximum as described in [4].

The gap value is computed by comparing the intra-cluster dispersions of the pairwise NWD distance matrix to that of uniformly distributed randomly generated data on the same range. For each value of k, the number of clusters in the data, we apply a spectral clustering algorithm to partition the data, assigning each element in the data to one of k clusters. Next, we compute

D_r, the sum of the distances between elements in each cluster C_r,

$$D_r = \sum_{i,j \in C_r} d_{i,j}.$$

The average intra-cluster dispersion is calculated,

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r,$$

where n_r is the number of points in cluster C_r. The gap statistic is then computed as the difference between the averages of the intra-cluster distances of our data and the intra-cluster distances of B randomly generated uniformly distributed data sets of the same dimension as our data,

$$Gap(k) = \frac{1}{B} \sum_{b=1}^{B} \log(W_{kb}) - \log(W_k),$$

where W_{kb} is the average intra-cluster dispersion obtained by running our clustering algorithm on each of the B randomly generated uniformly-distributed datasets. Following [4] we set B to 100. We compute the standard deviation of the gap value s_k from σ_k, the standard deviation of the B uniformly distributed randomly generated data, adjusted to account for simulation error, as

$$s_k = \sigma_k \sqrt{1 + {}^1/_B}.$$

Finally, we choose the smallest value of k for which

$$Gap(k) \geq Gap(k+1) - s_{k+1}.$$

We now describe results from a number of sample applications. For all of these applications, we use a single implementation based on co-occurrence counts. For each search engine that we used, including Amazon, Wikipedia and NCBI a custom MATLAB script was developed to parse the search count results. We used the page counts returned using the built in search from each website for the frequencies, and following the approach in [5] choose N as the frequency for the search term 'the'. The results described were not sensitive to the choice of search term used to establish N, for example identical classification results were obtained using the counts returned by the search term 'N' as the normalizing factor. Following each classification result below, we

include in parenthesis the 95% confidence interval for the result, computed as described in [19]

The first three classification questions we considered used the wikipedia search engine. These questions include classifying colors vs. animals, classifying colors vs. shapes and classifying presidential candidates by political party for the US 2008 U.S. presidential election. For colors vs animals and shapes, gap spectral clustering found two groups in the data and classified all of the elements 100% correctly. The NWD set formulation classified the terms perfectly (0.82,1.0). For the presidential candidate classification by party, the pairwise NWD formulation performed poorly, classifying 58% correctly (0.32,0.8), while the multiset formulation obtained 100% correct classification (0.76,1.0). Table I shows the data used for each question, together with the pairwise and set accuracy, the number of groups obtained by gap spectral clustering and the total number of website queries required for each method.

search engine: wikipedia	Multisets Correct	Pairwise Correct	Groups found by gap spectral	Number of queries (pairwise)	Number of queries (multisets)
{red, orange, yellow, green, blue, indigo} {lion, tiger, bear, monkey, zebra, elephant, aardvark, lamb, fox, ape, dog}	100%	100%	2	136	394
{red, orange, yellow, green, blue, indigo, violet, purple, cyan, white} {square,circle,rectangle,ellipse,triangle, rhombus}	100%	100%	2	105	342
{Barack Obama, Hillary Clinton, John Edwards, Joe Biden, Chris Dodd, Mike Gravel} {John McCain, Mitt Romney, Mike Huckabee, Ron Paul, Fred Thompson, Alan Keyes}	100%	58%	2	66	198

TABLE I
CLASSIFICATION RESULTS USING WIKIPEDIA.

The next classification question considered used page counts returned by the Amazon website search engine to classify book titles by author. Table II summarizes the sets of novels associated with each author, and the classification results for each author as a confusion matrix. The Multiset NWD (top) misclassified one of the Tolstoy novels ('War and Peace') to Stephen King, but

Shakespeare = {Macbeth, The Tempest, Othello, King Lear, Hamlet, The Merchant of Venice, A Midsummer Nights Dream, Much Ado About Nothing, Taming of the Shrew, Twelfth Night}

King = {Carrie, Salems Lot, The Shining, The Stand, The Dead Zone, Firestarter, Cujo}

Twain = {Adventures of Huckleberry Finn, A Connecticut Yankee in King Arthurs Court, Life on the Mississippi, Puddnhead Wilson}

Hemingway = {The Old Man and The Sea, The Sun Also Rises, For Whom the Bell Tolls, A Farewell To Arms}

