# An LCS-based string metric

Daniel Bakkelund

September 23, 2009

**Abstract**

These notes presents a string similarity measure which is a metric in the mathematical sense. In particular, the triangle inequality holds for this metric. The metric is based on the longest common subsequence (LCS) measure, and the complexity of any sensible implementation will be no worse than $O(n^2)$.

## 1 Introduction

These notes presents a string similarity measure based on longest common subsequences (LCS), which is a metric in the mathematical sense. The metric is defined as

$$d(x,y) \;=\; 1 - \frac{f(x,y)}{M(x,y)},$$

where $x$ and $y$ are finite sequences of symbols over some alphabeth $\Sigma$, $f(x,y)$ is the length of the longest common subsequence(s) of $x$ and $y$, and $M(x,y)$ is the length of the longest string of $x$ and $y$. Informally, the metric measures to which degree the shorter string differs from the longer.

### 1.1 Prior descriptions

The metric shares commonality with the Jaccard metric [14] and the normalized edit distance [8], and close, although non-metric constructions, are presented in [2, §2.1] and [7, §5]. The metric itself is previously described in [15, §2].

This text comprizes my notes in the search of a proof of the above function actually being a metric. They were written prior to my discovery of [15], and the text may bear some signs of this. However, the notes may still serve as an introduction to strings and to an understanding of longest common subsequences and shortest common supersequences. In some of the proofs, I have employed commutative diagrams with great success, and it seems to me that this tool should be used more often within this area.

## 2 Preliminaries

In this section, the theory necessary for the construction of the metric, and for the proof that the construction indeed is a metric, is presented. We first recall strings, and present three equivalent views of what strings are. Next, we introduce a type of maps between strings we call embeddings, replacing the

notions of subsequences and supersequences. And lastly, we briefly restate the definition of a metric on a set.

## 2.1 Strings

An **alphabeth** is a set of elements called **symbols**, and given an alphabeth $\Sigma$, we denote the **set of finite strings** composed by symbols from $\Sigma$ by $\Sigma^+$. If $\varepsilon$ is the empty string (consisting of no symbols), we define $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$. While $\Sigma^+$ is often referred to as the positive closure of $\Sigma$, one call $\Sigma^*$ the Kleen closure of $\Sigma$ after the late American logician Stephen Cole Kleene [5, p322].

We can view strings in different ways: Most commonly (and as in the definition above), we can see strings as finite sequences of symbols, $x = x_0x_1 \cdots x_n$, where $x_i \in \Sigma$ for $0 \leq i \leq n$. Alternatively, we may see strings as ordered multises $x = \{x_1 < x_2 < \cdots < x_n\}$, where possibly $x_i = x_j$ even though $i \neq j$. Ordered multisets are, however, nothing but ordered disjoint unions, or even more basic, sets on the form $x = \{(i, x_i)\}_{i=1}^n$, where the ordering is imposed by the natural ordering of the first component in each member of the set. We will use these different views as we see fit, and will purpously confuse them.

For $x \in \Sigma^*$, the **length of** $\boldsymbol{x}$, denoted by $|x|$, is the function $|\cdot| : \Sigma^* \longrightarrow \mathbb{N}$ mapping $x$ to the number of symbols in $x$. In particular, $|\varepsilon| = 0$. Note that $\Sigma \subset \Sigma^*$ is the subset of strings of length one.

We say that $x \in \Sigma^*$ is **maximal** with respect to a certain property if $|x|$ is maximal in the given context. That is, there is no $x'$ fulfilling the given conditions for which $|x| < |x'|$. Similarly, we say $x$ is **minimal** if $|x|$ is minimal.

$\Sigma^*$ is the free monoid (cf. [1, §I.6]) over $\Sigma$ under juxtaposition, where the juxtapose of two elements $x, y \in \Sigma^*$ is denoted by $xy$, and $\varepsilon$ serves as identity. Since $|xy| = |x| + |y|$, $|\cdot|$ is a homomorphism of monoids.

### 2.1.1 Subsequences and substrings

A **subsequence** of a string $x = x_1 \cdots x_n$ is any string obtained by removing an arbitrary number of symbols from $x$. The subsequence relation is a partial order, which we denote by $x \subset_q y$ if $x$ is a subsequence of $y$.

