Quick Matching of Big Binary Data: A Probabilistic Approach

Adnan A. Y. Mustafa*

Department of Mechanical Engineering, Kuwait University, P. O. Box 5969 - Safat - Kuwait 13060; adnan.mustafa@ku.edu.kw

Abstract

Given two sets of binary data, how can we determine if the data are dissimilar? The simplest technique is to simply subtract the two sets or to calculate the correlation between them. Both of these methods –as well as other methods– require some type of similarity operation to be applied to all points of the data. This implies that as the data becomes big, more processing time is required. In this paper, we present a novel approach to matching using a probabilistic model that requires a few number of points –and not all points – to be compared between two data sets to detect dissimilarity. Furthermore, the model is size invariant; big data can be matched just as quickly as matching small data. The similarity between the data can also be measured to a good degree by repeating the matching process several times.

Keywords: Big Data, Binary Data, Binary Matching, Pattern Matching, Probabilistic Model

1. Introduction

In the information age, big masses of data are being generated through experiments, measurements, modeling and simulation. This is not limited to engineering and science disciplines, but is true for all disciplines. The need to analyze this data in a timely manner is becoming an important issue.

In many instances, the data is binary, such as in the field of signal processing, computer science, space exploration, mineral mining and biology, where the need to compare binary data arises. If the data is large then an efficient way of comparing this data becomes essential.

For example, in the field of image analysis, binary images are usually produced as a result of applying a segmentation process to an image, and the resulting image is then compared to a library of candidate images for identification and recognition purposes. With image sizes exceeding 20 MP common today, and image size doubling every 5 years, comparing such big images to a library consisting of thousands of images using current state-of-the-art matching methods is an extremely time consuming process despite the processing speed of cur-

*Author for correspondence

rent computer systems. This is due to the fact that all current matching methods are image size dependent; as image size increases so does matching time.

Speech recognition is another field that is binary based. Sound acquisition is performed by converting an analog signal to a digital signal (via an ADC) and the data is saved in binary form. This data is usually large; sound recordings usually sampled at 48 kHz with 32 bits/sample produce slightly more than 1.5 million bits/ second. Hence, comparing 10 seconds of an audio recording requires the comparison of over 15 million bits. As a result, comparison of audio files can become a time-consuming process when the audio files are big for longer or higher-quality recordings.

As a third example –this time not from the field of signal processing– is in the field of computer file processing. Computer files are stored in binary format and the average computer file size is growing with every passing year. Twenty years ago, a 10 MB file was considered a large file; today file sizes of more than 100 MB, or even a 1 GB file, are common. Comparing files to detect file duplication or to measure the similarity between them for instance, should not require the comparison of the whole files. In general, comparing two sets of binary data to detect dissimilarity (or measuring their similarity) can be accomplished using many methods. Perhaps the most common technique is to calculate the correlation between the data¹, or to simply subtract the two sets². These methods, as well as the majority –if not all- of the methods require some type of similarity criteria to be evaluated at all point correspondences. Hence, these methods are data size dependent; as data size increases, more processing time is required. When the size of the data is big, these well-established methods can become quite slow, especially when matching thousands of data sets.

In this paper, we show that not all of the data needs to be compared to detect or measure dissimilarity/similarity, but rather only, a few randomly selected points is sufficient for the task. This is possible by performing the matching task probabilistically. We show using our probabilistic binary matching model that if the data is completely different, i.e. they are the inverse of each other and have complement values at each point, then only one point needs to be mapped to detect dissimilarity, as is intuitively obvious. As the amount of similarity increases between the data, more points need to be mapped to detect dissimilarity. For example: data that are 10% similar can be detected with a single mapping with 90% confidence; with 2 mappings the detection confidence jumps higher to 99%, and increases with more mappings. Data that are 50% similar can be detected with a single mapping with 50% confidence and with 2 mappings the detection confidence jumps up to 75%. On the other hand, very similar data that are 90% similar, have a detection confidence of only 10% for a single mapping, but reaches 50% confidence by the 7th mapping, and 90% confidence by the 22nd mapping. This scenario is analogist to looking at two very similar pictures that dictates the fine examination of many parts of the pictures before a difference between them can be detected. Furthermore, we show that using this approach, data size is irrelevant; big data can be processed just as fast as small data. As a result, this approach is magnitudes faster than current state-of-the-art methods that are size dependent.