Tolstoy = {Anna Karenina, War and Peace, The Death of Ivan Ilyich}

Multiset NWD

		True Class				
		Shakespeare	King	Twain	Hemingway	Tolstoy
Predicted Class	Shakespeare	10	0	0	0	0
	King	0	7	0	0	1
	Twain	0	0	4	0	0
	Hemingway	0	0	0	4	0
	Tolstoy	0	0	0	0	2

Correct: 96%

Pairwise NWD

		True Class				
		Shakespeare	King	Twain	Hemingway	Tolstoy
Predicted Class	Shakespeare	10	0	0	1	1
	King	0	6	0	0	0
	Twain	0	0	4	0	0
	Hemingway	0	1	0	3	3
	Tolstoy	0	0	0	0	0

Correct: 79%

TABLE II
CLASSIFYING NOVELS BY AUTHOR USING AMAZON

correctly classified all other novels correctly, 96% accurate (0.83,0.99). The pairwise NWD performed significantly more poorly, achieving only 79% accuracy (0.6,0.9).

The final application considered is to quantify similarities among diseases based on the results of genome wide association studies (GWAS). These studies scan the genomes from a large population of individuals to identify genetic variations occurring at fixed locations, or loci that can be associated with the given disease. Here we use the the NIH NCBI database to search for similarities among diseases, comparing loci identified by recent GWAS results for each disease. The diseases included Alzheimers [22], Parkinsons [27], Amyotrophic lateral sclerosis (ALS) [23], Schizophrenia [28], Leukemia [24], Obesity [26], and Neuroblastoma [25]. The top of Table III lists the loci used for each disease. The middle panel of Table III shows at each

location (i, j) of the distance matrix the NWD computed for the combined counts for the loci of disease i concatenated with disease j. The diagonal elements (i, i) show the NWD for the loci of disease i. The bottom panel of Table III shows the NWD for each element with the diagonal subtracted, $(i, j) - (i, i)$. This is equivalent to the $NWD(Ax) - NWD(A)$ value used in the previous classification problems. The two minimum values in the bottom panel, showing the relationships between Parkinsons and Obesity, as well as between Schizophrenia and Leukemia were surprising. The hypothesis was that neurological disorders such as Parkinsons, ALS and Alzheimers, would be more similar to each other. After these findings we found that there actually have been recent findings of strong relationships between both Schizophrenia and Leukemia [29] as well as between Parkinsons and Obesity [30], relationships that have also been identified by clinical evidence not relating to GWAS approaches.

Schizophrenia = {'rs1702294', 'rs11191419', 'rs2007044', 'rs4129585', 'rs35518360'}

Leukemia = {'rs17483466', 'rs13397985', 'rs757978', 'rs2456449', 'rs735665', 'rs783540', 'rs305061', 'rs391525', 'rs1036935', 'rs11083846'}

Alzheimers={'rs4420638', 'rs7561528', 'rs17817600', 'rs3748140', 'rs12808148', 'rs6856768', 'rs11738335', 'rs1357692'};

Obesity={'rs10926984', 'rs12145833', 'rs2783963', 'rs11127485', 'rs17150703', 'rs13278851'};

Neuroblastoma = {'rs6939340', 'rs4712653', 'rs9295536', 'rs3790171', 'rs7272481'};

Parkinsons={'rs356219', 'rs10847864', 'rs2942168', 'rs11724635'}

ALS = {'rs2303565', 'rs1344642', 'rs2814707', 'rs3849942', 'rs2453556', 'rs1971791', 'rs8056742'};

NWD(i,j)

	Alzheimers	Parkinsons	ALS	Schizophrenia	Leukemia	Obesity	Neuroblastoma
Alzheimers	1.29E-02	2.43E-02	1.38E-02	1.55E-02	1.23E-02	1.49E-02	1.61E-02
Parkinsons	2.43E-02	1.80E-02	1.83E-02	1.58E-02	1.68E-02	1.53E-02	2.23E-02
ALS	1.38E-02	1.83E-02	9.76E-03	1.19E-02	1.46E-02	9.96E-03	1.75E-02
Schizophrenia	1.55E-02	1.58E-02	1.19E-02	1.38E-02	1.13E-02	1.60E-02	1.93E-02
Leukemia	1.23E-02	1.68E-02	1.46E-02	1.13E-02	7.54E-03	1.15E-02	1.61E-02
Obesity	1.49E-02	1.53E-02	9.96E-03	1.60E-02	1.15E-02	1.23E-02	1.51E-02
Neuroblastoma	1.61E-02	2.23E-02	1.75E-02	1.93E-02	1.61E-02	1.51E-02	1.51E-02