Given $x, y \in \Sigma^*$, $x = x_0 \cdots x_n$ and $y = y_0 \cdots y_m$, $x$ is a **substring** of $y$ if there is a $k \in \mathbb{N}$ such that $x_i = y_{i+k}$ for $0 \leq i \leq n$. We denote this relation by $x \subset_s y$. Substrings are clarily a special type of subsequences, so the substring relation also induce a partial order on $\Sigma^*$.

**Definition 2.1** *A **common subsequence** of a set of strings $S$ is a string that is a subsequence of all strings in the set, and a **longest common subsequence** is a maximal common subsequence.*

*Similarly, a **common substring** of $S$ is a string that is a substring of all strings in $S$, and a **longest common substring** is a maximal common substring.*

Note that longest common subsequences and substrings are not necessarily unique.

We denote the set of longest common subsequences of $x$ and $y$ by $LCS(x, y)$, and somewhat misleading we denote the lengths of these sequences by $|LCS(x, y)|$.

This is well defined, as all longest common subsequences must necessarily have the same length.

### 2.1.2 Supersequences and superstrings

If $x$ is a subsequence of $y$, we also say that $y$ is a **supersequence** of $x$. A natural notation for this relation is $y \supset_q x$, and this partial order is clarily equivalent to $\subset_q$ in that $x \subset_q y$ if and only if $y \supset_q x$. Opposed to the longest common subsequences, given a set of strings $S$, a **common supersequence** of $S$ is a string that is a supersequence of all the strings in $S$, and a **shortest common supersequence** is a minimal common supersequence.

We define **superstrings** in the obvious way, and just as obviously we obtain **common superstrings** and **shortest common superstrings**.

The set of shortest common supersequences of $x$ and $y$ is denoted $SCS(x, y)$, and we denote the lengths of these sequences by $|SCS(x, y)|$.

### 2.1.3 On computational complexities

The *longest common subsequence problem* [3, A4.2] is to decide whether there is a common subsequence of length at least $N$ in a set of $M$ strings over a finite alphabeth $\Sigma$, and is proven to be NP-hard. In the case of finding a longest common subsequence of two strings, the complexity of a sensible algorithm is at most $O(|x||y|)$, see e.g. [11] or the enjoyable paper [13] of Wagner and Fisher from 1974, and algorithms exists which are $O(log(|x|) \cdot log(|y|))$ [6]. The *longest common substring problem* is, however, trivially solvable in polynomial time.

On the other hand, the *shortest common supersequence problem* [3, A4.2] is to decide whether there is a common supersequence of length at most $N$ in a set of $M$ strings over a finite alphabeth $\Sigma$. This problem is also proven NP-hard, and surprisingly, the *shortest common superstring problem* is also NP-hard.

These notes will discuss the complexities of these problems no further, but there is a variety of publications treating this in detail. See e.g. one of [4, 9, 12].

### 2.1.4 Embeddings

We will now redefine the concepts of subsequences and supersequences in terms of maps between strings. The notion of embeddings of subsequences is well known (cf. [4, §1], [9, §2]). Here, we formalize the definition in terms of maps of ordered multisets, and then derive some useful properties.

**Definition 2.2** *A **string map** is a mapping from one string to another preserving both symbols and order.*

*That is, if $x = \{x_1 < \cdots < x_n\}$ and $y = \{y_1 < \cdots < y_m\}$, a string map $\alpha : x \longrightarrow y$ is a map satisfying the following properties:*

$$\alpha(x_i) = y_j \;\Rightarrow\; x_i = y_j, \tag{2.1}$$

$$\alpha(x_i) \leq \alpha(x_j) \;\Leftrightarrow\; x_i \leq x_j. \tag{2.2}$$

*An **embedding** is an injective string map, and a **sequential embedding** $j : x \longrightarrow y$ is an embedding where*

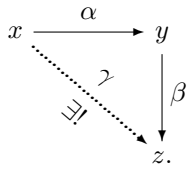
$$j(x_s) = y_t \;\Rightarrow\; j(x_{s+1}) = y_{t+1} \quad \textit{for} \quad 0 \leq s \leq n-1. \tag{2.3}$$

The **size of an embedding** $\alpha : x \longrightarrow y$ is defined as $|\alpha| = |x|$, i.e. the length of the domain. Often, an embedding like $\alpha$ will only be denoted by its domain.

