




When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

Dana Fisman   

Department of Computer Science, Ben-Gurion University, Beer-Sheva, Israel

Ilay Tzarfati  

Department of Computer Science, Ben-Gurion University, Beer-Sheva, Israel

Abstract

The well known Normalized Edit Distance (NED) [Marzal and Vidal 1993] is known to disobey the triangle inequality on contrived weight functions, while in practice it often exhibits a triangular behavior. Let d be a weight function on basic edit operations, and let NED_d be the resulting normalized edit distance. The question what criteria should d satisfy for NED_d to be a metric is long standing. It was recently shown that when d is the uniform weight function (all operations cost 1 except for no-op which costs 0) then NED_d is a metric. The question regarding non-uniform weights remained open. In this paper we answer this question by providing a necessary and sufficient condition on d under which NED_d is a metric.

2012 ACM Subject Classification Theory of computation \rightarrow Pattern matching; Theory of computation \rightarrow Formal languages and automata theory

Keywords and phrases Normalized Edit Distance, Non-uniform Weights, Triangle Inequality, Metric

Digital Object Identifier 10.4230/LIPIcs.CPM.2024.14

Funding *Ilay Tzarfati*: Supported by ISF grant 2507/21 and Frankel Center for Computer Science, BGU.

Acknowledgements We would like to thank Oded Margalit, Elina Sudit and Sandra Zilles for comments on an earlier draft of this paper.

1 Introduction

The question of quantifying the similarity between two strings is quite ancient [9, 11, 18, 17, 10, 19, 14]. A typical way to measure the distance between two strings, is the Levenshtein distance, aka, *edit distance* (ED) [11]. The edit distance between two strings $w_1, w_2 \in \Sigma^*$ is measured as the minimum *weight* of an *edit path* – a sequence of edit operations *delete*, *insert*, *replace*, or *no-op* – required to transform w_1 to w_2 . In the case of uniform weights, all edit operations cost 1 except for *no-op* which costs 0. For example, $ED(\text{Jane}, \text{John}) = 3$ since we can transform the string *Jane* to *John* using the edit path $\alpha = \text{no-op}(J), \text{replace}(a,o), \text{replace}(n,h), \text{replace}(e,n)$ which weighs 3 and there is no edit path transforming *Jane* to *John* that weighs less than 3. In many settings, a normalized version of the edit distance is required. To see why, note that for the same argument as above the distance between *JaneKennedy* and *JohnKennedy* is also 3 although clearly the latter pair of strings are much more similar to one another.

In [13] the well-known normalized version of the edit distance, henceforth NED, was suggested in which the distance between two non-empty strings w_1 and w_2 is the minimum *cost* of an edit path between w_1 and w_2 . The *cost* of an edit path is the weight of edit operations along the path, divided by the length of the path. The cost of the edit path α above is thus $\frac{3}{4}$ but since $\alpha' = \text{no-op}(J), \text{replace}(a,o), \text{insert}(h), \text{no-op}(n), \text{delete}(e)$ also transforms *Jane* to *John* we have that $NED(\text{Jane}, \text{John}) = \frac{3}{5}$. Similar arguments show that $NED(\text{JaneKennedy}, \text{JohnKennedy}) = \frac{3}{12}$, thus it is now apparent that *JaneKennedy* and *JohnKennedy* are more similar to one another (compared to *Jane* and *John*).



© Dana Fisman and Ilay Tzarfati;
licensed under Creative Commons License CC-BY 4.0

35th Annual Symposium on Combinatorial Pattern Matching (CPM 2024).

Editors: Shunsuke Inenaga and Simon J. Puglisi; Article No. 14; pp. 14:1–14:17

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The above discussion concerned the case of uniform weights. However, in many applications using a normalized edit distance, such as text retrieval, signal processing, and computational biology, non-uniform weights are used. In the case of non-uniform weights, any basic edit operation has its own weight, e.g. $delete(a)$, $insert(a)$, $delete(b)$, $replace(a, b)$, etc. can cost differently. A function d assigning a weight to each edit operation is assumed, and each such function gives rise to a different version, NED_d , of the normalized edit distance of [13]. It was noted in [13] that NED_d may not satisfy the triangle inequality for certain weight functions d , though it is observed to behave well in practice often enough. The question under which criteria on d is NED_d a metric is standing since. This motivated the introduction of other definitions of normalized edit distance, e.g., the generalized edit distance (GED) [12], and the contextual edit distance (CED) [5]. A sufficient condition on d for GED_d to be a metric was given in [12]¹ and in [5] it is shown that CED is a metric when d is the uniform weight. It was recently shown that under the uniform weights, NED is a metric, and that NED enjoys several nice properties that GED and CED do not [7]. The question under which criteria NED_d over a non-uniform weight function d is a metric was left unanswered.

In this paper we provide a necessary and sufficient condition on a weight function d on edit operations, in order for NED_d to be a metric. While it is reasonable to assume that d should be a metric (in the space of edit operations) we show that this is neither a necessary condition, nor a sufficient one. The exact criteria relaxes the requirement of the triangle inequality, makes an additional requirement on the cost of inserts and deletes, and in general concerns only edit operations we term *essentials*. We term d that satisfies these criteria *fine*. The proof that d being fine is also a sufficient condition for NED_d to be a metric generalizes and significantly simplifies the proof that NED_d is a metric in the uniform case [7].

The main result of the paper is the following theorem.

► **Theorem 1 (Necessary and Sufficient Condition).** *Let $d : (\Sigma \cup \{\varepsilon\}) \times (\Sigma \cup \{\varepsilon\}) \rightarrow [0, 1]$. Let $a, c \in \Sigma \cup \{\varepsilon\}$ and $b \in \Sigma$. Let $m = \sup\{NED_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\}$. A necessary and sufficient condition for NED_d to be a metric is that d satisfies the following properties after removing inessential edit operations.*

1. $d(a, c) = 0$ iff $a = c$
2. $d(a, c) = d(c, a)$
3. $d(a, b) + d(b, c) \geq \min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\}$
4. $d(\varepsilon, b) = d(b, \varepsilon) \geq \frac{m}{2}$

The rest of the paper is organized as follows. We provide some preliminaries in §2. In §3 we show that d being a metric is neither a necessary condition nor a sufficient one. In §4 we gradually develop the necessary condition on d , we term a weight function d satisfying these conditions *fine*. In §5 we show that d being fine is a sufficient condition for NED_d to be a metric. In §6 we provide some natural examples for weight functions that are fine, and discuss applications of NED_d in formal verification. Due to lack of space, the proofs regarding the examples in §6 are deferred to the full version of the paper.

2 Notations

Metric spaces

A metric space is an ordered pair (\mathbb{M}, d) where \mathbb{M} is a set and $d : \mathbb{M} \times \mathbb{M} \rightarrow \mathbb{R}$ is a *metric*, i.e., it satisfies the following properties for all $m_1, m_2, m_3 \in \mathbb{M}$:

¹ The condition is that d is a metric and all delete and insert operations cost the same.

1. $d(m_1, m_2) = 0$ iff $m_1 = m_2$;
2. $d(m_1, m_2) = d(m_2, m_1)$;
3. $d(m_1, m_3) \leq d(m_1, m_2) + d(m_2, m_3)$.

The first condition is referred to as *identity of indiscernibles*, the second as *symmetry*, and the third as the *triangle inequality*.

Words and Edit Operations

Let Σ be an alphabet and Σ^* (Σ^+) denote all the finite (non-empty) strings over Σ . The length of word $w = \sigma_1\sigma_2\dots\sigma_n$, denoted $|w|$, is n . We use $w[i]$ to denote the i -th letter of w , and $w[..i]$ for the prefix of w ending at the i -th letter. We denote the empty word by ε . Let $a, b \in \Sigma$. The usual edit operations are *delete a*, *insert a*, *replace a with b*, and *no-op a*. We use the following notations for them. Let $\hat{\Gamma} = (\Sigma \cup \{\varepsilon\})^2$. We use $\begin{bmatrix} a \\ b \end{bmatrix}$ to represent the pair (a, b) . An *edit operation* is a letter in $\Gamma = \hat{\Gamma} \setminus \{\begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}\}$. The letter $\begin{bmatrix} a \\ b \end{bmatrix}$ denotes *replace a with b*, the letter $\begin{bmatrix} a \\ a \end{bmatrix}$ denotes *no-op a*, the letter $\begin{bmatrix} a \\ \varepsilon \end{bmatrix}$ denote *delete a*, and the letter $\begin{bmatrix} \varepsilon \\ a \end{bmatrix}$ denotes *insert a*. This style of notation will come in handy in §5 when we prove the sufficient condition.

Edit Paths

An *edit path* between words w_1 and w_2 over Σ is a sequence $\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \dots \begin{bmatrix} a_m \\ b_m \end{bmatrix}$ of elements in Γ satisfying that $a_1a_2\dots a_m = w_1$ and $b_1b_2\dots b_m = w_2$. For instance, take $w_1 = aaa$ and $w_2 = bb$ then $\alpha = \begin{bmatrix} a \\ b \end{bmatrix} \begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$ is an edit path between w_1 and w_2 , and $\alpha' = \begin{bmatrix} \varepsilon \\ b \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} a \\ \varepsilon \end{bmatrix}$ is another edit path between w_1 and w_2 . It is sometimes convenient to represent these edit paths as $aaa \mapsto b_b$ and $_aaa \mapsto bb_$, respectively. In standard terminology the first edit path would correspond to *replace a with b*, *delete a*, *replace a with b* and the second to *insert b*, *replace a with b*, *delete a*, *delete a*. In §5 we use also strings over $\hat{\Gamma}$ which we refer to as *extended edit-paths*.