NWD(i,j)-NWD(i,i)

	Alzheimers	Parkinsons	ALS	Schizophrenia	Leukemia	Obesity	Neuroblastoma
Alzheimers	0	1.14E-02	9.20E-04	2.64E-03	-6.08E-04	1.98E-03	3.22E-03
Parkinsons	6.26E-03	0	2.77E-04	-2.28E-03	-1.28E-03	-2.76E-03	4.26E-03
ALS	4.04E-03	8.57E-03	0	2.11E-03	4.87E-03	2.00E-04	7.75E-03
Schizophrenia	1.75E-03	2.01E-03	-1.90E-03	0	-2.44E-03	2.20E-03	5.56E-03
Leukemia	4.73E-03	9.23E-03	7.09E-03	3.78E-03	0	3.99E-03	8.53E-03
Obesity	2.57E-03	3.01E-03	-2.33E-03	3.69E-03	-7.58E-04	0	2.78E-03
Neuroblastoma	1.01E-03	7.23E-03	2.43E-03	4.25E-03	9.92E-04	-1.04E-05	0

TABLE III
GWAS LOCI USED AS INPUT TO NWD FOR QUANTIFYING DISEASE SIMILARITY USING THE NIH NCBI WEBSITE.

V. CONCLUSION

Consider queries to a search engine using a data base divided in chunks called web pages. On each query the search engine returns a set of web pages. We propose a method, the normalized web distance (NWD) for sets of queries that quantifies in a single number between 0 and 1 the way in which the queries in the set are similar: 0 means all queries in the set are the same (the set has cardinality one) and 1 means all queries in the set are maximally dissimilar to each other.

The similarity among queries uses the frequency counts of web pages returned for each query and the set of queries. The method can be applied using any big data base and a search engine that returns reliable aggregate page counts. Since this method uses names for object, and not the objects themselves, we can view the common similarity of the names as a common semantics between those names (words or phrases). The common similarity between a finite nonempty set of queries can be viewed as a distance or diameter of this set. We show that this distance ranges in between 0 and 1, how it changes under adding members to the set, that it does not satisfy the triangle property, and that the NWD formally and provably expresses common similarity (common semantics).

To test the efficacy of the new method for classification we experimented with small data sets of queries based on search results from Wikipedia, Amazon, and the National Center for Biotechnology Information (NCBI) website from the U.S. National Institutes of Health. In particular we compared classification using pairwise NWDs with classification using set NWD. The last mentioned performed consistently equal or better, sometimes much better.

APPENDIX

A. Strings and the Self-Delimiting Property

We write *string* to mean a finite binary string, and ϵ denotes the empty string. (If the string is over a larger finite alphabet we recode it into binary.) The *length* of a string x (the number of bits in it) is denoted by $|x|$. Thus, $|\epsilon| = 0$. The *self-delimiting code* for x of length n is $\bar{x} = 1^{|x|}0x$ of length $2n + 1$, or even shorter $x' = 1^{\bar{x}}0x$ of length $n + 2\log n + 1$ (see [12] for still shorter self-delimiting codes). Self-delimiting code words encode where they end. The advantage is that if many strings of varying lengths are encoded self-delimitingly using the same code, then their concatenation can be parsed in their constituent code words in one pass going from left to right. Self delimiting codes are computable prefix codes. A *prefix code* has the property that no code word is a proper prefix of any other code word. The code-word set is called *prefix-free*.

We identify strings with natural numbers by associating each string with its index in the length-increasing lexicographic ordering according to the scheme $(\epsilon, 0), (0, 1), (1, 2), (00, 3), (01, 4), (10, 5), (11, 6), \ldots$. In this way the Kolmogorov complexity can be about finite binary strings or natural numbers.

B. Computability Notions

A pair of integers such as (p, q) can be interpreted as the rational p/q. We assume the notion of a function with rational arguments and values. A function $f(x)$ with x rational is *upper semicomputable* if it is defined by a rational-valued total computable function $\phi(x, k)$ with x a rational number and k a nonnegative integer such that $\phi(x, k+1) \leq \phi(x, k)$ for every k and $\lim_{k\to\infty} \phi(x, k) = f(x)$. This means that f can be computed from above (see [12], p. 35). A function f is *lower semicomputable* if $-f$ is semicomputable from above. If a function is both upper semicomputable and lower semicomputable then it is *computable*.