It is easy to see that a subsequence $x \subset_q y$ corresponds to an embedding $x \longrightarrow y$, and that a substring $x' \subset_s y'$ in the same way corresponds to a sequential embedding $x' \longrightarrow y'$. Hence, the set of embeddings, and also the set of sequential embeddings, imposes partial orders on $\Sigma^*$.

The following lemma is a direct consequence of the transitive property of partial orders and the fact that embeddings are injective:

**Lemma 2.3** *Given two (sequential) embeddings $\alpha : x \longrightarrow y$ and $\beta : y \longrightarrow z$, there is a unique (sequential) embedding $\gamma : x \longrightarrow z$ where $\gamma = \beta \circ \alpha$. That is, we have a commutative diagram*




**Definition 2.4** *A (sequential) **xy-embedding** for $x, y \in \Sigma^*$ is a diagram of (sequential) embeddings*

$$x \longleftarrow a \longrightarrow y.$$

Clarily, an $xy$-embedding is a **common subsequence** of $x$ and $y$, and a **longest common subsequence** is a maximal $xy$-embedding. Similarly, a sequential $xy$-embedding is a **common substring**, and a **longest common substring** is a maximal sequential $xy$-embedding.

We also have the dual notion of an $xy$-coembedding:

**Definition 2.5** *A (sequential) **xy-coembedding** for $x, y \in \Sigma^*$ is a diagram of (sequential) embeddings*

$$x \longrightarrow a \longleftarrow y.$$

The **length of an *xy*-coembedding** $x \xrightarrow{\alpha} a \xleftarrow{\beta} y$ is defined as the length of $\mathrm{im}(\alpha) \cup \mathrm{im}(\beta)$. Note that this union is the ordinary union of two subsets of the string $a = \{(i, a_i)\}_{i=1}^{k}$, and it is a subsequence of $a$. Since coembeddings have lengths, it makes sense to talk about maximal and minimal coembeddings.

An $xy$-coembedding $a$ obviously corresponds to a **common supersequence** of $x$ and $y$, and a **shortest common supersequence** is a minimal $xy$-coembedding. Similarly, a sequential $xy$-coembedding is a **common superstring**, and a **shortest common superstring** is a minimal sequential $xy$-coembedding.

It turns out that the $xy$-coembeddings are just as interesting as $xy$-embeddings:

**Definition 2.6** *A **merging (sequential) xy-coembedding** $x \xrightarrow{\alpha} a \xleftarrow{\beta} y$ is a (sequential) $xy$-coembedding satisfying*

$$\mathrm{im}(\alpha) \cup \mathrm{im}(\beta) \;=\; a. \tag{2.4}$$

*A merging (sequential) coembedding will sometimes be referred to as a (sequential) **merger**. We say that $\alpha$ and $\beta$ **cover** $a$ when $a \subset \mathrm{im}(\alpha) \cup \mathrm{im}(\beta)$.*

In particular, we note that if $x \longrightarrow a \longleftarrow y$ is a merger, the length of the $xy$-coembedding is $|a|$, i.e. the length of the common codomain, and we must also have $|xy| \geq |a|$.

**Lemma 2.7** *Given an $xy$-coembedding $x \xrightarrow{\alpha} a \xleftarrow{\beta} y$, there is an $xy$-embedding $d$ for which the diagram of embeddings*

$$\begin{array}{ccc} d & \longrightarrow & y \\ \downarrow & & \downarrow \beta \\ x & \xrightarrow[\alpha]{} & a \end{array}$$

*commutes and*

$$|xy| \;\leq\; |a| + |d|.$$

*If the coembedding is a merger, then $d$ is unique, and $|xy| = |a| + |d|$. We call $d$ a **remainder** of $a$.*

**Proof** Assume first that $a$ is a merger. If $|xy| = |a|$, $d = \varepsilon$ is the unique embedding satisfying the equation, so assume $|xy| = |a| + n$ for some $n > 0$. Since $|xy| > |a|$ and both $\alpha$ and $\beta$ are injective, $\mathrm{im}(\alpha) \cap \mathrm{im}(\beta)$ must be nonempty, and it must contain exactly $n$ elements due to Definition 2.6. If $d \longrightarrow a$ is the embedding with $\mathrm{im}(d) = \mathrm{im}(\alpha) \cap \mathrm{im}(\beta)$, the diagram commutes and $|xy| = |a| + |d|$, so the required remainder $d$ exists. Now, if $d' \longrightarrow a$ is another embedding satisfying the criteria, the domains and codomains of $d$ and $d'$ coincide, and since they are order preserving monics, they must be equal.

