Simple Wasserstein Two-Sample Statistical Test in a Tree Metric Space Applied to Sentences in Comparison with Random Trees

Kumiko Tanaka-Ishii*and
Kei Kobayashi†

Abstract

This article proposes a lightweight, simple, two-sample statistical hypothesis test method that is applicable in a tree metric space. Tree metric spaces provide a new mathematical possibility for analysis of real-world trees, but a method that scales is required, because trees are computationally costly and real-world data is typically large. We propose to measure the distance between two sets of trees via the Wasserstein distance, and we leverage this approach to propose our test method as a double resampling algorithm of bootstrap and permutation tests. The test is shown theoretically to have a good power to distinguish two sets of trees. Furthermore, the test method is shown empirically to be stable across different parameters that are necessary for data comparison. For an application, we compare sentences with various random trees, and we show that only random sentences sampled from recent large language models produced a relative positive p-value.

1 Introduction

This article presents a statistical test method to compare two sets of trees in a tree metric space. There have been emerging fundamental studies on the geometry of tree metric spaces (Billera *et al.*, 2001; Feragen *et al.*, 2013). These spaces will provide new means to analyze real-world trees, but their application to real-world data is still limited, to the best of our knowledge. One reason for this is that statistical tools for such analysis of real-world data have not been established.

A set of real-world trees can be embedded in a metric space via their distances and then compared with sets of other trees. The real-world trees are biased in shape and thus locally allocated in the space, and we require methods to statistically capture this bias in the space. There are recent topological methods that consider the nature of a point set in a space (Krishna and Yusu, 2022), but real-world data often does not concern topological continuity. We believe there could be a more standard statistical method to consider the nature of trees in a space, via comparison among surrogate data sets.

Recently there have been reports about two-sample tests that are potentially applicable for analysis of tree metric spaces. One report was based on the Multi-Response Permutation Procedure (Liu *et al.*, 2022), and another uses the Kernel Projected Wasserstein Distance (Wang *et al.*, 2022). The tests' computational costs are cubic in the sample size, however, and they would not scale; in addition, the application to trees is costly and non-trivial. Instead, we seek a more lightweight, simpler test method that works seamlessly with the distance function between trees. Furthermore, data analysis involves parameters, and we seek a test method that produces stable results across the parameters.

Our proposed test method serves for comparison among sets of tree data in general, and we compare sentences with respect to various random tree structures, including the structures of recent large-scale

*Waseda University, Nishi-Shinjuku, Japan kumiko@waseda.jp
†Keio University, Yokohama, Japan

language models. After formalizing the tree metric space, we use the Wasserstein distance within the space to quantify the distance between two sets of trees (Villani, 2008; Peyré and Cuturi, 2019). We then propose to statistically test this distance, indicating whether the two sets of trees are the same, by naturally incorporating previous test methods (Good, 2013; Pesarin, 2010) in a double resampling algorithm of bootstrap and permutation tests.

We apply our proposed framework to analyze sentences in comparison with various random trees, which are derived from various representative language models that have appeared over time: Markov models, context-free models, and texts generated by recent large language models. We believe that trees hold an important key to understanding the new technology for data science, because they work not only through scale improvement but also through training via prompting in units of sentences.

We report that, among the random models, only trees generated by ChatGPT are judged as similar to natural language trees. This result provides important evidence implying that, for the first time in history, a language model can perform unsupervised acquisition of sentence structure. Furthermore, we examine the differences among sentences across multiple languages, and we show that the differences in tree structure even without words can capture the grammatical characteristics of a language.

2 Related Work

There are three kinds of related work with respect to this article: 1. hypothesis test methods; 2. methods to analyze sentence structure; and 3. tree metric learning.