This paper is organized as follows: section 2 discusses our approach of representing binary data as binary vectors. We also discuss vector closeness and how it is measured. Section 3 presents the main theme of this paper and presents the probabilistic model for matching binary vectors. Section 4 explains the mechanics of how dissimilar vectors can be detected by examining only few points. Section 5 presents examples of the application of the matching model. Section 6 presents our conclusions.

2. Binary Vectors

Our approach is to represent binary data as binary vectors and match these vectors. The data can be of any dimension: one-dimensional (e.g. sound files), two-dimensional: (e.g. images and matrices), three-dimensional (e.g. geographical data and CAT-scans), etc. The geometry of the data can be of any type as long as they have a similar arrangement and a one-to-one correspondence exists between all points of the matched data, as shown in Figure 1. Regardless of the dimension of the data or its geometry, all data will be matched as if they are binary vectors, with the order of the data preserved. Hence, we are matching one-dimensional binary patterns.



Figure 1. Different Binary pattern pairs. a) Rectangular Grids b) Hexagonal Grids c) Triangular Grids.

2.1 Binary Vector Mapping

Initially let us define some terms that will be used in this paper. Let **u** and **v** be two binary vectors. Let $u \in \mathbf{u}$ and $v \in \mathbf{v}$, where *u* and *v* are independent random variables.

Element mapping (P_1) between two binary vectors refers to how an element value in the first vector maps to the corresponding element value in the second vector, i.e. how the values of the two vectors map to each other in a specific direction,

 $P_{1} = \{ \mathbf{u} \rightarrow \mathbf{v} \}, \, \forall u \,, v \in \{0, 1\}$ $\tag{1}$

The ' \rightarrow ' symbol is used to denote element mapping. Hence, $u \rightarrow v$ implies element value u of the first vector maps to element value v in the second vector.

2.2 Binary Vector Similarity

Many similarity measures and distances have been developed for binary data over the last century that can be found in the literature^{3,4}. In this section we discuss vector closeness and the similarity measure that is used in our work to measure similarity.

2.2.1 Similar and Dissimilar Vectors

The closeness between two vectors is based on an element-to-element comparison of the two vectors. Vector closeness is categorized as either *similar* or *dissimilar*:

1. *Similar* vectors (*s*): The two vectors are considered to be exactly the same. This can only be true if the two vectors have the same values at all corresponding elements, e.g. $\mathbf{u} = [1 \ 1 \ 1 \ 0]$ and $\mathbf{v} = [1 \ 1 \ 1 \ 0]$.

2. *Dissimilar* vectors (*d*): The two vectors are different. This can only be true if the two vectors are not similar, and they are of two types:

a. Inverse vectors (*i*): The two vectors have compliment values at all corresponding elements, e.g. $\mathbf{u} = [0 \ 1 \ 0 \ 1]$ and $\mathbf{v} = [1 \ 0 \ 1 \ 0]$.

b. Quasi-dissimilar vectors (*q*): The two vectors are neither similar nor inverse, e.g. $\mathbf{u} = [0\ 1\ 0\ 1]$ and $\mathbf{v} = [1\ 0\ 1\ 1]$.

2.2.2 Vector Similarity and Dissimilarity Coefficients

Let κ^0 and κ^1 denote the vector similarity and dissimilarity coefficients, respectively, defined as:

$$\kappa^0(u,v) \equiv \phi((u \oplus v) = 0) \tag{2}$$

$$\kappa^{1}(u,v) \equiv \phi((u \oplus v) = 1)$$
⁽³⁾

where \oplus denotes the exclusive-or operator, $\varphi()$ is the probability mass function and $0 \le \kappa^0, \kappa^1 \le 1$. Hence,

$$\kappa^{0} + \kappa^{1} = 1 \tag{4}$$

If the Hamming distance $(d_{H})^{5}$ is given by,

$$d_H(\mathbf{u}, \mathbf{v}) = \sum_{i=0}^n (u_i \oplus v_i)$$
(5)

then it can be seen that,

$$\kappa^{1}(\mathbf{u} \oplus \mathbf{v}) = \frac{1}{n} d_{H}(\mathbf{u}, \mathbf{v})$$
(6)

where *n* is the size of the vector. Furthermore, κ^1 is equivalent to the Sokal-Michener Metric⁶ (aka Simple Matching Coefficient (*SMC*)). Values of κ^0 and κ^1 correspond to,

- $\kappa^0 = 0$ or $\kappa^1 = 1$ for inverse vector pairs.
- $0 < \kappa^0, \kappa^1 < 1$ for quasi-dissimilar vector pairs.
- $\kappa^0 = 1$ or $\kappa^1 = 0$ for similar vector pairs.