If, on the other hand, $a$ is not a merger, we have $a \not\subset \mathrm{im}(\alpha) \cup \mathrm{im}(\beta)$. Then there is a subsequence $a' \subset_q a$ fulfilling $a' = \mathrm{im}(\alpha) \cup \mathrm{im}(\beta)$ which, according to the first part of the proof, is a merger with an $xy$-embedding $d'$ where

$$|xy| = |a'| + |d'| < |a| + |d'|.$$

And we can extend $d' \longrightarrow a'$ to an embedding $d' \longrightarrow a$ by composing with the inclusion embedding, giving us the desired diagram. □

**Lemma 2.8** *An $xy$-embedding $x \xleftarrow{\alpha} a \xrightarrow{\beta} y$ induces a merger with remainder $a$.*

**Proof** The proof is by induction: Let $|x| = n$ and $|y| = m$. Assume first $|a| = 1$ and that $\alpha(a) = x_i$ and $\beta(a) = y_j$. The string

$$b \;=\; x_1 \cdots x_{i-1} y_1 \cdots y_{j-1} a x_{i+1} \cdots x_n y_{j+1} \cdots y_m$$

is of length $|xy| - 1$ and allows for an $xy$-coembedding covering $b$, hence, it is a merger with remainder $a$.

Now, assume the hypothesis holds for embeddings of lengths up to $k-1$, and assume $|a| = k$. Let $a' \subset_s a$ be the substring where the last symbol of $a$ is removed, and let $b'$ be the merger with remainder $a'$, which exists due to the induction hypothesis. Assume $\alpha(a_{k-1}) = x_{i'}$, $\alpha(a_k) = x_i$, $\beta(a_{k-1}) = y_{j'}$ and $\beta(a_k) = y_j$. If $\tilde{b} \subset_s b'$ is the substring with the last symbol equal to $a'_{k-1}$, we compose a new string

$$b = \tilde{b} x_{i'+1} \cdots x_{i-1} y_{j'+1} \cdots y_{j-1} a_k x_{i+1} \cdots x_n y_{j+1} \cdots y_m.$$

This string is of length $|xy| - k$ and is indeed a merger with $a$ as a remainder. □

This proof also shows that the merger is not in any way unique: we may permute the $x_i$s and $y_j$s between every symbol from the remainder, as long as we maintain the respective orders of the domains.

**Lemma 2.9** *The shortest common supersequences are mergers, and their remainders are the longest common subsequences of $x$ and $y$.*

**Proof** Minimal $xy$-coembeddings are obviously mergers, so assume that $a$ is a minimal $xy$-coembedding with remainder $d$. Now, $d$ is a common subsequence of $x$ and $y$. And it must be maximal, for otherwise there would be a longer embedding $d'$ for which we would have a merger $a'$ according to Lemma 2.8 contradicting $a$ being a minimal merger since

$$|xy| = |a| + |d| = |a'| + |d'| \;\Rightarrow\; |a| > |a'|.$$

Now assume $x \longleftarrow d \longrightarrow y$ is a maximal $xy$-embedding. From Lemma 2.8, there is a merger $a$ with remainder $d$. This $a$ is minimal, for if there is a merger $a'$ with $|a'| < |a|$, then the remainder $d'$ of $a'$ must satisfy $|d'| > |d|$, and since $x \longleftarrow d' \longrightarrow y$, this means $d$ was not maximal after all, which is a contradiction. □

## 2.2 Metric

For completeness, we restate the definition of a metric on a set (cf. [10, §20]):

**Definition 2.10** *A **metric** on a set $A$ is a function $d : A \times A \longrightarrow \mathbb{R}$ satisfying the following properties:*

**M1 - *Positive definiteness:***
*For all $x, y \in A$ we have $d(x, y) \geq 0$, and $d(x, y) = 0$ if and only if $x = y$.*