We use π_i for the projection of a tuple or a sequence of tuples on its i -th component. E.g. if $\alpha = \begin{bmatrix} \varepsilon \\ b \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} a \\ \varepsilon \end{bmatrix}$ then $\pi_1(\alpha) = \varepsilon aaa$ and $\pi_2(\alpha) = bb\varepsilon\varepsilon$. Let $\alpha = a_1a_2\dots a_k$ be a sequence of symbols over $\Sigma \cup \{\varepsilon\}$. We use $\text{word}(\alpha)$ for the word obtained by concatenating these letters. For instance, $\text{word}(\pi_1(\alpha)) = aaa$. Let $w_1, w_2 \in \Sigma^*$. Let α be an (extended) edit path between w_1, w_2 . Then $\text{word}(\pi_1(\alpha)) = w_1$ and $\text{word}(\pi_2(\alpha)) = w_2$. We use $\text{input}(\alpha)$ for $\text{word}(\pi_1(\alpha))$ and $\text{output}(\alpha)$ for $\text{word}(\pi_2(\alpha))$.

Weight, Length and Costs of Edit Paths

Let $d : \Gamma \rightarrow [0, 1]$ be a function assigning cost for the basic edit operations. Although $\begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$ is not an edit operation, it is sometimes convenient to assume d is defined on it as well, namely $d : \hat{\Gamma} \rightarrow [0, 1]$. When this is the case we simply assume $d(\varepsilon, \varepsilon) = 0$. Let $w_1 \in \Sigma^*$ and $w_2 \in \Sigma^+$. Let $\alpha = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \dots \begin{bmatrix} a_n \\ b_n \end{bmatrix}$ be an edit path between w_1 and w_2 . We say that the *length* of α is n , and that the *weight* of α is $\sum_{i=1}^n d(a_i, b_i)$. We denote them by $\text{len}(\alpha)$ and $\text{wgt}(\alpha)$ respectively. The cost of an edit path α , denoted $\text{cost}(\alpha)$ is defined as $\frac{\text{wgt}(\alpha)}{\text{len}(\alpha)}$.²

² Note that it is well defined since $w_2 \in \Sigma^+$ guarantees that $\text{len}(\alpha) \neq 0$. To obtain a definition that works also for $w_1 \in \Sigma^+$ and $w_2 \in \Sigma^*$ we can consider also edit paths from w_2 to w_1 .

The Normalized Edit Distance (NED)

Let Σ be an alphabet and $d : \Gamma \rightarrow [0, 1]$. Note that d may depend on the exact letters, and it could be that e.g. $d(a, b) \neq d(b, a)$ or that $d(\varepsilon, b) \neq d(\varepsilon, a)$. The Levenstein distance [11] (ED) and the Normalized Edit Distance [13] (NED) between w_1 and w_2 (with respect to d) can be defined as follows:

$$ED_d(w_1, w_2) = \min \{ \text{wgt}(\alpha) : \alpha \text{ is an edit path between } w_1 \text{ and } w_2 \}.$$

$$NED_d(w_1, w_2) = \min \{ \text{cost}(\alpha) : \alpha \text{ is an edit path between } w_1 \text{ and } w_2 \}.$$

The ED_d distance looks for an edit path with minimum weight, whereas NED_d looks for an edit path with minimum cost.

► **Example 2** (ED_d and NED_d). For instance, consider the words $w_1 = abaad$, $w_2 = baaadc$ over $\Sigma = \{a, b, c, d\}$. Then both $\alpha_1 = [\varepsilon][b][a][a][\varepsilon][d][\varepsilon]$ and $\alpha_2 = [b][b][a][a][d][\varepsilon]$ are edit paths between w_1 and w_2 . Consider first the setting of uniform weights, namely $d : \Gamma \rightarrow [0, 1]$ is defined as $d(\sigma, \sigma) = 0$ and $d(\sigma, \sigma') = 1$ if $\sigma \neq \sigma'$. In this setting, we have that $\text{wgt}(\alpha_1) = 1 + 0 + 0 + 0 + 1 + 0 + 1 = 3$ and $\text{wgt}(\alpha_2) = 1 + 1 + 0 + 0 + 0 + 1 = 3$, so using ED α_1 and α_2 are equally good. However $\text{len}(\alpha_1) = 7$ and $\text{len}(\alpha_2) = 6$ so $\text{cost}(\alpha_1) = \frac{3}{7}$ and $\text{cost}(\alpha_2) = \frac{3}{6}$ thus using NED, α_1 is preferable.

Consider now the non-uniform weights $d(\sigma, \sigma') = 0.5$ for every $\sigma \neq \sigma'$, and $d(\sigma, \sigma) = 0$, $d(\sigma, \varepsilon) = d(\varepsilon, \sigma) = 1$ for every $\sigma, \sigma' \in \Sigma$. We get that $\text{wgt}(\alpha_1) = 1 + 0 + 0 + 0 + 1 + 0 + 1 = 3$ and $\text{wgt}(\alpha_2) = 0.5 + 0.5 + 0 + 0 + 0 + 1 = 2$ and so $\text{cost}(\alpha_1) = \frac{3}{7}$ and $\text{cost}(\alpha_2) = \frac{2}{6}$, thus α_2 is preferable.

► **Definition 3.** An edit path α is termed optimal if $\text{cost}(\alpha) = NED_d(\text{input}(\alpha), \text{output}(\alpha))$.

3 A metric weight function is neither necessary nor sufficient

Let $d : \Gamma \rightarrow [0, 1]$ we are interested in finding a necessary and sufficient condition on d for NED_d to be a metric. A reasonable conjecture is that d is a metric on the space Γ . We show that this is neither a sufficient nor a necessary condition.

We first show that d being a metric is not a sufficient condition for NED_d to be a metric.

► **Claim 4.** There exists d which is a metric while NED_d is not.

Proof. Let $\Sigma = \{a, b\}$ and let $d(\sigma, \sigma) = 0$ for every $\sigma \in \Sigma$. Let $d(a, b) = d(b, a) = 1$, $d(a, \varepsilon) = d(\varepsilon, a) = 0.1$ and $d(b, \varepsilon) = d(\varepsilon, b) = 1$. It is easy to verify that d is a metric.

We show now that NED_d breaks the triangle inequality. Take $w_1 = a$ and $w_3 = b$. Then $NED_d(a, b) = 0.55$ via the edit path that deletes a and inserts b namely $[\varepsilon][b]$. Its weight is $0.1 + 1$ and its lengths is 2. Thus it costs $\frac{1.1}{2} = 0.55$.

Consider now going via $w_2 = baaaa$. Then $\alpha_{1,2} = [a][\varepsilon][a][\varepsilon][\varepsilon]$ is an edit path between w_1 and w_2 and $\alpha_{2,3} = [b][\varepsilon][\varepsilon][\varepsilon][\varepsilon]$ is an edit path between w_2 and w_3 . Notice that $NED_d(w_1, w_2) \leq \frac{1+4(0.1)}{5}$ and $NED_d(w_2, w_3) \leq \frac{4(0.1)}{5}$. Thus, $NED_d(w_1, w_2) + NED_d(w_2, w_3) \leq \frac{1.8}{5} = 0.36 < 0.55 = NED_d(w_1, w_3)$. Hence, the triangle inequality for NED_d breaks. ◁

► **Corollary 5.** d being a metric is not a sufficient condition for NED_d to be a metric.

Next we show that d being a metric is not a necessary condition for NED_d to be a metric: NED_d can be a metric although d breaks the triangle inequality or the symmetry condition.

► **Claim 6.** There exists NED_d which is a metric while d breaks the triangle inequality.

Proof. Let $\Sigma = \{a, b\}$ and let $d(\sigma, \sigma) = 0$ for every $\sigma \in \Sigma$. Let $d(a, b) = d(b, a) = 1$, $d(\varepsilon, a) = d(a, \varepsilon) = 0.4$ and $d(\varepsilon, b) = d(b, \varepsilon) = 0.5$. Then d is not a metric since going from a to b via ε is less costly than going directly (0.9 vs. 1). However, NED_d is a metric. It is easy to see that the first two requirements of a metric hold for NED_d . Regarding the triangle inequality, while it seems at first that it breaks in going from a to b (as it does for d) this is not the case. The optimal edit path from a to b is $\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ b \end{bmatrix}$ whose cost is $\frac{0.5+0.4}{2} = 0.45$ which is smaller than going via ε which costs $\text{NED}_d(a, \varepsilon) + \text{NED}_d(\varepsilon, b) = 0.4 + 0.5$. The proof that NED_d is a metric follows from the fact that it adheres to the sufficient and necessary conditions we provide. We come back to this in Remark 33. \triangleleft

▷ **Claim 7.** There exists NED_d which is a metric while d breaks symmetry.

Proof. Let $\Sigma = \{a, b\}$ and let $d(\sigma, \sigma) = 0$ for every $\sigma \in \Sigma$. Let $d(a, b) = 1$, $d(b, a) = 0.9$, $d(\varepsilon, a) = d(a, \varepsilon) = 0.4$ and $d(\varepsilon, b) = d(b, \varepsilon) = 0.45$. Then d is not a metric since $d(a, b) \neq d(b, a)$ breaks symmetry. However, NED_d is a metric. Indeed, the symmetry of NED_d does not break since it never uses in an optimal path the operation $\begin{bmatrix} a \\ b \end{bmatrix}$ or $\begin{bmatrix} b \\ a \end{bmatrix}$. For example, consider $w_1 = a$ and $w_2 = b$. Then $\text{NED}_d(w_1, w_2) = \text{wgt}(\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ b \end{bmatrix})/2 = 0.425$ and $\text{NED}_d(w_2, w_1) = \text{wgt}(\begin{bmatrix} b \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ a \end{bmatrix})/2 = 0.425$ (and the fact that $d(b, a) = 0.9 \neq d(a, b) = 1$ doesn't come in the way). Here as well the proof that NED_d is a metric is deferred to Remark 33. \triangleleft

► **Corollary 8.** d being a metric is not a necessary condition for NED_d to be a metric.