3. A Probabilistic Mapping Model for Binary Vectors

In this section, we present the development of a probabilistic mapping model for detecting dissimilar binary vectors. The model predicts the probability of occurrence of dissimilar vectors when matching two binary vectors. In such a model, it is assumed that vector element locations are randomly selected and their intensity values are mapped.

3.1 The Probabilistic Mapping Model

Let *d* denote the event of occurrence of dissimilar vectors and *s* denote the event of occurrence of similar vectors. Let k^* , $0 \le k^* \le 1$, be a random variable representing the probability of event *d* occurring at any given mapping; as a result, the probability of *s* occurring is $(1 - k^*)$. On the first mapping two possible states are possible; *d* or *s*; the probability of occurrence of *d* is k^* and the probability of *s* occurring is $(1 - k^*)$. On the second mapping, four cases are possible: *dd*, *ds*, *sd* and *ss*; their probabilities are k^{*2} , $k^*(1 - k^*)$, $k^*(1 - k^*)$ and $(1 - k^*)^2$, respectively. On the third mapping there are 8 cases, and so on for further mappings. It can be seen that the probability distribution of *d* is a Binomial distribution given by,

$$\varphi(X = x, p, k^*) = \binom{p}{x} (k^*)^x (1 - k^*)^{p - x}$$

$$x = 0, 1... p$$
(7)

where, *X* is a random variable denoting the number of times *d* occurs in *p* mappings and φ is the probability mass function of *d* occurring *x* times in *p* mappings. Let *S*

denote the *s* events only set, *I* the *d* events only set, and *M* the mixed events set, defined as follows:

$$S = \{s, ss, sss, \ldots\}$$
(8)

$$I = \{d, dd, ddd, \ldots\}$$
(9)

$$M = \{sd, ssd, dss, \ldots\}$$
(10)

These three sets partition the sample space. The probability of occurrence of *S* in *p* mappings, Pr(S,p,k), is then,

$$Pr(S, p, k) = \varphi(X = 0, p, k)$$

= $(1 - k^*)^p$ $p > 1.2$ (11)

If $D = I \cup M = \overline{S}$, then the probability of occurrence of *D* in *p* mappings, Pr(D,p,k), is then,

$$Pr(D, p, k^*) = \varphi(0 \le X < p, p, k^*)$$

= 1-\varphi(X = p, p, k^*) (12)

Hence,

$$Pr(D,p,k^*) = 1 - (1 - k^*)^p$$

$$p = 1, 2, \dots \text{ and } 0 \le k \le 1$$
(13)

But by its definition we see that,

$$(1 - k^*) = \kappa^0(\mathbf{u}, \mathbf{v}) = \phi((\mathbf{z} = (\mathbf{u} \bigoplus \mathbf{v})) = 0)$$
(14)

where, $\phi(\mathbf{z})$ is the probability mass function of $\mathbf{z} = (\mathbf{u} \bigoplus \mathbf{v})$. For convenience we will use, $k = 1 - k^* = \kappa^0$, as the vector similarity coefficient. Thus,

 $\Pr(D,p,k) = 1 - k^p$

$$p = 1, 2, \dots \text{ and } 0 \le k \le 1$$
 (15)

This simple equation defines the probabilistic model for matching binary vectors. Note that $k \in [0,1]$ is a continuous variable while p is a positive nonzero integer variable. Pr increases with increasing k, but decreases with increasing p except at k = 0 and k = 1. Hence,

$$Pr(D,p,k) < Pr(D,p+i,k)$$

$$p = 1, 2, ..., 0 < k < 1 \text{ and } i = 1, 2, ...$$
(16)

$$\Pr(D,p,k) > \Pr(D,p,k+a)$$

$$p = 1, 2, \dots \text{ and } 0 < k + a < 1$$
 (17)

Pr(D,p,k) is a cumulative distribution function (cdf) of the probability distribution function of detecting dissimilar vectors at a given mapping. Furthermore, the value of Pr(D,p,k) can be interpreted as the detection confidence (*DC*) of detecting dissimilar binary vectors for given values of *p* and *k*. Hence, (15) can be stated as,

$$DC(D,p,k) = 1 - k^{p}$$
 $p = 1, 2, ...$ and $0 \le k \le 1$ (18)

We will use both equivalent forms of this equation ((15) and (18)) as warranted in our discussion. Furthermore, from (18) the vector similarity coefficient kcan be expressed as a function of *DC* and *p*,

$$k(D, DC, p) = (1 - DC)^{\frac{1}{p}}$$

 $p = 1, 2, ... \text{ and } 0 \le DC \le 1$ (19)

Similarly, the number of mappings p can be expressed as a function of *DC* and k,

$$p(D, DC, k) = \frac{\log(1 - DC)}{\log(k)}$$

$$0 \le DC \le 1 \text{ and } 0 \le k \le 1$$
(20)

which determines the number of mappings required to detect dissimilarity for given values of *DC* and *k*. Several curves of Pr(D,p,k) versus *p* for different values of *k* are shown in Figure 2. Figure 3 shows curves of Pr(D,p,k) for highly similar vectors, $k \ge 0.75$.