C. Kolmogorov Complexity

The Kolmogorov complexity is the information in a single finite object [8]. Informally, the Kolmogorov complexity of a finite binary string is the length of the shortest string from which the original can be lossless reconstructed by an effective general-purpose computer such as a particular universal Turing machine. Hence it constitutes a lower bound on how far a lossless compression program can compress. For technical reasons we choose Turing machines with a separate read-only input tape that is scanned from left to right without backing up, a separate work tape on which the computation takes place, an auxiliary tape inscribed with the *auxiliary* information, and a separate output tape. All tapes are divided into squares and are semi-infinite. Initially, the input tape contains a semi-infinite binary string with one bit per square starting at the leftmost square, and all heads scan the leftmost squares on their tapes. Upon halting, the initial segment p of the input that has been scanned is called the input program and the contents of the output tape is called the output. By construction, the set of halting programs is prefix free (Appendix A), and this type of Turing machine is called a *prefix Turing machine*. A standard enumeration of prefix Turing machines $T_1, T_2, \ldots$ contains a universal machine U such that $U(i, p, y) = T_i(p, y)$ for all indexes i, programs p, and auxiliary strings y. (Such universal machines are called "optimal" in contrast with universal machines like U' with $U'(i, pp, y) = T_i(p, y)$ for all i, p, y, and $U'(i, q, y) = 1$ for $q \neq pp$ for some p.) We call U the *reference universal prefix Turing machine*. This leads to the definition of prefix Kolmogorov complexity.

Formally, the *conditional prefix Kolmogorov complexity* $K(x|y)$ is the length of the shortest input z such that the reference universal prefix Turing machine U on input z with auxiliary information y outputs x. The *unconditional Kolmogorov complexity* $K(x)$ is defined by $K(x|\epsilon)$

where ϵ is the empty string. In these definitions both x and y can consist of strings into which finite sets of finite binary strings are encoded. Theory and applications are given in the textbook [12].

For a finite set of strings we assume that the strings are length-increasing lexicographic ordered. This allows us to assign a unique Kolmogorov complexity to a set. The conditional prefix Kolmogorov complexity $K(X|x)$ of a set X given an element x is the length of a shortest program p for the reference universal Turing machine that with input x outputs the set X. The prefix Kolmogorov complexity $K(X)$ of a set X is defined by $K(X|\epsilon)$. One can also put set in the conditional such as $K(x|X)$ or $K(X|Y)$. We will use the straightforward laws $K(\cdot|X,x) = K(\cdot|X)$ and $K(X|x) = K(X'|x)$ up to an additive constant term, for $x \in X$ and X' equals the set X with the element x deleted.

We use the following notions from the theory of Kolmogorov complexity. The *symmetry of information* property [6] for strings x, y is

$$K(x,y) = K(x) + K(y|x) = K(y) + K(x|y), \tag{A.1}$$

with equalities up to an additive term $O(\log(K(x,y)))$.

D. Metricity

A *distance function* d on $\mathcal{X}$ is defined by $d : \mathcal{X} \to \mathcal{R}^+$ where $\mathcal{R}^+$ is the set of nonnegative real numbers. If $X, Y, Z \in \mathcal{X}$, then $Z = XY$ if Z is the set consisting of the elements of the sets X and Y ordered length-increasing lexicographic. A distance function d is a *metric* if

1) *Positive definiteness*: $d(X) = 0$ if all elements of X are equal and $d(X) > 0$ otherwise. (For sets equality of all members means $|X| = 1$.)
2) *Symmetry*: $d(X)$ is invariant under all permutations of X.
3) *Triangle inequality*: $d(XY) \leq d(XZ) + d(ZY)$.

E. Proofs

Proof: of Lemma II.4.

We can run all programs dovetailed fashion and at each time instant select a shortest program that with inputs $e(x)$ for all $x \in X$ has terminated with the same output $e(X)$.

The lengths of these shortest programs gets shorter and shorter, and in for growing time eventually reaches $EG_{\max}(X)$ (but we do not know the time for which it does). Therefore $EG_{\max}(X)$ is upper semicomputable. It is not computable since for $X = \{x, y\}$ we have $EG_{\max}(X) = \max\{K(e(x)|e(y)), K(e(y)|e(x))\} + O(1)$, the information distance between $e(x)$ and $e(y)$ which is known to be incomputable [1]. ■

Proof: of Theorem II.5.