4 Necessary condition

We turn to extract necessary conditions on d for NED_d to be a metric. We start by showing that, as expected, if NED_d is a metric then d satisfies the first requirement of a metric. The proof relies on the following simple observation.

▷ **Claim 9.** Let $a, b \in \Sigma$. Then

1. $\text{NED}_d(a, \varepsilon) = d(a, \varepsilon)$ and $\text{NED}_d(\varepsilon, a) = d(\varepsilon, a)$
2. $\text{NED}_d(a, b) = \min\{d(a, b), \frac{1}{2}(d(a, \varepsilon) + d(\varepsilon, b))\}$

Proof. The first item holds since there is a single edit path from ε to $a \in \Sigma$: the edit path $\begin{bmatrix} \varepsilon \\ a \end{bmatrix}$. Hence $\text{NED}_d(a, \varepsilon) = \frac{d(a, \varepsilon)}{1}$. The claim on $\text{NED}_d(\varepsilon, a)$ is symmetric.

The second item holds since there are exactly two edit paths from a to b : either $\begin{bmatrix} a \\ b \end{bmatrix}$ or $\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ b \end{bmatrix}$. Thus $\text{NED}(a, b) = \min\{\frac{d(a, b)}{1}, \frac{d(a, \varepsilon) + d(\varepsilon, b)}{2}\}$. \triangleleft

▷ **Claim 10.** If NED_d is a metric then d must satisfy the identity of indiscernibles condition.

Proof. Let $a, b \in \Sigma$. By Claim 9, $\text{NED}_d(a, \varepsilon) = d(a, \varepsilon)$ and $\text{NED}_d(\varepsilon, a) = d(\varepsilon, a)$. Thus, $\text{NED}_d(a, \varepsilon) > 0$ implies $d(a, \varepsilon) > 0$. By symmetry we get $d(\varepsilon, a) > 0$. Consider now the case where $b = a$. We have $0 = \text{NED}_d(a, a) = \min\{d(a, a), \frac{1}{2}(d(a, \varepsilon) + d(\varepsilon, a))\}$. Since we have shown that the second argument is non-zero it follows that $d(a, a) = 0$. Last, consider the case where $b \neq \varepsilon$ and $b \neq a$. Assume towards contradiction the replace between some non-identical letters a and b is zero, then $\text{NED}_d(a, b) \leq 0$ via the direct path involving this replace contradicting that NED_d satisfies the first requirement of a metric. \triangleleft

The proof of Claim 6 shows that NED_d can satisfy the condition of triangle inequality although d does not. The reason is that in NED_d there are two options for a direct path between two letters a and b : either a replace or a delete followed by an insert. In the perspective of d a

14:6 When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

path from a to b that takes a short detour via ε is not direct. Hence the triangle inequality for d can be relaxed as stated below and as we show in the next section this relaxation suffices.

▷ **Claim 11 (Relaxed Triangle Inequality).** If NED_d satisfies the triangle inequality then d should satisfy

$$d(a, b) + d(b, c) \geq \min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\}$$

for all $b \in \Sigma$ and $a, c \in \Sigma \cup \{\varepsilon\}$.

► **Remark 12.** Note that when $c = \varepsilon$ the requirement is $d(a, b) + d(b, \varepsilon) \geq \min\{d(a, \varepsilon), d(a, \varepsilon) + d(\varepsilon, \varepsilon)\}$ and given $d(\varepsilon, \varepsilon) = 0$ this amounts to

$$d(a, b) + d(b, \varepsilon) \geq d(a, \varepsilon)$$

which says that replacing and deleting cannot cost less than deleting. Similarly, when $a = \varepsilon$ this amounts to $d(\varepsilon, b) + d(b, c) \geq d(b, \varepsilon)$ which says that inserting and replacing cannot cost less than inserting.

Proof of Claim 11. We first consider the case that $c = \varepsilon$. Following Remark 12, assume towards contradiction that there exists $a, b \in \Sigma$ such that $d(a, b) + d(b, \varepsilon) < d(a, \varepsilon)$. Consider $w_1 = a$, $w_2 = b$, $w_3 = \varepsilon$. Let $\alpha_{1,3} = [\frac{a}{\varepsilon}]$. Notice that it is the only possible edit path from w_1 to w_3 and thus the optimal. Let $\alpha_{1,2} = [\frac{a}{b}]$ and $\alpha_{2,3} = [\frac{b}{\varepsilon}]$. Hence $\text{cost}(\alpha_{1,2}) = d(a, b)$, $\text{cost}(\alpha_{2,3}) = d(b, \varepsilon)$ and $\text{cost}(\alpha_{1,3}) = d(a, \varepsilon)$. Since NED_d satisfies the triangle inequality then we know that $\text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) \geq \text{cost}(\alpha_{1,3})$ hence $d(a, b) + d(b, \varepsilon) \geq d(a, \varepsilon)$ in contradiction to the assumption. The case where $a = \varepsilon$ is similar.

Assume now neither a nor c is ε and assume towards contradiction that there exists $a, b, c \in \Sigma$ such that $d(a, b) + d(b, c) < \min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\}$. Let $i \in \mathbb{N}$ and consider $w_1 = a^{i+1}$, $w_2 = a^i b$ and $w_3 = a^i c$. Let $\alpha_{1,3} \in \Gamma^*$ be an optimal edit path between w_1 to w_3 . Notice that either $\alpha_{1,3} = ([\frac{a}{c}]^i [\frac{a}{c}])$ or $\alpha_{1,3} = ([\frac{a}{\varepsilon}]^i [\frac{a}{\varepsilon}] [\frac{\varepsilon}{c}])$. Consider the two edit paths $\alpha_{1,2} = ([\frac{a}{b}]^i [\frac{a}{b}])$ and $\alpha_{2,3} = ([\frac{a}{c}]^i [\frac{a}{c}])$ between w_1 to w_2 and between w_2 to w_3 , respectively.

■ **Case 1:** $\alpha_{1,3} = ([\frac{a}{c}]^i [\frac{a}{c}])$.

Then $\text{wgt}(\alpha_{1,3}) = d(a, c)$, $\text{len}(\alpha_{1,3}) = i + 1$ and $\text{cost}(\alpha_{1,3}) = \frac{d(a,c)}{i+1}$. In order for NED_d to satisfy the triangle inequality $\text{cost}(\alpha_{1,3}) \leq \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3})$ must hold. Thus

$$\frac{d(a, c)}{i + 1} \leq \frac{d(a, b)}{i + 1} + \frac{d(b, c)}{i + 1}$$

$$d(a, c) \leq d(a, b) + d(b, c)$$

in contradiction to the assumption.

■ **Case 2:** $\alpha_{1,3} = ([\frac{a}{\varepsilon}]^i [\frac{a}{\varepsilon}] [\frac{\varepsilon}{c}])$.

Then $\text{wgt}(\alpha_{1,3}) = d(a, \varepsilon) + d(\varepsilon, c)$, $\text{len}(\alpha_{1,3}) = i + 2$ and $\text{cost}(\alpha_{1,3}) = \frac{d(a,\varepsilon)+d(\varepsilon,c)}{i+2}$. In order for NED_d to satisfy the triangle inequality $\text{cost}(\alpha_{1,3}) \leq \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3})$ must hold. Thus,

$$\frac{d(a, \varepsilon) + d(\varepsilon, c)}{i + 2} \leq \frac{d(a, b)}{i + 1} + \frac{d(b, c)}{i + 1}$$

$$\frac{(i + 1)(d(a, \varepsilon) + d(\varepsilon, c))}{i + 2} \leq d(a, b) + d(b, c)$$

By taking i to infinity we get that

$$\lim_{i \rightarrow \infty} \frac{(i+1)(d(a, \varepsilon) + d(\varepsilon, c))}{i+2} = d(a, \varepsilon) + d(\varepsilon, c) \leq d(a, b) + d(b, c)$$

in contradiction to the assumption.

Hence either way we get that for NED_d to satisfy the triangle inequality then d should satisfy

$$d(a, b) + d(b, c) \geq \min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\}$$

for every $b \in \Sigma$ and $a, c \in \Sigma \cup \{\varepsilon\}$. \triangleleft

In Claim 4 we have shown that NED_d fails to be a metric although d is. Intuitively, the reason is that going through more and more insert and delete operations can decrease the overall cost. In the following, we will show that requiring insert and delete operations to be at least half of the costliest replace operation prevents this.

\triangleright **Claim 13 (At least half).** If NED_d is a metric and $m = \sup\{\text{NED}_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\}$. Then d should satisfy the requirement $d(\varepsilon, b) = d(b, \varepsilon) \geq \frac{m}{2}$ for every $b \in \Sigma$.

Proof. First note that if $m = \sup\{\text{NED}_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\}$ then $m = \sup\{\text{NED}_d(\sigma_1, \sigma_2) : \sigma_1, \sigma_2 \in \Sigma\}$. Indeed the way to obtain the maximum cost of an edit path is using the edit operation with maximal cost, and using more than one such operation will not increase the total cost.

Suppose inserting/deleting some letter b costs c for some $c < \frac{m}{2}$. Consider the words $w_1 = \sigma_1$, $w_3 = \sigma_3$ and assume that w_1 and w_3 are such that $\text{NED}_d(w_1, w_3) = m$. Note that there are only two possible edit paths that transform w_1 to w_3 . That is, either $\alpha_{1,3} = [\sigma_1]$ or $\alpha_{1,3} = [\frac{\sigma_1}{\varepsilon}] [\frac{\varepsilon}{\sigma_3}]$. Hence $\text{NED}_d(w_1, w_3) = m$ implies $m = \min\{d(\sigma_1, \sigma_3), \frac{1}{2}(d(\sigma_1, \varepsilon) + d(\varepsilon, \sigma_3))\}$. This in turn implies that $d(\sigma_1, \sigma_3) \geq m$ and $d(\sigma_1, \varepsilon) + d(\varepsilon, \sigma_3) \geq 2m$.