Figure 2. Curves of Pr(D,p,k) versus p for several values of k.



Figure 3. Curves of Pr(D,p,k) versus p for high values of k.

We see from the figures, that the curves share a common profile:

 DC values increase as p increases for 0 < k < 1 and reach a limit of 1,

$$\lim_{p \to \infty} \Pr(p, k) = \lim_{p \to \infty} \left(1 - k^p \right) = 1$$
(21)

• Curves with smaller *k* values reach unity faster than those with larger *k* values, i.e., as *k* increases

							_			
р	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
1	1.0000	0.9000	0.8000	0.7000	0.6000	0.5000	0.4000	0.3000	0.2000	0.1000
2	1.0000	0.9900	0.9600	0.9100	0.8400	0.7500	0.6400	0.5100	0.3600	0.1900
3	1.0000	0.9990	0.9920	0.9730	0.9360	0.8750	0.7840	0.6570	0.4880	0.2710
4	1.0000	0.9999	0.9984	0.9919	0.9744	0.9375	0.8704	0.7599	0.5904	0.3439
5	1.0000	1.0000	0.9997	0.9976	0.9898	0.9688	0.9222	0.8319	0.6723	0.4095
6	1.0000	1.0000	0.9999	0.9993	0.9959	0.9844	0.9533	0.8824	0.7379	0.4686
7	1.0000	1.0000	1.0000	0.9998	0.9984	0.9922	0.9720	0.9176	0.7903	0.5217
8	1.0000	1.0000	1.0000	0.9999	0.9993	0.9961	0.9832	0.9424	0.8322	0.5695
9	1.0000	1.0000	1.0000	1.0000	0.9997	0.9980	0.9899	0.9596	0.8658	0.6126
10	1.0000	1.0000	1.0000	1.0000	0.9999	0.9990	0.9940	0.9718	0.8926	0.6513
11	1.0000	1.0000	1.0000	1.0000	1.0000	0.9995	0.9964	0.9802	0.9141	0.6862
12	1.0000	1.0000	1.0000	1.0000	1.0000	0.9998	0.9978	0.9862	0.9313	0.7176
13	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	0.9987	0.9903	0.9450	0.7458
14	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	0.9992	0.9932	0.9560	0.7712
15	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9995	0.9953	0.9648	0.7941
16	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9997	0.9967	0.9719	0.8147
17	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9998	0.9977	0.9775	0.8332
18	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	0.9984	0.9820	0.8499
19	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	0.9989	0.9856	0.8649
20	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9992	0.9885	0.8784

 Table 1. Detection confidence vs. number of mappings (p) for selected values of k

more mappings are required to reach a specific *DC* value.

For the special two cases of k = 0 and k = 1, these two curves have unique profiles,

- The *k* = 0 curve (inverse vectors) is a constant line at 1, achieved on the first mapping.
- The *k* = 1 curve (similar vectors) is zero for all mappings.

Table 1 lists the *DC* values for the first 20 mappings for $0 \le k \le 0.9$, in increments of 0.1 (rounded to 4 decimal digits). We see that, for k = 0.1, the *DC* value quickly reaches a value of 1.0000 by the 5th mapping, i.e., performing 5 mappings is sufficient to detect dissimilarity 100% for vectors with similarity k = 0.1. For k = 0.2, *DC* = 1.0000 in 7 mappings and for k = 0.3, *DC* = 1.0000 in 9 mappings. As vectors become more similar and k increases more mappings are required to reach *DC* = 1.0000. For example, *DC* = 1.0000 is reached at 29, 46 and 94 mappings for k = 0.7, 0.8 and 0.9, respectively.