**M2 - *Symmetry:***
*For all $x, y \in A$ we have $d(x, y) = d(y, x)$.*

**M3 - *Triangle inequality:***
*For all $x, y, z \in A$ we have $d(x, y) + d(y, z) \geq d(x, z)$.*

# 3 An LCS based metric on $\Sigma^*$

For notational simplicity, we introduce the two functions: $f : \Sigma^* \times \Sigma^* \longrightarrow \mathbb{N}$ defined by

$$f(x, y) \;=\; |LCS(x, y)|,$$

and $M : \Sigma^* \times \Sigma^* \longrightarrow \mathbb{N}$ defined by

$$M(x, y) \;=\; max\{|x|, |y|\}.$$

**Theorem 3.1** *The function $d : \Sigma^* \times \Sigma^* \longrightarrow \mathbb{Q}$ defined by*

$$d(x, y) \;=\; 1 - \frac{f(x, y)}{M(x, y)} \;=\; 1 - \frac{|LCS(x, y)|}{max\{|x|, |y|\}} \tag{3.1}$$

*is a metric on $\Sigma^*$. We conveniently define $d(\varepsilon, \varepsilon) = 0$.*

The theorem will be proved through the below lemmas, and as previously mentioned, the triangle inequality turns out to be the hardest property to prove.

**Lemma 3.2** *Function* (3.1) *is positive definite and symmetric.*

**Proof** First, note that $d(x,x) = 1 - \frac{f(x,x)}{M(x,x)} = 1 - \frac{|x|}{|x|} = 0$. If, on the other hand, $d(x,y) = 0$, this means $\frac{f(x,y)}{M(x,y)} = 1$, implying $f(x,y) = M(x,y)$. But since $f(x,y) \leq M(x,y)$, this means that $x = y$. Also since $f(x,y) \leq M(x,y)$, we have $d(x,y) \geq 0$, and lastly, symmetry follows from both $f$ and $M$ being symmetric. □

What remains is to prove the triangle inequality. The below lemma describes an important relation between the lengths of the longest common subsequences of three strings:

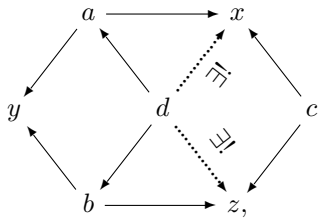
**Lemma 3.3** *The following relation holds for all* $x, y, z \in \Sigma^*$*:*

$$f(x,y) + f(y,z) - f(x,z) \ \leq \ |y|.$$

**Proof** Assume contrarily that $|y| < f(x,y)+f(y,z)-f(x,z)$ for some $y \in \Sigma^*$. The three lengths on the right hand side corresponds to three maximal diagrams of embeddings

$$x \longleftarrow a \longrightarrow y, \qquad y \longleftarrow b \longrightarrow z \qquad \text{and} \qquad x \longleftarrow c \longrightarrow z, \tag{3.2}$$

where $f(x,y) = |a|$, $f(y,z) = |b|$ and $f(x,z) = |c|$. We can therefore rewrite the assumption as $|y| < |a| + |b| - |c|$, which is the same as $|ab| > |y| + |c|$. We have an $ab$-coembedding $a \longrightarrow y \longleftarrow b$, and according to Lemma 2.7 this coembedding has a remainder $d$ for which $|ab| \leq |y| + |d|$, implying $|d| > |c|$. This gives us a commutative diagram of embeddings




but as indicated by the dotted arrows, which are due to Lemma 2.3, we have a diagram of embeddings $x \longleftarrow d \longrightarrow z$, contradicting $c$ being a maximal $xz$-embedding. Thus the assumption that $|y| < f(x,y) + f(y,z) - f(x,z)$ must be wrong. □

**Corollary 3.4** *The triangle inequality holds for* (3.1).

**Proof** Due to symmetry of (3.1), we can safely assume that $|x| \leq |z|$, which implies $M(x,y) \leq M(y,z)$. Thus,

$$M(x,y)(f(y,z) - f(x,z)) \leq M(y,z)(|y| - f(x,y)),$$

since $f(y,z) - f(x,z) \leq |y| - f(x,y)$ due to Lemma 3.3. And since we have $M(x,z) \leq M(y,z)$, this means that

$$\begin{aligned} f(x,y)M(y,z) + f(y,z)M(x,y) &\leq |y|M(y,z) + M(x,y)f(x,z) \\ &\leq M(x,y)M(y,z) + \frac{M(y,z)}{M(x,z)}M(x,y)f(x,z). \end{aligned}$$