($\leq$) We use a modification of the proof of [11, Theorem 2]. According to Definition II.1 $x = y$ iff $e(x) = e(y)$. Let $X = \{x_1, \ldots, x_n\}$ and $k = \max_{x \in X}\{K(e(X)|e(x)\}$. A set of cardinality n in S is for the purposes of this proof represented by an n-vector of which the entries consist of the lexicographic length-increasing sorted members of the set. For each $1 \leq i \leq n$ let $\mathcal{Y}_i$ be the set of computably enumerated n-vectors $Y = (y_1, \ldots, y_n)$ with entries in S such that $K(e(Y)|e(y_i)) \leq k$ for each $1 \leq i \leq n$. Define the set $V = \bigcup_{i=1}^{n} \mathcal{Y}_i$. This V is the set of vertices of a graph $G = (V, E)$. The set of edges E is defined by: two vertices $u = (u_1, \ldots, u_n)$ and $v = (v_1, \ldots, v_n)$ are connected by an edge iff there is $1 \leq j \leq n$ such that $u_j = v_j$. There are at most 2^k self-delimiting programs of length at most k computing from input $e(u_j)$ to different $e(v)$'s with u_j in vertex v as jth entry. Hence there can be at most 2^k vertices v with u_j as jth entry. Therefore, for every $u \in V$ and $1 \leq j \leq n$ there are at most 2^k vertices $v \in V$ such that $v_j = u_j$. The vertex-degree of graph G is therefore bounded by $n2^k$. Each graph can be vertex-colored by a number of colors equal to the maximal vertex-degree. This divides the set of vertices V into disjoint color classes $V = V_1 \bigcup \cdots \bigcup V_D$ with $D \leq n2^k$. To compute $e(X)$ from $e(x)$ with $x \in X$ we only need the color class of which $e(X)$ is a member and the position of x in n-vector X. Namely, by construction every vertex with the same element in the jth position is connected by an edge. Therefore there is at most a single vertex with x in the jth position in a color class. Let x be the jth entry of n-vector X. It suffices to have a program of length at most $\log(n2^k) + O(\log nk) = k + O(\log nk)$ bits to compute $e(X)$ from $e(x)$. From n and k we can generate G and given $\log(n2^k)$ bits we can identify the color class V_d of $e(X)$. Using another $\log n$ bits we define the position of x in the n-vector X. To make such a program self-delimiting add a logarithmic term. In total $k + O(\log k)$ suffices since $O(\log k) = O(\log n + \log nk)$.

($\geq$) That $EG_{\max}(X) \geq \max_{x \in X}\{K(e(X)|e(x)\}$ follows trivially from the definitions. ■

Proof: of Lemma III.1.

(≥ 0) Since $f(X) \leq f(x)$ for all $x \in X$ the numerator of the right-hand side of (II.3) is

nonnegative. Since the denominator is also nonnegative we have $NWD(X) \geq 0$. Example of the lower bound: if $\max_{x\in X}\{\log f(x)\} = \log f(X)$, then $NWD(X) = 0$.

(≤ 1) Intuitively the upper bound on $g(X)$ is reached if the web events $e(x)$ for $x \in X$ are mutually almost disjoint. We say "almost" since if $\bigcap_{x\in X} e(x) = \varnothing$ then $NWD(X)$ is undefined.

Case 1 Let the web events $e(x)$ satisfy $|\bigcap_{x\in X} e(x)| = 1$. Then $g(X) = \prod_{x\in X}(g(x) - 1/N) + 1/N$. By (II.2) therefore $\sum_{x\in X} G(x) - G(X) = \epsilon$ for some very small positive ϵ.

Subcase 1.a Let $|e(x)| = |e(y)|$ for all $x, y \in X$. Then $G(X) - \min_{x\in X}\{G(x)\} = (X-1)\max_{x\in X}\{G(x)\} - \epsilon$. By (II.3) we have $NWD(X) = 1 - \epsilon'$ where $\epsilon' = \epsilon/((X-1)\max_{x\in X}\{G(x)\})$.

Subcase 1.b Let $|e(x)| \neq |e(y)|$ for some $x, y \in X$. Then $G(X) - \min_{x\in X}\{G(x)\} < (X-1)\max_{x\in X}\{G(x)\}$. By (II.3) we have $NWD(X) < 1 - \epsilon'$.