Figure 4 shows curves of Pr(D,p,k) versus *k* for several values of *p*. "For *p* = 1, the Pr(D,1,k) curve" reduces to a straight line which can be seen from (15),

 $\Pr(D,1,k) = 1-k$

(22)

This line represents the probability of detecting dissimilar vectors on the first mapping, which decreases linearly as k increases. At k = 1 (i.e. similar vectors) Pr(D,1,1) = 0, i.e. there is no possibility that the vectors can be determined to be dissimilar on the first mapping; in fact Pr(D,p,1) = 0 implying that it's impossible for the vectors to be dissimilar for any number of mappings. On the other hand, if k = 0 (i.e. inverse vectors) then Pr(D,1,0)= 1 (and Pr(D,p,0) = 1), i.e. just one mapping is sufficient to determine that the vectors are dissimilar. It is interesting to note that even when vectors are very similar, the probability of detecting dissimilarity between them with a single mapping is still possible, even though that probability is very low (e.g. if k = 0.9, then Pr(D,1, 0.9) = 0.1).

Figure 5 shows curves of p(D,DC,k) versus k for several *DC* values. For 50% confidence rate, *DC* = 0.50, the curve starts at k = 0.5 and is monotonically increasing. For $k \le 0.5$, *DC* = 0.50 can be achieved by the first mapping; there is 50% chance or more that dissimilarity can be detected on the first mapping. For k > 0.5 more than one mapping is required to detect dissimilarity and as $k \rightarrow$

1, the curve rapidly grows upward signifying that much more points are needed to be mapped to reach a 50% detection. This profile is common for all DC values, but as DC increases, the curves intersect the single mapping line at lower values; more points are required to detect dissimilarity with higher DC values.



Figure 4. Curves of Pr(D,p,k) versus k for several values of p.



Figure 5. Curves of *p*(*D*,*DC*,*k*) versus *k* for several *DC* values.

3.2 The Probability of Detecting Dissimilar Vectors

The probability distribution function of detecting dissimilar vectors, $P_D(p,k)$, can be obtained from its cdf; Pr(D,p,k),

$$P_{D}(p,k) = k^{p-1} - k^{p} \qquad p = 1, 2, \dots$$
 (23)

Table 2 lists values of $P_D(p,k)$ for the first 20 mappings for $0 \le k \le 1.0$, in increments of 0.1 (rounded to 4 decimal digits). Figure 6 shows plots of $P_D(p,k)$ versus p for sample values of k. Some remarks about the distribution:

- For *k* = 0: the distribution has a value of unity at *p* = 1 and zero elsewhere.
- For *k* = 1: the distribution values are very small and for all practical purposes can be considered to be zero everywhere, with the sum of the distribution being unity.
- For 0 < k < 1.0:
 - The distribution is monotonically decreasing.

- The first value in the distribution, P_D(1,k) = 1 k, is always the largest value of the distribution.
- As *k* increases the values of the distribution at any given *p* decreases.



Figure 6. Plots of $P_{D}(p,k)$ versus p for various values of k.

3.3 The Expected Value of the Mapping Detection Number

The expected value of *p* is given by,

$$E[p] = \sum_{p} p(k^{p-1} - k^{p})$$

= $\sum_{p} (pk^{p-1} - pk^{p})$
= $(1-k) + (2k - 2k^{2}) + (3k^{2} - 3k^{3}) + \cdots$
= $1 + (2k - k) + (3k^{2} - 2k^{2}) + \cdots$ (24)
= $1 + k + k^{2} + \cdots$

which is a geometric series with ratio 1, and thus the expected value of *p* is given by,

$$E[p] = \frac{1}{1-k} \tag{25}$$

This is an important equation as it gives the expected number of mappings required to detect dissimilarity.

p	k												
0	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0		
1	1.0000	0.9000	0.8000	0.7000	0.6000	0.5000	0.4000	0.3000	0.2000	0.1000	0.0000		
2	0.0000	0.0900	0.1600	0.2100	0.2400	0.2500	0.2400	0.2100	0.1600	0.0900	0.0000		
3	0.0000	0.0090	0.0320	0.0630	0.0960	0.1250	0.1440	0.1470	0.1280	0.0810	0.0000		
4	0.0000	0.0009	0.0064	0.0189	0.0384	0.0625	0.0864	0.1029	0.1024	0.0729	0.0000		
5	0.0000	0.0001	0.0013	0.0057	0.0154	0.0313	0.0518	0.0720	0.0819	0.0656	0.0000		
6	0.0000	0.0000	0.0003	0.0017	0.0061	0.0156	0.0311	0.0504	0.0655	0.0591	0.0000		
7	0.0000	0.0000	0.0001	0.0005	0.0025	0.0078	0.0187	0.0353	0.0524	0.0531	0.0000		
8	0.0000	0.0000	0.0000	0.0002	0.0010	0.0039	0.0112	0.0247	0.0419	0.0478	0.0000		
9	0.0000	0.0000	0.0000	0.0000	0.0004	0.0020	0.0067	0.0173	0.0336	0.0431	0.0000		
10	0.0000	0.0000	0.0000	0.0000	0.0002	0.0010	0.0040	0.0121	0.0268	0.0387	0.0000		
11	0.0000	0.0000	0.0000	0.0000	0.0001	0.0005	0.0024	0.0085	0.0215	0.0349	0.0000		
12	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0015	0.0059	0.0172	0.0314	0.0000		
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0009	0.0042	0.0137	0.0282	0.0000		
14	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0005	0.0029	0.0110	0.0254	0.0000		
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0020	0.0088	0.0229	0.0000		
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0014	0.0070	0.0206	0.0000		
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0010	0.0056	0.0185	0.0000		
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0007	0.0045	0.0167	0.0000		
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0005	0.0036	0.0150	0.0000		
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0029	0.0135	0.0000		