Consider now the word $w_2 = \sigma_3 \cdot b^k$ for some $k \in \mathbb{N}$ where $k \geq 1$. Then we can transform w_1 to w_2 using the edit path $\alpha_{1,2} = [\sigma_1] \cdot ([\frac{\varepsilon}{b}])^k$ or $\alpha_{1,2} = [\frac{\sigma_1}{\varepsilon}] [\frac{\varepsilon}{\sigma_3}] \cdot ([\frac{\varepsilon}{b}])^k$. To transform w_2 to w_3 we can use the edit path $\alpha_{2,3} = [\frac{\sigma_3}{\varepsilon}] ([\frac{b}{\varepsilon}])^k$. Then the sum of the edit paths is one of the following:

1. In case of $\alpha_{1,2} = [\frac{\sigma_1}{\varepsilon}] \cdot ([\frac{\varepsilon}{b}])^k$:

$$\text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) = \frac{d(\sigma_1, \sigma_3) + k \cdot d(\varepsilon, b)}{1+k} + \frac{k \cdot d(b, \varepsilon)}{1+k} \geq \frac{m + k \cdot c}{1+k} + \frac{k \cdot c}{1+k} = \frac{2k \cdot c + m}{1+k}$$

Since $\text{NED}_d(w_1, w_3) = m$ and since NED_d is a metric then by the triangle inequality we require $\text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) \geq \frac{2k \cdot c + m}{1+k} \geq m = \text{NED}_d(w_1, w_3)$. Which entails that $2k \cdot c + m \geq m + mk$ and hence $c \geq \frac{m}{2}$.

2. In case of $\alpha_{1,2} = [\frac{\sigma_1}{\varepsilon}] [\frac{\varepsilon}{\sigma_3}] \cdot ([\frac{\varepsilon}{b}])^k$:

$$\text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) = \frac{d(\sigma_1, \varepsilon) + d(\varepsilon, \sigma_3) + k \cdot d(\varepsilon, b)}{2+k} + \frac{k \cdot d(b, \varepsilon)}{1+k} \geq \frac{2m + k \cdot c}{2+k} + \frac{k \cdot c}{1+k}$$

Again, since $\text{NED}_d(w_1, w_3) = m$ and since NED_d is a metric by the triangle inequality we require $\text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) \geq \frac{2m + k \cdot c}{2+k} + \frac{k \cdot c}{1+k} \geq m = \text{NED}_d(w_1, w_3)$. Which entails that

$$(2m + kc)(1+k) + kc(2+k) \geq m(2+k)(1+k)$$

$$2m + 2mk + kc + k^2c + 2kc + k^2c \geq 2m + 3mk + mk^2$$

$$c(2k^2 + 3k) \geq mk + mk^2$$

$$c \geq \frac{mk^2 + mk}{2k^2 + 3k}$$

14:8 When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

Taking k to infinity we get that

$$\lim_{k \rightarrow \infty} \frac{mk^2 + mk}{2k^2 + 3k} = \frac{m}{2}$$

Hence either way we get $c \geq \frac{m}{2}$. \triangleleft

► **Definition 14.** We say that an edit operation $\gamma \in \Gamma$ is essential if there exists $\alpha \in \Gamma^*$ such that α is an optimal edit path that uses γ . Otherwise γ is called inessential.

For example in the proof of Claim 7 we can see that $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ and $\left[\begin{smallmatrix} b \\ a \end{smallmatrix}\right]$ are inessential since transforming a to b via ε is always preferable, while $\left[\begin{smallmatrix} a \\ \varepsilon \end{smallmatrix}\right]$ is essential. We show that we can ignore inessential edit operations without changing the result.

▷ **Claim 15 (Essentials suffice).** Let Γ' be the restriction of Γ to only essential edit operations and let $d' : \Gamma' \rightarrow [0, 1]$ be the restriction of d to Γ' . Then $\text{NED}_d(w_1, w_2) = \text{NED}_{d'}(w_1, w_2)$ for every $w_1, w_2 \in \Sigma^*$.

Proof. Let $w_1, w_2 \in \Sigma^*$ we will show that $\text{NED}_d(w_1, w_2) = \text{NED}_{d'}(w_1, w_2)$. From Definition 3 we know that an edit path α for which $\text{NED}_d(w_1, w_2) = \text{cost}(\alpha)$ is an optimal edit path hence every edit operation in it is essential by Definition 14. Thus, all the edit operations in α exist in Γ' . It follows that $\text{NED}_{d'}(w_1, w_2) \leq \text{cost}(\alpha)$ and since Γ' does not have additional edit operations compared to Γ (and they agree on the costs of the mutual ones) the cost of $\text{NED}_{d'}(w_1, w_2)$ cannot be bigger than $\text{cost}(\alpha)$ or else α is not optimal in contradiction. Hence $\text{NED}_{d'}(w_1, w_2) = \text{NED}_d(w_1, w_2)$. \triangleleft

It follows from Claim 15 that we can assume without loss of generality that there are no inessential operations in d .

► **Remark 16.** Note that $\left[\begin{smallmatrix} a \\ \varepsilon \end{smallmatrix}\right]$ and $\left[\begin{smallmatrix} \varepsilon \\ a \end{smallmatrix}\right]$ are essential for every $a \in \Sigma$. This is since there is only one edit path from a to ε (and similarly from ε to a) and it involves these operations. Moreover, by Claim 9 if NED_d is a metric then $d(a, \varepsilon) = d(\varepsilon, a)$.

▷ **Claim 17 (A bound on the cost of an essential replace).** Let $a, b \in \Sigma$. If $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is an essential edit operation then there exists $i \in \mathbb{N}$ such that $d(a, b) < (d(a, \varepsilon) + d(\varepsilon, b))(1 - 1/i)$.

Proof. Since $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is essential there exists an optimal edit path α that uses it. Consider the shortest such optimal path. Note that $\text{cost}(\alpha) = \frac{\text{wgt}(\alpha) - d(a, b) + d(a, b)}{\text{len}(\alpha)}$. Consider a new edit path α' that does the same edit operations as α apart from $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ which it will replace by $\left[\begin{smallmatrix} a \\ \varepsilon \end{smallmatrix}\right] \left[\begin{smallmatrix} \varepsilon \\ b \end{smallmatrix}\right]$. Note that $\text{input}(\alpha) = \text{input}(\alpha')$ and $\text{output}(\alpha) = \text{output}(\alpha')$. Since α is optimal we know $\text{cost}(\alpha) \leq \text{cost}(\alpha')$. Moreover notice that $\text{cost}(\alpha') = \frac{\text{wgt}(\alpha) - d(a, b) + d(a, \varepsilon) + d(\varepsilon, b)}{\text{len}(\alpha) + 1}$.

Hence

$$\text{cost}(\alpha) = \frac{\text{wgt}(\alpha) - d(a, b) + d(a, b)}{\text{len}(\alpha)} \leq \frac{\text{wgt}(\alpha) - d(a, b) + d(a, \varepsilon) + d(\varepsilon, b)}{\text{len}(\alpha) + 1} = \text{cost}(\alpha')$$

Thus

$$\begin{aligned} (\text{len}(\alpha) + 1)(\text{wgt}(\alpha) - d(a, b)) + (\text{len}(\alpha) + 1) \cdot d(a, b) &\leq \\ \text{len}(\alpha)(\text{wgt}(\alpha) - d(a, b)) + \text{len}(\alpha)(d(a, \varepsilon) + d(\varepsilon, b)) & \end{aligned}$$

implying

$$(\text{wgt}(\alpha) - d(a, b)) + (\text{len}(\alpha) + 1) \cdot d(a, b) \leq \text{len}(\alpha)(d(a, \varepsilon) + d(\varepsilon, b))$$

Therefore

$$\begin{aligned} d(a, b) &\leq \frac{\text{len}(\alpha)(d(a, \varepsilon) + d(\varepsilon, b)) - (\text{wgt}(\alpha) - d(a, b))}{\text{len}(\alpha) + 1} \\ &\leq \frac{\text{len}(\alpha)(d(a, \varepsilon) + d(\varepsilon, b))}{\text{len}(\alpha) + 1} = \frac{(\text{len}(\alpha) + 1 - 1)(d(a, \varepsilon) + d(\varepsilon, b))}{\text{len}(\alpha) + 1} \\ &= d(a, \varepsilon) + d(\varepsilon, b) - \frac{d(a, \varepsilon) + d(\varepsilon, b)}{\text{len}(\alpha) + 1} = (d(a, \varepsilon) + d(\varepsilon, b)) \left(1 - \frac{1}{\text{len}(\alpha) + 1}\right) \end{aligned}$$

hence the claim holds for $i > \text{len}(\alpha) + 1$. \triangleleft

In Claim 7 we have shown that symmetry of d is not a necessary condition for NED_d to be a metric. In Claim 9 we showed that insert and delete operations are essential and must be symmetric. The following claim clarifies that if we restrict d to the essential operations then symmetry must hold.

\triangleright **Claim 18 (Symmetry of Essentials).** Let $a, b \in \Sigma$, if $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is an essential edit operation and NED_d is a metric then $\left[\begin{smallmatrix} b \\ a \end{smallmatrix}\right]$ is also essential and $d(a, b) = d(b, a)$.