Case 2 Let the web events $e(x)$ satisfy $|\bigcap_{x\in X} e(x)| > 1$. Then $g(X) > \prod_{x\in X}(g(x) - 1/N) + 1/N$ yielding $\sum_{x\in X} G(x) - G(X) < \epsilon$ and therefore $G(X) - \min_{x\in X}\{G(x)\} < (X-1)\max_{x\in X}\{G(x)\}$. By (II.3) we have $NWD(X) < 1 - \epsilon'$. ■

Proof: of Lemma III.2.

(i) Since $X \subseteq Y$ and because of the condition of item (i) we have $\min_{y\in Y}\{\log f(y)\} = \min_{x\in X}\{\log f(x)\}$. From $X \subseteq Y$ also follows $\max_{y\in Y}\{\log f(y)\} \geq \max_{x\in X}\{\log f(x)\}$, and $\log f(X) \geq \log f(Y)$. Therefore the numerator of $NWD(Y)$ is at least as great as that of $NWD(X)$, and the denominator of $NWD(Y)$ equals $(|Y|-1)/(|X|-1)$ times the denominator of $NWD(X)$.

(ii) We have $\min_{x\in Y} \log f(y) < \min_{x\in X}\{\log f(x)\}$. If $NWD(X) = 1$ then $NWD(Y) = 1$ (in both cases there is no common similarity of the members of the set). Item (ii) follows vacuously in this case. Therefore assume that $NWD(X) < 1$. Write $NWD(X) = a/b$ with a equal to the numerator of $NWD(X)$ and b equal to the denominator. If c, d are real numbers satisfying $c/d \geq a/b$ then $bc \geq ad$. Therefore $ab + bc \geq ab + ad$ which rearranged yields $(a+c)/(b+d) \geq a/b$. If $c/d < a/b$ then by similar reasoning $(a+c)/(b+d) < a/b$.

Assume (III.1) holds. We take the logarithms of both sides of (III.1) and rearrange it to obtain $\log f(X) - \max_{x\in X}\{\log f(x)\} - \log f(Y) + \max_{y\in Y}\{\log f(y)\} \geq (\min_{x\in X}\{\log f(x)\} - \min_{y\in Y}\{\log f(y)\})(|X| - 1)NWD(X)$. Let the lefthand side of the inequality be c and the

righthand side of the inequality be $dNWD(X)$. Then

$$NWD(X) = \frac{\max_{x\in X}\{\log f(x)\} - \log f(X)}{(\log N - \min_{x\in X}\{\log f(x)\})(|X|-1)} \leq \frac{\max_{y\in Y}\{\log f(y)\} - \log f(Y)}{(\log N - \min_{y\in Y}\{\log f(y)\})(|X|-1)} = \frac{|Y|-1}{|X|-1} NWD(Y). \tag{A.2}$$

The inequality holds by the rewritten (III.1) and the a, b, c, d argument above since $c/d \geq NWD(X) = a/b$.

Assume (III.1) does not hold, that is, it holds with the $\geq$ sign replaced by a $<$ sign. We take logarithms of both sides of this last version and rewrite it to obtain $\log f(X) - \max_{x\in X}\{\log f(x)\} - \log f(Y) + \max_{y\in Y}\{\log f(y)\} < (\min_{x\in X}\{\log f(x)\} - \min_{y\in Y}\{\log f(y)\})(|X|-1)NWD(X)$. Let the lefthand side of the inequality be c and the righthand side $dNWD(X)$. Since $c/d < NWD(X) = a/b$ we have $a/b > (a+c)/(b+d)$ by the $a, b, c.d$ argument above. Hence (A.2) holds with the $\leq$ sign switched to a $>$ sign. It remains to prove that $NWD(Y) \geq NWD(Z)(|Z|-1)/(|Y|-1)$. This follows directly from item (i). ■

Proof: of Lemma III.5.