Table 2. $P_p(p,k)$ vs. number of mappings (p) for selected values of k

Let MDN denote the *Mapping Detection Number* which is defined as the number of mappings required to detect a pair of vectors as being dissimilar. MDN will be used to measure detection quickness. The expected MDN as a function of k is then,

$$E[MDN(k)] = \frac{1}{1-k} \tag{26}$$

A plot of this equation appears in Figure 7, and values of E[MDN] (i.e. E[p]) for selected k values are tabulated in Table 3. E[MDN] is a monotonically increasing curve: The values of E[MDN] are fairly small for $k \le 0.9$ (where $E[MDN] \le 10$) and increase gradually from a value of 1 at k = 0 to a value of 10 at k = 0.9. For larger values of k the curve increases rapidly; as $k \rightarrow 1$ results in $E[MDN] \rightarrow \infty$. Let MDN_{μ} be the mean value of MDN, i.e.,

$$MDN_{\mu} = E[MDN]$$
 (27)
Then (26) can be used to estimate *k*,

$$k(MDN_{\mu}) = \frac{MDN_{\mu} - 1}{MDN_{\mu}}$$
(28)

This equation will be used to measure the vector similarity coefficient quickly as described below.



Figure 7. A plot of the expected MDN vs. k.

The variance of *p* is given by,

$$V[p] = E[p^{2}] - (E[p])^{2}$$
⁽²⁹⁾

 $E[p^2]$ is derived as follows,

$$E[p^{2}] = \sum_{p} p^{2}(k^{p-1} - k^{p})$$

= (1-k) + 4(k - k^{2}) + 9(k^{2} - k^{3}) + ... (30)
= 1 + 3k + 5k^{2} + 7k^{3} + ...
= (2 + 4k + 6k^{2} + 8k^{3} ...) - (1 + k + k^{2} + ...)

The first term on the left hand side of (30) is the derivative of a geometric series with ratio 1. Hence,

$$2+4k+6k^{2}+8k^{3}\dots = 2(1+2k+3k^{2}+4k^{3}+\dots)$$

$$= 2\frac{d}{dk}(1+k+k^{2}+k^{3}+k^{4}+\dots)$$

$$= 2\frac{d}{dk}\left(\frac{1}{1-k}\right)$$

$$= \frac{2}{(1-k)^{2}}$$

(31)

The second term on the left hand side of (30) is also a geometric series with ratio 1. Hence, (30) simplifies to,

$$E[p^{2}] = \frac{2}{(1-k)^{2}} - \frac{1}{1-k}$$

$$= \frac{1+k}{(1-k)^{2}}$$
(32)

Substituting (25) and (32) into (29), the variance of *p* is f finally obtained,

$$V[p] = \frac{1+k}{(1-k)^2} - \left(\frac{1}{1-k}\right)^2$$

$$= \frac{k}{(1-k)^2}$$
(33)

The standard deviation of p is then,

$$\sigma[p] = \frac{\sqrt{k}}{1-k} \tag{34}$$

Both the variance and standard deviation have similar curve profiles to that of the expected value; they are monotonically increasing with the rate increasing as k approaches 1, and in the limit as $k \rightarrow 1$ their values become infinite. These equations can be used to estimate the variance and standard deviation of *MRN*. Values of $\sigma[p]$ for selected k values are tabulated in Table 4.

3.4 Size Invariance

Equations (15), (19) and (20) are not a function of the vector size (n); hence detecting dissimilarity is size invari-

Table 3. E[p] for selected values of k

ant using this probabilistic approach. Vector of any sizes can be matched with the same quickness, whether they consist of only a few elements or if they consist of millions of elements. The matching quickness is only dictated by the amount of similarity k between the vectors.