Proof. Assume that $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is an essential edit operation and NED_d is a metric. Assume towards contradiction that $\left[\begin{smallmatrix} b \\ a \end{smallmatrix}\right]$ is not essential. From Claim 17 we know that there exists $i \in \mathbb{N}$ such that $d(a, b) < (d(a, \varepsilon) + d(\varepsilon, b))(1 - 1/i)$. Consider $w_1 = a^{i+1}$, $w_2 = a^i b$, let $\alpha = \left(\left[\begin{smallmatrix} a \\ a \end{smallmatrix}\right]\right)^i \left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ and $\alpha' = \left(\left[\begin{smallmatrix} a \\ a \end{smallmatrix}\right]\right)^i \left[\begin{smallmatrix} a \\ \varepsilon \end{smallmatrix}\right] \left[\begin{smallmatrix} \varepsilon \\ b \end{smallmatrix}\right]$. Notice that $\text{cost}(\alpha) = \frac{d(a, b)}{i+1}$ and $\text{cost}(\alpha') = \frac{d(a, \varepsilon) + d(\varepsilon, b)}{i+2}$. Since the only edit path that can cost less than α is α' we can check which of them is optimal. We argue that $\text{cost}(\alpha) < \text{cost}(\alpha')$. If this is the case then

$$\frac{d(a, b)}{i+1} < \frac{d(a, \varepsilon) + d(\varepsilon, b)}{i+2}$$

hence

$$d(a, b) < \frac{(i+1) \cdot (d(a, \varepsilon) + d(\varepsilon, b))}{i+2}$$

and so

$$d(a, b) < d(a, \varepsilon) + d(\varepsilon, b) - \frac{d(a, \varepsilon) + d(\varepsilon, b)}{i+2} = (d(a, \varepsilon) + d(\varepsilon, b)) \cdot \left(1 - \frac{1}{i+2}\right)$$

And this holds since $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is essential and so $d(a, b) < (d(a, \varepsilon) + d(\varepsilon, b))(1 - \frac{1}{i})$, by Claim 17. Hence α is optimal which means that $\text{NED}_d(w_1, w_2) = \text{cost}(\alpha)$ and since NED_d is a metric we know that $\text{cost}(\alpha) = \text{NED}_d(w_2, w_1)$ as well. Consider now the optional optimal edit paths from w_2 to w_1 . Let $\beta = \left(\left[\begin{smallmatrix} a \\ a \end{smallmatrix}\right]\right)^i \left[\begin{smallmatrix} b \\ a \end{smallmatrix}\right]$ and $\beta' = \left(\left[\begin{smallmatrix} a \\ a \end{smallmatrix}\right]\right)^i \left[\begin{smallmatrix} b \\ \varepsilon \end{smallmatrix}\right] \left[\begin{smallmatrix} \varepsilon \\ a \end{smallmatrix}\right]$. We know that NED_d is a metric hence following Claim 9 we know that $\text{cost}(\beta') = \text{cost}(\alpha')$ hence we know that $\text{cost}(\alpha) = \text{NED}_d(w_2, w_1) < \text{cost}(\beta')$. Thus $\text{cost}(\alpha) = \text{NED}_d(w_2, w_1) = \text{cost}(\beta)$, implying $\left[\begin{smallmatrix} b \\ a \end{smallmatrix}\right]$ is essential too and moreover $d(a, b) = d(b, a)$. \triangleleft

From Remark 16 we know that the only operations that can be inessential are $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ where $a, b \neq \varepsilon$. The following claim provides means to check if $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is essential or not.

\triangleright **Claim 19 (Essentialness Check).** For every $a, b \in \Sigma$ we have $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is inessential iff $d(a, b) \geq d(a, \varepsilon) + d(\varepsilon, b)$.

14:10 When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

Proof. \implies From Claim 17 we know that if $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is essential then there exists $i \in \mathbb{N}$ such that $d(a, b) < (d(a, \varepsilon) + d(\varepsilon, b))(1 - 1/i)$. Thus if such i does not exist (which means that $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is inessential), then for every i we have

$$d(a, b) \geq (d(a, \varepsilon) + d(\varepsilon, b))(1 - 1/i)$$

hence $d(a, b) \geq d(a, \varepsilon) + d(\varepsilon, b)$.

\Leftarrow Assume towards contradiction that $d(a, b) \geq d(a, \varepsilon) + d(\varepsilon, b)$ and $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ is essential. Let $n = d(a, b)$ and $m = d(a, \varepsilon) + d(\varepsilon, b)$. From the definition of essential, we know that there exists $w_1, w_2 \in \Sigma^*$ and $\alpha \in \Gamma^*$ such that α is an optimal path from w_1 to w_2 that uses $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$. Let k denote the number of occurrences of $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ in α . Let $\text{wgt}(\alpha) = p + n \cdot k$, and $\text{len}(\alpha) = \ell + k$. Now notice that if we replace every occurrence of $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$ with $\left[\begin{smallmatrix} a \\ \varepsilon \end{smallmatrix}\right] \left[\begin{smallmatrix} \varepsilon \\ b \end{smallmatrix}\right]$ we will get path α' from w_1 to w_2 where $\text{cost}(\alpha') = \frac{p+m \cdot k}{\ell+2 \cdot k} < \frac{p+n \cdot k}{\ell+k}$ in contradiction to α being an optimal path. \triangleleft

We are now ready to state the necessary condition on d for NED_d to be a metric.

► **Corollary 20 (Necessary Condition).** *Let $a, c \in \Sigma \cup \{\varepsilon\}$ and $b \in \Sigma$. Let $m = \sup\{\text{NED}_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\}$. A necessary condition for NED_d to be a metric is that d satisfies the following properties after removing inessential edit operations.*

1. $d(a, c) = 0$ iff $a = c$
2. $d(a, c) = d(c, a)$
3. $d(a, b) + d(b, c) \geq \min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\}$
4. $d(\varepsilon, b) = d(b, \varepsilon) \geq \frac{m}{2}$

Indeed, the first requirement is necessary by Claim 10, the second requirement by Claim 18, the third by Claim 11, and the fourth by Claim 13.³

► **Remark 21.** Let $m = \sup\{\text{NED}_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\}$. Note that we can assume without loss of generality that $m = 1$. If this is not the case then we define $d'(\sigma_1, \sigma_2) = \frac{1}{m}d(\sigma_1, \sigma_2)$. Then d' would satisfy that $\sup\{\text{NED}_{d'}(w_1, w_2) : w_1, w_2 \in \Sigma^*\} = 1$. In this case, it may be that there are σ_1, σ_2 for which $d'(\sigma_1, \sigma_2)$ are greater than 1 but such an edit operation is inessential.

► **Definition 22 (Fine Weight Function, Fine Metric).** *We call a function $d : \Gamma \rightarrow [0, 1]$ satisfying the conditions of Corollary 20 fine. Note that if d is a metric it satisfies the first three requirements. If it also satisfied the fourth requirement, we call it a fine metric.*

In the next section, we show that if $d : \Gamma \rightarrow [0, 1]$ is fine then NED_d is a metric. That is, d being fine is a sufficient and necessary condition for NED_d to be a metric.

5 Sufficient Condition

We turn to show that if d is fine then NED_d is a metric. That is, that the necessary condition provided in the previous section is also a sufficient.

► **Claim 23.** If d is fine then NED_d satisfies the identity of indiscernibles requirement.

³ For GED, a sufficient condition was given in [12]. We conjecture that it is not a necessary condition, and GED may be a metric also when deletion of different letters costs differently as in d of Claim 7.

Proof. Assume d is fine, and let $w_1, w_2 \in \Sigma^*$.

1. Case $w_1 = w_2$. We show that $\text{NED}_d(w_1, w_2) = 0$. Since d is fine we know that $d(a, a) = 0$ for every $a \in \Sigma$. Thus, we can construct an edit path α that applies *no-op* to each letter which leads to that $\text{wgt}(\alpha) = 0$. Hence $\text{NED}_d(w_1, w_2) = 0$.
2. Case $w_1 \neq w_2$. Let $\alpha \in \Gamma^*$ be an optimal edit path that transforms w_1 to w_2 . Notice that α needs at least one edit operation, denote it γ , that is not *no-op*. Since d is fine we know that $\text{wgt}(\gamma) > 0$. Hence $\text{NED}_d(w_1, w_2) = \text{cost}(\alpha) = \frac{\text{wgt}(\alpha)}{\text{len}(\alpha)} \geq \frac{\text{wgt}(\gamma)}{\text{len}(\alpha)} > 0$. \triangleleft

▷ **Claim 24.** If d is fine then NED_d satisfies the symmetry requirement.

Proof. Assume that d is fine and assume towards contradiction that there exists $w_1, w_2 \in \Sigma^*$ such that $\text{NED}_d(w_1, w_2) \neq \text{NED}_d(w_2, w_1)$. Assume w.l.o.g. that $\text{NED}_d(w_1, w_2) < \text{NED}_d(w_2, w_1)$. Let $\alpha_{1,2}$ be an optimal path that transforms w_1 to w_2 . Let $\gamma \in \alpha_{1,2}$ such that $\gamma = \left[\begin{smallmatrix} a \\ b \end{smallmatrix} \right]$ where $a, b \in \Sigma \cup \{\varepsilon\}$ and either $a \neq \varepsilon$ or $b \neq \varepsilon$. From Definition 14 and Claim 18 we know that γ is essential and so is $\left[\begin{smallmatrix} b \\ a \end{smallmatrix} \right]$. Since d is fine we know that $d(a, b) = d(b, a)$. We refer to $\left[\begin{smallmatrix} b \\ a \end{smallmatrix} \right]$ as the opposite edit operation of $\left[\begin{smallmatrix} a \\ b \end{smallmatrix} \right]$. Note that if we replace every edit operation in $\alpha_{1,2}$ with its opposite edit operation, we will receive a new edit path $\alpha_{2,1}$ that transforms w_2 to w_1 and $\text{cost}(\alpha_{2,1}) = \text{cost}(\alpha_{1,2}) < \text{NED}_d(w_2, w_1)$, contradicting the definition of NED . \triangleleft

We proceed to show that the triangle inequality also holds.

The idea of the proof of [7] that NED satisfies the triangle inequality for the uniform case is to take two edit paths $\alpha_{1,2}$ and $\alpha_{2,3}$ from words w_1 to w_2 and from w_2 to w_3 and extract from them an edit path $\alpha_{1,3}$ from w_1 to w_3 that costs at most their sum. We follow that idea but generalize and simplify the proof.

The heart of the simplification lies in finding a way to align the two edit paths so that their composition to a new edit path from w_1 to w_3 is seamless, and we can easily prove that it costs less than the sum.