The following is a counterexample. Let $X = \{x_1\}$, $Y = \{x_2\}$, $Z = \{x_3, x_4\}$, $\max_{x\in XY}\{\log f(x)\} = 10$, $\max_{x\in XZ}\{\log f(x)\} = 10$, $\max_{x\in ZY}\{\log f(x)\} = 5$, $\log f(XY) = \log f(XZ) = \log f(ZY) = 3$, $\min_{x\in XY}\{\log f(x)\} = \min_{x\in XZ}\{f(x)\} = \min_{x\in ZY}\{\log f(x)\} = 4$, and $\log N = 35$. This arrangement can be realized for queries x_1, x_2, x_3, x_4. (As usual we assume that $e(x_i) \neq e(x_j)$ for $1 \leq i, j \leq 4$ and $i \neq j$.) Computation shows $NWD(XY) > NWD(XZ) + NWD(ZY)$ since $7/31 > 7/62 + 1/62$. ■

Proof: of Theorem III.7.

We start with the following:

Claim A.1. $EG_{\max}(X)$ is an admissible web distance function and $EG_{\max}(X) \leq D(X)$ for every computable admissible web distance function D.

Proof: Clearly $EG_{\max}(X)$ satisfies items (i) and (ii) of Definition III.6. To show it is an admissible web distance it remains to establish the density requirement (iii). For fixed x consider the sets $X \ni x$ and $|X| \geq 2$. We have

$$\sum_{X: X\ni x \ \&\ |X|\geq 2} 2^{-EG_{\max}(X)} \leq 1,$$

since for every x the set $\{EG_{\max}(X) : X \ni x \ \& \ EG_{\max}(X) > 0\}$ is the length set of a binary prefix code and therefore the summation above satisfies the Kraft inequality [9] given by (II.1). Hence $EG_{\max}$ is an admissible distance.

It remains to prove minorization. Let D be a computable admissible web distance, and the function f defined by $f(X, x) = 2^{-D(X)}$ for $x \in X$ and 0 otherwise. Since D is computable the function f is computable. Given D, one can compute f and therefore $K(f) \leq K(D) + O(1)$. Let $\mathbf{m}$ denote the universal distribution [12]. By [12, Theorem 4.3.2] $c_D\mathbf{m}(X|x) \geq f(X, x)$ with $c_D = 2^{K(f)} = 2^{K(D)+O(1)}$, that is, c_D is a positive constant depending on D only. By [12, Theorem 4.3.4] we have $-\log \mathbf{m}(X|x) = K(X|x) + O(1)$. Altogether, for every $X \in \mathcal{X}$ and for every $x \in X$ holds $\log 1/f(X, x) \geq K(X|x) + \log 1/c_D + O(1)$. Hence $D(X) \geq EG_{\max}(X) + \log 1/c_D + O(1)$. ■

By Lemma II.4 the function $EG_{\max}$ is upper semicomputable but not computable. The function $G(X) - \min_{x\in X}\{G(x)\}$ is a computable and an admissible function as in Definition III.6. By Claim A.1 it is an upper bound on $EG_{\max}(X)$ and hence $EG_{\max}(X) < G(X) - \min_{x\in X}\{G(x)\}$. Every admissible property or feature that is common to all members of X is quantized as an upper bound on $EG_{\max}(X)$. Thus, the closer $G(X) - \min_{x\in X}\{G(x)\}$ approximates $EG_{\max}(X)$, the better it approximates the common admissible properties among all search terms in X. This $G(X) - \min_{x\in X}\{G(x)\}$ is the numerator of $NWD(X)$. The denominator is $\max_{x\in X}\{G(x)\}(|X| - 1)$, a normalizing factor suited to the numerator of $NWD(X)$. It is chosen such that the quotient $NWD(X)$ has a value in $[0, 1]$ (Lemma III.1). ■

REFERENCES

[1] C.H. Bennett, P. Gács, M. Li, P.M.B. Vitányi, and W. Zurek. Information distance, *IEEE Trans. Inform. Theory*, 44:4(1998), 1407–1423.

[2] D. Bollegala, M. Yutaka, and I. Mitsuru, Measuring semantic similarity between words using web search engines, Proc. WWW., Vol. 766, 2007.

[3] P.-I. Chen and S.-J. Lin, Automatic keyword prediction using Google similarity distance, *Expert Systems with Applications*, 37:3(2010), 1928–1938.

[4] Cohen, A. R., C. Bjornsson, S. Temple, G. Banker and B. Roysam, Automatic Summarization of Changes in Biological Image Sequences using Algorithmic Information Theory, *IEEE Trans. Pattern Anal. Mach. Intell.* 31(8):(2009) 1386-1403.

[5] R.L. Cilibrasi, P.M.B. Vitányi, The Google similarity distance, *IEEE Trans. Knowledge and Data Engineering*, 19:3(2007), 370-383.