4. Detecting Dissimilar Vectors

In this section, the mechanics of detecting dissimilar vectors is explained. The mapping model predicts the number of mappings required to detect dissimilarity with a given confidence. We discuss below a method that can be used with the mapping model for dissimilarity detection.

4.1 A Dissimilarity Detection Method

Detecting dissimilarity between vectors based on the discussion presented can be accomplished in many ways; perhaps the simplest method is to repeatedly select points and compare them until the values differ. This strategy is very fast and can remarkably detect even very similar vectors very rapidly. If this strategy is to be used then the following items should be noted:

1. Points should be randomly selected.

2. A mapping limit (L_{map}) should be used to terminate the mapping process.

- Using such a simple procedure will result in either,
 - the vectors determined to be dissimilar, or
 - the matching trial being inconclusive (i.e. L_{map} is reached).

In the latter case, the vectors are highly similar, and L_{map} can be increased and the process repeated. If the type of dissimilarity (inverse/quasi-dissimilar) is to be determined, then a more elaborate detection scheme based on mapping tuples needs to be followed (not discussed in this paper).

4.2 Measuring Similarity

To measure similarity, (28) can be used. To obtain a good estimate of MDN_{u} , several detection trials should be con-

k	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	0.9999	0.99999
E[<i>p</i>]	1.000	1.111	1.25	1.429	1.667	2	2.5	3.333	5	10	100	1000	10000	100000

Table 4. $\sigma[p]$ for selected values of k

k	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	0.9999	0.99999
σ[<i>p</i>]	0.000	0.351	0.559	0.782	1.054	1.414	1.936	2.789	4.472	9.487	99.5	999.5	9999	100000



Figure 8. The binary vectors represented as 1D patterns. From top to bottom: v_1 , v_2 and v_3 .

ducted and the average value taken. It should be noted that using (28) gives an approximate value and not an exact value, due to the nature of the mapping process which is integer based. Nevertheless, MDN_{μ} values should coincide with E[MDN] values obtained from (26), which we show to be very true.

5. Discussion

For illustrative purposes, we present two examples.

5.1 Example 1

We initially present an example with small sized vectors. Let \mathbf{v}_1 , \mathbf{v}_2 and \mathbf{v}_3 , be three binary vectors of size n = 30, defined as follows,

 $\mathbf{v}_{1} = [000111010000010100100100000010]$

 $\mathbf{v}_2 = [010111011000010100101101000010]$

 $\mathbf{v}_{3} = [011100100000010000001101010010]$

For visualization purposes, these vectors are shown as patterns in Figure 8. Visual inspection of the three vectors immediately reveals that the vector pair consisting of $(\mathbf{v}_1, \mathbf{v}_2)$ is the most similar pair. In fact, k = 0.867 for this pair. The vector pair consisting of $(\mathbf{v}_1, \mathbf{v}_3)$ is the least similar with k = 0.633 and the last vector pair $(\mathbf{v}_2, \mathbf{v}_3)$ has k = 0.700. With these k values (26) predicts E[MDN] to be 7.5, 2.727, 3.333 mappings, respectively, i.e. on average 8 mappings should be sufficient to detect $(\mathbf{v}_1, \mathbf{v}_2)$ as being dissimilar, 3 mappings is sufficient to detect $(\mathbf{v}_1, \mathbf{v}_3)$ as being dissimilar, and 4 mappings should be sufficient to detect (v_2, v_3) as being dissimilar. Indeed for 500 trials tested, the MDN_u values found were 7.265, 2.634 and 3.230 for $(\mathbf{v}_1, \mathbf{v}_2)$, $(\mathbf{v}_1, \mathbf{v}_3)$ and $(\mathbf{v}_2, \mathbf{v}_3)$, respectively. All values are within a single mapping of the predicted E[MDN]. Figure 9 plots *MDN*_µ vs. trial no. for the three vector pairs where MDN₄ is calculated up to the trial no. shown. Also shown as a dotted line is E[MDN]. We see that the MDN_{μ} curves for all pairs quickly approach their respective E[MDN] values, and are within ± 1 mapping of E[MDN]by the 100th trial. Figure 10 shows plots of the estimated value of k vs. trial no. for the three vector pairs using (28). Also shown are the exact k values for each pair as dotted lines. Once again, we see that the estimated k value approaches the exact k values with increasing trials. The % error of estimating k vs. trial no. for the three vector pairs is shown in Figure 11, where it can be seen that the % error is less than 5% for all pairs after 200 trials.