We proceed by showing how to compose the two paths. The composition uses as an intermediate step a pair of extended edit paths $\alpha'_{1,2}, \alpha'_{2,3}$ that align the give edit paths $\alpha_{1,2}$ and $\alpha_{2,3}$, with respect to one another.⁴

► **Definition 25** (Alignment of edit paths). *Let $\alpha_{1,2}$ and $\alpha_{2,3}$ be such that $\text{output}(\alpha_{1,2}) = \text{input}(\alpha_{2,3})$. We say that $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ is the alignment of $\alpha_{1,2}$ and $\alpha_{2,3}$ if $\alpha'_{1,2}$ and $\alpha'_{2,3}$ are the shortest extended edit paths satisfying that*

- $\alpha'_{1,2}$ is obtained from $\alpha_{1,2}$ by inserting some $\left[\begin{smallmatrix} \varepsilon \\ \varepsilon \end{smallmatrix} \right]$ letters,
- $\alpha'_{2,3}$ is obtained from $\alpha_{2,3}$ by inserting some $\left[\begin{smallmatrix} \varepsilon \\ \varepsilon \end{smallmatrix} \right]$ letters,
- and $\pi_2(\alpha'_{1,2}) = \pi_1(\alpha'_{2,3})$.

The first requirement guarantees that the input and output of $\alpha'_{1,2}$ is the same as those of $\alpha_{1,2}$, and the second requirement gives the analogous guarantees regarding $\alpha'_{2,3}$ and $\alpha_{2,3}$. The third requirement strengthens the connection between $\text{output}(\alpha_{1,2})$ and $\text{input}(\alpha_{2,3})$ and demands that they agree not only on the letters of the interim word w_2 , but also on the occurrences of ε . This in particular requires $\alpha'_{1,2}$ and $\alpha'_{2,3}$ to be of the same length. Using the $\left[\begin{smallmatrix} \sigma \\ \sigma' \end{smallmatrix} \right]$ notations, if we write $\alpha'_{1,2}$ and $\alpha'_{2,3}$ one above the other then the second and third lines are the same.

⁴ Recall that an extended edit path is a string over $\hat{\Gamma}$, namely it may use $\left[\begin{smallmatrix} \varepsilon \\ \varepsilon \end{smallmatrix} \right]$ on top of the usual edit operations.

14:12 When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

► **Example 26.** Let $w_1 = a_1a_2a_3$, $w_2 = b_1b_2$ and $w_3 = c_1c_2c_3c_4$. Then $\alpha_{1,2} = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \begin{bmatrix} a_2 \\ \varepsilon \end{bmatrix} \begin{bmatrix} a_3 \\ b_2 \end{bmatrix}$ is an edit path between w_1 and w_2 and $\alpha_{2,3} = \begin{bmatrix} \varepsilon \\ c_1 \end{bmatrix} \begin{bmatrix} b_1 \\ c_2 \end{bmatrix} \begin{bmatrix} b_2 \\ c_3 \end{bmatrix} \begin{bmatrix} \varepsilon \\ c_4 \end{bmatrix}$ is an edit path between w_2 and w_3 . Using the \mapsto notation, we can write these as $a_1a_2a_3 \mapsto b_1_b_2$ and $_b_1b_2_ \mapsto c_1c_2c_3c_4$.

Let $\alpha'_{1,2} = \begin{bmatrix} \varepsilon \\ b_1 \end{bmatrix} \begin{bmatrix} a_1 \\ \varepsilon \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \begin{bmatrix} a_3 \\ \varepsilon \end{bmatrix}$ and $\alpha'_{2,3} = \begin{bmatrix} \varepsilon \\ c_1 \end{bmatrix} \begin{bmatrix} b_1 \\ c_2 \end{bmatrix} \begin{bmatrix} \varepsilon \\ c_3 \end{bmatrix} \begin{bmatrix} b_2 \\ c_4 \end{bmatrix}$. Then $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ is their *alignment*. Using the \mapsto notations these are $_a_1a_2a_3_ \mapsto _b_1_b_2_$ and $_b_1_b_2_ \mapsto c_1c_2_c_3c_4$, on which it is perhaps easier to see that the output of $\alpha'_{1,2}$ and the input of $\alpha'_{2,3}$ agree also on ε positions.

Note that if $\pi_2(\alpha_{1,2})$ contains i occurrences of ε and $\pi_1(\alpha_{2,3})$ contains j occurrences of ε then there exist $\alpha'_{1,2}$ and $\alpha'_{2,3}$ of length at most $w_2 + i + j$ such that $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ is their alignment. Moreover, the alignment can be constructed iteratively by following $\pi_2(\alpha_{1,2})$ and $\pi_1(\alpha_{2,3})$ and if the current index (of the considered projections) is not the same, inserting $(\varepsilon, \varepsilon)$ to either $\alpha_{1,2}$ or $\alpha_{2,3}$ depending on which has advanced less (in terms of letters of w_2).

We are now ready to define the composition of $\alpha_{1,2}$ and $\alpha_{2,3}$.

► **Definition 27 (Compose).** Let $\alpha_{1,2}$ and $\alpha_{2,3}$ be such that $\text{output}(\alpha_{1,2}) = \text{input}(\alpha_{2,3})$ and let $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ be their alignment. Assume $\alpha'_{1,2} = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \dots \begin{bmatrix} a_k \\ b_k \end{bmatrix}$ and $\alpha'_{2,3} = \begin{bmatrix} b_1 \\ c_1 \end{bmatrix} \begin{bmatrix} b_2 \\ c_2 \end{bmatrix} \dots \begin{bmatrix} b_k \\ c_k \end{bmatrix}$. Let $\alpha''_{1,3} = \begin{bmatrix} a_1 \\ c_1 \end{bmatrix} \begin{bmatrix} a_2 \\ c_2 \end{bmatrix} \dots \begin{bmatrix} a_k \\ c_k \end{bmatrix}$. Let $\alpha'_{1,3}$ be the edit path obtained from $\alpha''_{1,3}$ by replacing $\begin{bmatrix} a \\ c \end{bmatrix}$ with $\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ c \end{bmatrix}$ for every a, c for which $d(a, c) \geq d(a, \varepsilon) + d(\varepsilon, c)$. Finally, let $\alpha_{1,3}$ be the edit path obtained from $\alpha'_{1,3}$ by removing the $\begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$ letters.

▷ **Claim 28.** Let $\alpha_{1,2}$ and $\alpha_{2,3}$ be such that $\text{output}(\alpha_{1,2}) = \text{input}(\alpha_{2,3})$. If $\alpha_{1,3}$ is the result of composing $\alpha_{1,2}$ and $\alpha_{2,3}$ as per Definition 27 then $\alpha_{1,3}$ is an edit path between $\text{input}(\alpha_{1,2})$ and $\text{output}(\alpha_{2,3})$.

Proof. Let $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ be the alignment of $\alpha_{1,2}$ and $\alpha_{2,3}$. Then α and α' agree on their input and output for $\alpha \in \{\alpha_{1,2}, \alpha_{2,3}\}$ since they only differ in $\begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$ letters. Let $\alpha''_{1,3}$ and $\alpha'_{1,3}$ be as described in Definition 27. It is easy to see that $\text{input}(\alpha''_{1,3}) = \text{input}(\alpha'_{1,2})$ and $\text{output}(\alpha''_{1,3}) = \text{output}(\alpha'_{2,3})$ since $\text{input}(\begin{bmatrix} a \\ c \end{bmatrix}) = a = \text{input}(\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ c \end{bmatrix})$ and similarly $\text{output}(\begin{bmatrix} a \\ c \end{bmatrix}) = c = \text{output}(\begin{bmatrix} a \\ \varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon \\ c \end{bmatrix})$. The claim follows by transitivity of equality. ◁

The following lemma is the heart of the proof that the triangle inequality holds for NED_d given d is fine.

► **Lemma 29.** Assume $d : \Gamma \rightarrow [0, 1]$ is fine. Let $\alpha_{1,2}$ and $\alpha_{2,3}$ be such that $\text{output}(\alpha_{1,2}) = \text{input}(\alpha_{2,3})$. Let $\alpha''_{1,3}$, $\alpha'_{1,3}$ and $\alpha_{1,3}$ be as described in Definition 27. Let n be the number of occurrences of $\begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$ in $\alpha'_{1,3}$. Then

1. $\text{len}(\alpha_{1,3}) \geq \max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\} - n$
2. $\text{wgt}(\alpha_{1,3}) \leq \text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - n$
3. $\text{cost}(\alpha_{1,3}) \leq \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) - n$

The proof relies on the following fact.

► **Fact 30.** Let $d, e, n \in \mathbb{N}$ such that $e \leq d$. Then $\frac{e-n}{d-n} \leq \frac{e}{d}$

We can now prove Lemma 29.

Proof of Lemma 29. Let $\langle \alpha'_{1,2}, \alpha'_{2,3} \rangle$ be the alignment of $\alpha_{1,2}$ and $\alpha_{2,3}$.

1. By the construction of the aligned edit paths there is no index i such that both $\alpha'_{1,2}[i] = \begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$ and $\alpha'_{2,3}[i] = \begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}$. Thus for every index i of $\alpha''_{1,3}$ either $\alpha'_{1,2}[i]$ is an element of $\alpha_{1,2}$ or $\alpha'_{2,3}[i]$ is an element of $\alpha_{2,3}$ (or both are). It follows that

$$\text{len}(\alpha''_{1,3}) \geq \max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\}$$

Since $\alpha_{1,3}$ is obtained from $\alpha'_{1,3}$ by removing the occurrences of $[\frac{\varepsilon}{\varepsilon}]$ and there are n such we get that

$$\text{len}(\alpha_{1,3}) = \text{len}(\alpha'_{1,3}) - n$$

Because $\text{len}(\alpha'_{1,3}) \geq \text{len}(\alpha''_{1,3})$ we get overall that

$$\text{len}(\alpha_{1,3}) = \text{len}(\alpha'_{1,3}) - n \geq \text{len}(\alpha''_{1,3}) - n \geq \max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\} - n$$

as required.