[6] P. Gács, On the symmetry of algorithmic information, *Soviet Math. Doklady*, 15:1477–1480, 1974. Correction, Ibid., 15(1974), 1480.

[7] R. Gligorov, W. ten Kate, Z. Aleksovski and F. van Harmelen, Using Google distance to weight approximate ontology matches, Proc. 16th Intl Conf. World Wide Web, ACM Press, 2007, 767–776.

[8] A.N. Kolmogorov, Three approaches to the quantitative definition of information, *Problems Inform. Transmission* 1:1(1965), 1–7.

[9] L.G. Kraft, A device for quantizing, grouping, and coding amplitude modulated pulses, MS Thesis, EE Dept., Massachusetts Institute of Technology, Cambridge. Mass., USA, 1949.

[10] L.A. Levin, Laws of information conservation (nongrowth) and aspects of the foundation of probability theory, *Probl. Inform. Transm.*, 10(1974), 206–210.

[11] C. Long, X. Zhu, M. Li, B. Ma, Information shared by many objects, Proc. 17th ACM Conf. Information and Knowledge Management, 2008, 1213–1220.

[12] M. Li and P.M.B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*, Springer-Verlag, New York, Third edition, 2008.

[13] B. McMillan, Two inequalities implied by unique decipherability, *IEEE Trans. Information Theory*, 2:4(1956), 115-116.

[14] J.-B. Michel, Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, T. G. B. Team, et al., Quantitative Analysis of Culture Using Millions of Digitized Books, *Science*, vol. 331, pp. 176-182, January 14 2011.

[15] Ng, A. Y., M. Jordan and Y. Weiss, On Spectral Clustering: Analysis and an algorithm, *Advances in Neural Information Processing Systems*, 14,2002.

[16] C.E. Shannon, The mathematical theory of communication, *Bell System Tech. J.*, 27(1948), 379–423, 623–656.

[17] Tibshirani, R., G. Walther and T. Hastie, Estimating the number of clusters in a dataset via the gap statistic, *Journal of the Royal Statistical Society* 63:(2001) 411 - 423.

[18] P.M.B. Vitányi, Information distance in multiples, *IEEE Trans. Inform. Theory*, 57:4(2011), 2451-2456.

[19] I.H. Witten, and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2005.

[20] W.L. Woon and S. Madnick, Asymmetric information distances for automated taxonomy construction, *Knowl. Inf. Systems*, 21(2009), 91–111.

[21] Z. Xian, K. Weber and D.R. Fesenmaier, Representation of the online tourism domain in search engines, *J. Travel Research*, 47:2(2008), 137–150.

[22] Kamboh, M. I., et al. (2012). "Genome-wide association study of Alzheimer's disease." *Transl Psychiatry* 2: e117.

[23] (2013). "Age of onset of amyotrophic lateral sclerosis is modulated by a locus on 1p34.1." *Neurobiology of Aging* 34(1): 357.e357-357.e319.

[24] Sill, F. C. M., et al. (2012). "Post-GWAS Functional Characterization of Susceptibility Variants for Chronic Lymphocytic Leukemia." *PLoS One* 7(1): e29632.

[25] Maris, J. M., et al. (2008). "Chromosome 6p22 Locus Associated with Clinically Aggressive Neuroblastoma." *New England Journal of Medicine* 358(24): 2585-2593.

[26] Scherag, A., et al. (2010). "Two New Loci for Body-Weight Regulation Identified in a Joint Analysis of Genome-Wide Association Studies for Early-Onset Extreme Obesity in French and German Study Groups." *PLoS Genet* 6(4): e1000916.

[27] Soto-Ortolaza, A. I., et al. (2013). "GWAS risk factors in Parkinson's disease: LRRK2 coding variation and genetic interaction with PARK16." *Am J Neurodegener Dis* 2(4): 287-299.

[28] Schizophrenia Working Group of the Psychiatric Genomics, C. (2014). "Biological insights from 108 schizophrenia-associated genetic loci." *Nature* 511(7510): 421-427.

[29] Huang, H. S., et al. (2007). "Prefrontal dysfunction in schizophrenia involves mixed-lineage leukemia 1-regulated histone methylation at GABAergic gene promoters." J Neurosci 27(42): 11254-11262.

[30] Chen, H., et al. (2004). "Obesity and the risk of Parkinson's disease." Am J Epidemiol 159(6): 547-555.