Figure 10. Similarity coefficient (k) vs. number of trials for the binary vector pairs. Also shown are the k values for each vector pair (dotted lines).



Figure 9. MDN_{μ} vs. number of trials for the binary vector pairs. Also shown the E[MDN] for each vector pair (dotted lines).



Figure 11. Percent similarity coefficient error vs. number of trials for the binary vector pairs.



Figure 12. The binary matrices represented as 2D patterns. From left to right: M_1 , M_2 and M_3 .



Figure 13. Difference matrices represented as 2D patterns. From left to right: d_{12} , d_{13} and d_{23} .



Figure 14. MDN_{μ} vs. number of trials for the binary matrices. Also shown the E[MDN] **for each matrices pair.**



Figure 15. Similarity coefficient (k) vs. number of trials for the binary matrices. Also shown are the k values for each vector pair (dotted lines).



Figure 16. Percent similarity coefficient error vs. number of trials for the binary matrices.

5.2 Example 2

Figure 12 shows three binary matrices displayed as patterns, M₁, M₂ and M₃, of size $n = 50 \times 50$, represented as 2D patterns. The difference matrices are shown in Figure 13, where $\mathbf{d}_{ii} = |\mathbf{M}_i - \mathbf{M}_i|$. For the first matrix pair $(\mathbf{M}_1, \mathbf{M}_2)$ the matrices exhibit very high similarity with k = 0.965and corresponding E[MDN] = 28.74 mappings. Thus on average 29 mappings are required to detect dissimilarity between $(\mathbf{M}_{1}, \mathbf{M}_{2})$. This somewhat higher mapping value is due to the higher similarity that exists between these two matrices. Tests conducted produced $MDN_{\mu} = 26.76$ after 100 trials; ~2 mappings off E[MDN] but with increased trials produces closer results. This is shown in Figure 14 which plots MDN_{μ} vs. trial no. for the three matrix pairs where *MDN*_µ is calculated up to the trial no. shown. Also shown as a dotted line is E[MDN]. For the second pair (M_1, M_3) , k = 0.507 with corresponding E[MDN] = 2.03mappings. Thus on average 3 mappings are sufficient to detect dissimilarity between M₁ and M₂. From the first 5 trials $MDN_{\mu} = 1.6$ mappings, which is within 1 mapping of E[MDN] and as the number of trials increases the MDN_{μ} values rapidly approach E[MDN]. For the last matrix pair ($\mathbf{M}_{2}, \mathbf{M}_{2}$), k = 0.498 with E[MDN] = 1.99 mappings; once again the MDN_{μ} values are within ± 1 mapping off of *E*[*MDN*]. Figure 15 shows plots of the estimated value of *k* vs. trial no. for the three matrix pairs. Also shown are the exact k values for each pair as dotted lines. The estimated *k* values are in close proximity to the exact *k* values. The % error of estimating k vs. trial no. for the three matrix pairs is shown in Figure 16, where it can be seen that the % error is less than 4% for all three pairs after 200 trials.

6. Conclusion

In this paper we have presented a quick way to compare big binary data quickly. We showed that only a small fraction of the data needs to be compared to detect dissimilarity and not all of the data. This is accomplished by using a probabilistic matching model that predicts the number of points required to be compared between the two data sets to detect dissimilarity. The model shows that size is irrelevant. Big data can be matched as quickly as small data. The similarity between the data can also be measured to a good degree by repeating the detection process a few times. Tests conducted showed that experimental results are in good agreement with the models prediction.

7. References

- 1. Montgomery DC, Runger GC. Applied statistics and probability for engineers. John Wiley & Sons; 2010.
- Barnea D, Silverman H. A class of algorithms for fast digital vector registration. IEEE Transactions on Computers. 1972 Feb; c-21(N2):179–86.

- 3. Choi S, Sung-Hyuk C, Tappert CC. A survey of binary similarity and distance measures. Journal of Systemics, Cybernetics and Informatics. 2010; 8(1):43–8.
- Consonni V, Todeschini R. New similarity coefficients for binary data. Match-Communications in Mathematical and Computer Chemistry. 2012; 68(2):581.
- Hamming RW. Error detecting and error correcting codes. Bell System Technical Journal. 1950; 29(2):147–60. DOI: 10.1002/j.1538-7305.1950.tb00463.x.
- Sokal RR, Michener CD. A statistical method for evaluating systematic relationships. Science Bulletin. 1958; 38:1409– 38.