Dividing both sides by $M(x,y)M(y,z)$ gives

$$\begin{aligned} \frac{f(x,y)}{M(x,y)} + \frac{f(y,z)}{M(y,z)} &\leq 1 + \frac{f(x,z)}{M(x,z)} \qquad \Leftrightarrow \\ 1 - \frac{f(x,z)}{M(x,z)} &\leq 1 - \frac{f(x,y)}{M(x,y)} + 1 - \frac{f(y,z)}{M(y,z)} \qquad \Leftrightarrow \\ d(x,z) &\leq d(x,y) + d(y,z). \end{aligned}$$

The remainder of the proof, that the triangle inequality holds if $x$ and $y$ are both the empty string, is left to the reader. □

### 3.1 Expressing $d$ in terms of shortest common supersequences

We introduce two new functions, again to ease notation: $g : \Sigma^* \times \Sigma^* \longrightarrow \mathbb{N}$ by

$$g(x,y) = |SCS(x,y)|,$$

and $m : \Sigma^* \times \Sigma^* \longrightarrow \mathbb{N}$ by

$$m(x,y) = min\{|x|, |y|\}.$$

Based on the observation $|xy| = g(x,y) + f(x,y)$, due to Lemmas 2.7 and 2.9, we may rewrite (3.1) as follows:

$$d(x,y) = \frac{g(x,y) - m(x,y)}{M(x,y)}. \tag{3.3}$$

Writing the metric on this form tells us that it is not only the length of the longer string that influences the result.

We close with an observation regarding shortest common supersequences mathcing the result of Lemma 3.3. The proof is another easy application of the observation $|xy| = g(x,y) + f(x,y)$:

**Lemma 3.5** *For $x, y, z \in \Sigma^*$ we have*

$$g(x,y) + g(y,z) - g(x,z) \geq |y|.$$

## Acknowledgements

Thanks to Tron Omland for supplying a shorter proof of Corollary 3.4 than my original one.

The commutative diagrams in this text were set with Payl Taylor's "diagrams" package.

# References


[1] Claude Chevalley. *Fundamental Concepts of Algebra*. Academic Press Inc, 1956.

[2] Natalia Elita, Monica Gavrila, and Vertan Cristina. Experiments with String Similarity Measures in the EBMT Framework. In *Proceedings of the RANLP 2007 Conference*, September 2007.

[3] M. R. Garey and D. S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W. H. Freeman, 1979.

[4] Ronald I. Greenberg. Bounds on the Number of Longest Common Subsequences. *arXiv:cs/0301030v2* [*cs.DM*], August 2003.

[5] Ralph P. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. Addison-Wesley Longman Publishing Co., Inc., 3rd edition, 1994.

[6] Kim S. Larsen. Length of Maximal Common Subsequences, 1992. Available at http://www.daimi.au.dk/PB/426/PB-426.pdf.

[7] Mark Last, Abraham Kandel, and Horst Bunke. *Data Mining In Time Series Databases (Series in Machine Perception and Artificial Intelligence)*. World Scientific Pub Co Inc, 2004.

[8] A. Marzal and E. Vidal. Computation of normalized edit distances and applications. *IEEE Transactions on Pattern Analysis and Applications*, 15(9):926–932, September 1993.

[9] Martin Middendorf and David F. Manlove. Combined super-/substring and super-/subsequence problems. *Theor. Comput. Sci.*, 320(2-3):247–267, 2004.

[10] James R. Munkres. *Topology*. Prentice Hall, second edition, 2000.

[11] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3):443–453, March 1970.

[12] V. G. Timkovskii. Complexity of common subsequence and supersequence problems and related problems. *Cybernetics*, 25(5):565–580, September 1989.

[13] Robert A. Wagner and Michael J. Fischer. The string-to-string correction problem. *J. ACM*, 21(1):168–173, 1974.

[14] G. A. Watson. An algorithm for the single facility location problem using the Jaccard metric. *j-SIAM-J-SCI-STAT-COMP*, 4(4):748–756, December 1983.

[15] Li Zhao, Sung Sam Yuan, Sun Peng, and Tok Wang Ling. A New Efficient Data Cleansing Method. In *DEXA '02: Proceedings of the 13th International Conference on Database and Expert Systems Applications*, pages 484–493, London, UK, 2002. Springer-Verlag.