2. First note that $\text{wgt}(\alpha) = \text{wgt}(\alpha')$ for all $\alpha \in \{\alpha_{1,2}, \alpha_{2,3}, \alpha_{1,3}\}$ since α and α' differ only by elements of the form $[\frac{\varepsilon}{\varepsilon}]$ and since d is fine, by the first requirement, $d(\varepsilon, \varepsilon) = 0$. Second, we claim that $\text{wgt}(\alpha'_{1,3}) \leq \text{wgt}(\alpha'_{1,2}) + \text{wgt}(\alpha'_{2,3})$. This holds since for every element $[\frac{a}{c}]$ of $\alpha'_{1,3}$ there exists elements $[\frac{a}{b}]$ and $[\frac{b}{c}]$ in $\alpha'_{1,2}$ and $\alpha'_{2,3}$ respectively, where $a, b, c \in \Sigma \cup \{\varepsilon\}$. Hence for every respective element $[\frac{a}{c}]$ or respective two elements of $[\frac{a}{\varepsilon}][\frac{\varepsilon}{c}]$ of $\alpha'_{1,3}$ there exists elements $[\frac{a}{b}]$ and $[\frac{b}{c}]$ in $\alpha'_{1,2}$ and $\alpha'_{2,3}$ respectively. To see how the corresponding weights relate we split into cases.
- If both $a \neq \varepsilon$ and $c \neq \varepsilon$ then by the third requirement of being fine $\min\{d(a, c), d(a, \varepsilon) + d(\varepsilon, c)\} \leq d(a, b) + d(b, c)$ and according to the minimum $[\frac{a}{c}]$ or $[\frac{a}{\varepsilon}][\frac{\varepsilon}{c}]$ occurs in $\alpha'_{1,3}$.
 - If $a \neq \varepsilon$ and $c = \varepsilon$ then by the third requirement of being fine and Remark 12 we have $d(a, b) + d(b, \varepsilon) \geq d(a, \varepsilon)$.
 - If $a = \varepsilon$ and $c \neq \varepsilon$ then by the third requirement of being fine and Remark 12 we have $d(\varepsilon, b) + d(b, c) \geq d(\varepsilon, c)$.

Thus we get

$$\text{wgt}(\alpha'_{1,3}) \leq \text{wgt}(\alpha'_{1,2}) + \text{wgt}(\alpha'_{2,3}) = \text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3})$$

Last, we note that each occurrence of $[\frac{\varepsilon}{\varepsilon}]$ in $\alpha'_{1,3}$ corresponds to an occurrence of $[\frac{\varepsilon}{b}]$ in $\alpha_{1,2}$ and $[\frac{b}{\varepsilon}]$ in $\alpha_{2,3}$ for some $b \in \Sigma$. Let b_1, b_2, \dots, b_n be the respective letters in $\alpha_{1,2}$ or $\alpha_{2,3}$. The weight of $[\frac{\varepsilon}{\varepsilon}]$ in $\alpha'_{1,3}$ is 0 whereas the original components had some non-zero weight. Hence

$$\text{wgt}(\alpha'_{1,3}) \leq \text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - \sum_{i=1}^n (\text{wgt}([\frac{\varepsilon}{b_i}]) + \text{wgt}([\frac{b_i}{\varepsilon}]))$$

From the fourth requirement of being fine we know $d(\varepsilon, b) = d(b, \varepsilon) \geq \frac{1}{2}$. Thus $\sum_{i=1}^n (\text{wgt}([\frac{\varepsilon}{b_i}]) + \text{wgt}([\frac{b_i}{\varepsilon}])) \geq n$ and hence

$$\text{wgt}(\alpha_{1,3}) = \text{wgt}(\alpha'_{1,3}) \leq \text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - n$$

as required.

3. From items (2) and (1) we get

$$\frac{\text{wgt}(\alpha_{1,3})}{\text{len}(\alpha_{1,3})} \leq \frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - n}{\text{len}(\alpha_{1,3})} \leq \frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - n}{\max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\} - n}$$

Applying Fact 30 we get

$$\frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3}) - n}{\max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\} - n} \leq \frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3})}{\max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\}}$$

14:14 When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?

Assume without loss of generality that $\text{len}(\alpha_{1,2}) \geq \text{len}(\alpha_{2,3})$ then

$$\begin{aligned} \frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3})}{\max\{\text{len}(\alpha_{1,2}), \text{len}(\alpha_{2,3})\}} &= \frac{\text{wgt}(\alpha_{1,2}) + \text{wgt}(\alpha_{2,3})}{\text{len}(\alpha_{1,2})} \\ &= \frac{\text{wgt}(\alpha_{1,2})}{\text{len}(\alpha_{1,2})} + \frac{\text{wgt}(\alpha_{2,3})}{\text{len}(\alpha_{1,2})} \leq \frac{\text{wgt}(\alpha_{1,2})}{\text{len}(\alpha_{1,2})} + \frac{\text{wgt}(\alpha_{2,3})}{\text{len}(\alpha_{2,3})} \end{aligned}$$

Overall we get

$$\text{cost}(\alpha_{1,3}) = \frac{\text{wgt}(\alpha_{1,3})}{\text{len}(\alpha_{1,3})} \leq \frac{\text{wgt}(\alpha_{1,2})}{\text{len}(\alpha_{1,2})} + \frac{\text{wgt}(\alpha_{2,3})}{\text{len}(\alpha_{2,3})} = \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3})$$

as required. \blacktriangleleft

With this proof in place we can conclude that d being fine is a sufficient condition for NED_d to be a metric.

► **Theorem 31.** *Let $d : \hat{\Gamma} \rightarrow [0, 1]$ be fine. Then NED_d is a metric.*

Proof. Given $d : \hat{\Gamma} \rightarrow [0, 1]$ is fine it is easy to see that NED_d satisfies the first two requirements of a metric. To see that it also satisfies the triangle inequality, let $w_1, w_2, w_3 \in \Sigma^*$. Let $\alpha_{1,2}$ be an optimal edit path between w_1, w_2 and $\alpha_{2,3}$ an optimal edit path between w_2, w_3 . Let $\alpha_{1,3}$ be the result of composing $\alpha_{1,2}, \alpha_{2,3}$ via Definition 27. By Claim 28, $\alpha_{1,3}$ is an edit path between w_1 and w_3 and by Lemma 29, $\text{cost}(\alpha_{1,3}) \leq \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3})$. Thus

$$\text{NED}_d(w_1, w_3) \leq \text{cost}(\alpha_{1,3}) \leq \text{cost}(\alpha_{1,2}) + \text{cost}(\alpha_{2,3}) = \text{NED}_d(w_1, w_2) + \text{NED}_d(w_2, w_3)$$

as required. \blacktriangleleft

► **Corollary 32.** *NED_d is a metric if and only if d is fine.*

This corollary proves Theorem 1.

► **Remark 33.** Consider d of Claim 6. The first two requirements of being fine are obviously met. By Claim 19 the operations $\begin{bmatrix} a \\ b \end{bmatrix}$ and $\begin{bmatrix} b \\ a \end{bmatrix}$ are inessential. Hence the third requirement clearly hold. Note that $m = \sup\{\text{NED}_d(w_1, w_2) : w_1, w_2 \in \Sigma^*\} = 0.5$ and $\frac{m}{2} = 0.25$ hence the fourth requirement holds.

Consider d of Claim 7. The first requirement of being fine clearly holds. While it breaks symmetry, if we remove the inessential operations, namely $\begin{bmatrix} a \\ b \end{bmatrix}$ and $\begin{bmatrix} b \\ a \end{bmatrix}$ then symmetry is maintained. Since $\begin{bmatrix} a \\ b \end{bmatrix}$ and $\begin{bmatrix} b \\ a \end{bmatrix}$ are inessentials the third requirement holds as well. Finally the fourth requirement holds since $m = 0.45$ (as $\begin{bmatrix} a \\ b \end{bmatrix}$ and $\begin{bmatrix} b \\ a \end{bmatrix}$ are inessentials) and $\frac{m}{2} = 0.225$.

6 Discussion

Now that we have a sufficient and necessary condition for $d : \Gamma \rightarrow [0, 1]$ for NED_d to be a metric, it is easy to verify or come up with such d 's for certain applications. We give some examples in §6.1. In §6.2 we discuss extensions to infinite words and applications in formal verification.

6.1 Examples for fine weight functions

Recall that given an alphabet Σ , we use Γ for $\hat{\Gamma} \setminus \{[\varepsilon]\}$ where $\hat{\Gamma} = (\Sigma \cup \{\varepsilon\})^2$. Given a function $d : \Sigma \times \Sigma \rightarrow [0, 1]$ and given $c \in [\frac{1}{2}, 1]$ we augment it to a function $d^c : \Gamma \rightarrow [0, 1]$ as follows:

$$d^c(\sigma_1, \sigma_2) = \begin{cases} d(\sigma_1, \sigma_2) & \text{if } \sigma_1 \neq \varepsilon \text{ and } \sigma_2 \neq \varepsilon \\ c & \text{if } \sigma_1 = \varepsilon \text{ or } \sigma_2 = \varepsilon \end{cases}$$

Consider the case where $\Sigma = [0, n]$ for some $n \in \mathbb{N}$, that is Σ is a finite interval of the natural numbers, starting with 0. Then the following distance over Σ is fine.

► **Example 34** (Distances in $[0, n]$). Let $d_n : [0, n] \times [0, n] \rightarrow [0, 1]$ be defined as follows:

$$d_n(n_1, n_2) = \frac{|n_1 - n_2|}{n + 1}$$

▷ **Claim 35.** The weight function d_n^c is fine.

Consider now the case that $\Sigma = \mathbb{N}$, i.e., Σ is the set of natural number. We can show that the following distance [16] is fine.

► **Example 36** (Distances in \mathbb{N}). Let $d_{\mathbb{N}} : \mathbb{N} \times \mathbb{N} \rightarrow [0, 1]$ be defined as follows:

$$d_{\mathbb{N}}(n_1, n_2) = 1 - \frac{1}{|n_1 - n_2| + 1}$$

▷ **Claim 37.** The weight function $d_{\mathbb{N}}^c$ is fine.

► **Example 38** (Distances between sets). Let $\Sigma = 2^A$ for some finite set of elements A . Let $d_{\text{set}} : 2^A \times 2^A \rightarrow [0, 1]$ be defined as follows, where \oplus denotes the symmetrical difference:

$$d_{\text{set}}(S_1, S_2) = \frac{|S_1 \oplus S_2|}{|A|}$$

▷ **Claim 39.** The weight function d_{set}^c is fine.

In *model checking* [1, 2, 3], automata are defined with respect to a set $AP = \{p_1, p_2, \dots, p_k\}$ of atomic propositions and the alphabet is $\Sigma = 2^{AP}$. We can use d_{set} to measure the distance between letters, but in a setting where a noise may alter the value of one of the atomic propositions it makes sense to define the distance between two letters as the Hamming distance between the two letters, divided by k for normalization.⁵

► **Example 40** (Distances in $\Sigma = 2^k$). Let $\Sigma = 2^k$. Let $d_{\text{prop}} : 2^k \times 2^k \rightarrow [0, 1]$ be defined as follows:

$$d_{\text{prop}}(v_1, v_2) = \frac{\text{HD}(v_1, v_2)}{k}$$

▷ **Claim 41.** The weight function d_{prop}^c is fine.

The transitions in automata used in model checking, are usually expressed using Boolean expressions over the set of atomic propositions, e.g. the Boolean expression $p_1 \wedge (\neg p_5 \vee p_7)$ abbreviates the set of letters $\sigma \in 2^k$ where the first bit is 1 and either the fifth bit is zero or the seventh bit is 1, and the rest of the bits can be anything. In general, a Boolean expression b is a compact way to represent the set of letters $\{\sigma \in 2^k \mid \sigma \models b\}$. This type of automata is a special case of *symbolic finite automata* (SFA) that are defined with respect to a concrete alphabet Σ and a symbolic alphabet Ψ of predicates over Σ (see [4] for an introduction to SFAs). The predicates are associated with a semantic function $\llbracket \cdot \rrbracket$ that maps a predicate ψ to a subset of Σ that consists of the concrete letters satisfying it. The distance d_{pred} between predicates ψ_1 and ψ_2 can thus be defined using d_{set} on $\llbracket \psi_1 \rrbracket$ and $\llbracket \psi_2 \rrbracket$.

► **Corollary 42.** We have that $NED_{d_n^c}$, $NED_{d_{\mathbb{N}}^c}$, $NED_{d_{\text{set}}^c}$, $NED_{d_{\text{prop}}^c}$ and $NED_{d_{\text{pred}}^c}$ are metrics.

⁵ The Hamming distance, $\text{HD} : \bigcup_{k \in \mathbb{N}} (\Sigma^k \times \Sigma^k) \rightarrow \mathbb{N}$, is defined between two strings of the same length, as the number of positions in which they differ [9].

6.2 Applications in Formal Verification

The *robustness* question in verification, roughly speaking, asks how much a system S can be altered so that it still satisfies its specification T . Suppose the distance between words is given by dist , and that $\llbracket S \rrbracket$ is the set of computations induced by the system and $\llbracket T \rrbracket$ is the set of allowed computations according to the specification T . It is noted in [6] that the robustness question can be reduced to question of computing the distance between the languages $\llbracket S \rrbracket$ and $\llbracket T \rrbracket$ defined as: $\inf_{w_1 \in \llbracket S \rrbracket} \inf_{w_2 \in \llbracket T \rrbracket} \text{dist}(w_1, w_2)$. It is shown in [8, Theorem 18] that when S and T are given by non-deterministic finite automata and dist is NED over the uniform weights this can be computed in polynomial time. The proof is by building a so called *edit distance graph* of two NFAs, and using the fact that the infimum of the mean weights of paths from a set of origin nodes to a set of target nodes can be computed in polynomial time [6]. Since the same graph can be constructed for NED_d , with the only difference that the weights of edges follow the given d rather than follow the uniform weights, and since the proof in [6] works on any weighted graph in which the weights are rationals, we can conclude that NED_d between languages can be computed in polynomial time, if d gives rational weights. Note that this is the case in all examples considered in §6.1.

In formal verification, systems and specifications are usually defined over infinite words. It is thus desired to have a function $\text{dist} : \Sigma^\omega \times \Sigma^\omega \rightarrow [0, 1]$ that measure the distance between two infinite words. In [8, Thm. 6] it was shown that $\bar{w}\text{-NED}(w_1, w_2)$ which is defined as $\limsup_{i \rightarrow \infty} \text{NED}(w_1[..i], w_2[..i])$ is a metric on infinite words. We can similarly define $\bar{w}\text{-NED}_d(w_1, w_2)$ as $\limsup_{i \rightarrow \infty} \text{NED}_d(w_1[..i], w_2[..i])$ and the same proof goes through. To compute the distance between two ultimately periodic words,⁶ it is shown [8, Thm. 8] that it suffices to consider the best rotations of the periodic parts. Thus reducing computation of $\bar{w}\text{-NED}$ to computation of NED, which can be done in polynomial time [13]. This proof works also for $\bar{w}\text{-NED}_d$ if d is non-uniform and gives rational weights. To compute the distance between S and T given by non-deterministic Büchi automata (NBA),⁷ [8] requires a more sophisticated version of the edit graph, which tracks along a cycle the number of insert and deletes to coordinate that they are balanced, namely that the same number of letters is read in both automata. The same technique would work in the case of non-uniform weights. We conclude that the robustness question when S and T are NBAs and the considered distance is $\bar{w}\text{-NED}_d$ for some fine non-uniform weight function d that gives rational weights can also be computed in polynomial time.

References

- 1 C. Baier and J-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.
- 2 Edmund M. Clarke, Orna Grumberg, Daniel Kroening, Doron A. Peled, and Helmut Veith. *Model checking, 2nd Edition*. MIT Press, 2018. URL: <https://mitpress.mit.edu/books/model-checking-second-edition>.
- 3 Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem, editors. *Handbook of Model Checking*. Springer, 2018. doi:10.1007/978-3-319-10575-8.
- 4 Loris D’Antoni and Margus Veanes. The power of symbolic automata and transducers. In *Computer Aided Verification - 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part I*, pages 47–67, 2017.

⁶ Restricting infinite words to ultimately periodic words is common in formal verification since two regular ω -languages are equivalent iff they agree on the set of ultimately periodic words [15].

⁷ NBA is the most common computational model used in formal verification.

- 5 Colin de la Higuera and Luisa Micó. A contextual normalised edit distance. In *Proceedings of the 24th International Conference on Data Engineering Workshops, ICDE 2008, April 7-12, 2008, Cancún, Mexico*, pages 354–361. IEEE Computer Society, 2008.
- 6 Emmanuel Filiot, Nicolas Mazzocchi, Jean-François Raskin, Sriram Sankaranarayanan, and Ashutosh Trivedi. Weighted transducers for robustness verification. In *31st International Conference on Concurrency Theory, CONCUR 2020, September 1-4, 2020, Vienna, Austria (Virtual Conference)*, pages 17:1–17:21, 2020.
- 7 Dana Fisman, Joshua Grogin, Oded Margalit, and Gera Weiss. The normalized edit distance with uniform operation costs is a metric. In Hideo Bannai and Jan Holub, editors, *33rd Annual Symposium on Combinatorial Pattern Matching, CPM 2022, June 27-29, 2022, Prague, Czech Republic*, volume 223 of *LIPICs*, pages 17:1–17:17, 2022.
- 8 Dana Fisman, Joshua Grogin, and Gera Weiss. A normalized edit distance on infinite words. In *31st EACSL Annual Conference on Computer Science Logic, CSL 2023, February 13-16, 2023, Warsaw, Poland*, pages 20:1–20:20, 2023.
- 9 R. W. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160, April 1950. doi:10.1002/j.1538-7305.1950.tb00463.x.
- 10 Karen Kukich. Techniques for automatically correcting words in text. *ACM Comput. Surv.*, 24(4):377–439, December 1992. doi:10.1145/146370.146380.
- 11 Vladimir Iosifovich Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710, February 1966. Doklady Akademii Nauk SSSR, V163 No4 845-848 1965.
- 12 Yujian Li and Bi Liu. A normalized levenshtein distance metric. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(6):1091–1095, 2007.
- 13 Andrés Marzal and Enrique Vidal. Computation of normalized edit distance and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(9):926–932, 1993.
- 14 Gonzalo Navarro. A guided tour to approximate string matching. *ACM Comput. Surv.*, 33(1):31–88, March 2001. doi:10.1145/375360.375365.
- 15 Büchi J. R. On a decision method in restricted second order arithmetic. In *Int. Congress on Logic, Method, and Philosophy of Science*, pages 1–12. Stanford University Press, 1962.
- 16 Sanda Zilles. A distance on \mathbb{N} . Private communication, 2023.
- 17 David Sankoff and Joseph B. Kruskal. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983.
- 18 Robert A. Wagner and Michael J. Fischer. The string-to-string correction problem. *J. ACM*, 21(1):168–173, January 1974. doi:10.1145/321796.321811.
- 19 Achim Weigel and Frank Fein. Normalizing the weighted edit distance. In *12th IAPR International Conference on Pattern Recognition, Conference B: Patern Recognition and Neural Networks, ICPR 1994, Jerusalem, Israel, 9-13 October, 1994, Volume 2*, pages 399–402, 1994.