Various methods have been proposed for two-sample hypothesis tests in metric spaces, and nonparametric methods that can be applied to complex metric spaces have recently attracted attention. Mielke and Berry (2007) introduced methods belonging to the so-called Multi-Response Permutation Procedure, some of which are capable of performing two-sample tests on metric spaces. One such potential method was recently proposed by Liu *et al.* (2022), but the test is computationally expensive at $O(K^3BS)$, where K is the resampling size, B is the bootstrap size, and S is the number of sampling repetitions, and it would not scale for large samples. Furthermore, the method is based on a costly distance that was originally proposed in the article. Because we use the Wasserstein distance to measure the similarity between sets, a statistical test based on it is better suited and is a natural extension.

It is important to emphasize that when evaluating a hypothesis test's performance, the power, or the "absolute" measure of rejecting the hypothesis, is usually evaluated. In contrast, our motivation in this study is a "relative" comparison of the similarity of two sets through p-values. When using the distance between sets directly, the similarity as a distance usually depends on the sets' parameters, such as the sample size. Therefore, the judgment via a statistical test using p-values based on this distance should ideally be stable across parameters.

A recent work on the Kernel Projected Wasserstein Distance (KPWD) (Wang *et al.*, 2022) also proposed a two-sample test method based on the Wasserstein distance through kernel projection, which fits the above objective. Although the method has good power performance, for two samples of sizes n and m, it costs $O(Bd^3J^3log(J))$, where $J = max(n, m)$ and d is the dimension of the KPWD's projected space. Furthermore, it is non-trivial to design a kernel adapted to real-world trees whose distances are measured by the tree edit distance (TED). Instead, we need a simpler, more lightweight method that scales, and that reflects the direct features of the distance in the metric space without using an additional method such as kernel functions or projection methods.

From another perspective, of our application to sentences, there are two kinds of related work with respect to analyzing the tree structure of natural language sentences. Both are supported by tree-annotated sentence corpora, called treebanks (Marcus *et al.*, 1993; Nivre *et al.*, 2020), which we use here, too. The first field is parsing in natural language processing, which seeks to reverse engineer the sentence structure from a sequence. For the example of "I saw a girl with a telescope," the tree structure would define

the sentence's meaning in terms of whether the subject or the girl has the telescope. Before machine learning, it was necessary to parse a sentence for input to an application such as machine translation. However, today's large language models capture sentence structure, as will be shown later in this paper, and the importance of the parsing task itself has changed. Furthermore, parsing has a focus on the processing performance, but the performance does not elucidate a qualitative mathematical understanding of sentence structure.

The second field is quantitative linguistics, in which the characteristics of sentence structure have been sought by the use of treebanks, with a focus on the dependency length and direction. For length, a sentence's minimal dependency length has been assumed to be *optimal*, and the optimality of sentence structure has been investigated. A good summary is found in Liu *et al.* (2017). As for the dependency direction, linguistic theory since Greenberg universals (Greenberg, 1963; Dryer, 1992) has found statistical verification with corpora (Liu, 2010). Those previous works analyzed sentence structure via certain attributes that were mainly acquired from the modifier-modified relation. In contrast, this work takes a different basis by reconsidering the characteristics of sentence structure via general trees in a tree metric space.

Lastly, this work has a contrasting relation with the genre of studies called metric learning. Originally, the main target of tree metric learning was phylogenetic trees, for which tree metrics (Buneman, 1971, 1974) were proposed and used to acquire a phylogenetic tree as a "minimum evolution." A metric was trained via metric learning in Bernard *et al.* (2006); Boyer *et al.* (2007), which comprise the origins of tree metric learning. The concept has since been leveraged to implement unsupervised parsing (Parikh *et al.*, 2014), which trains a metric from the part of speech and word kind. The resulting metric remains inexplicit, with limited accuracy due to the unsupervised setting.

Tree metric learning aims to minimize the model's distance to real-world trees, and the result is always evaluated via the model's performance. Tree metric learning could provide insight for understanding real-world trees, but it does not directly answer questions such as "what kind of bias underlies real-world trees?", "how exactly does real-world data differ from random trees?", or "what statistical characteristics do real-world trees possess?", because of the fact that metric learning tunes the metric itself. We believe that tackling these questions requires starting from a solid metric that is applicable to any tree, including random trees. Hence, we use the most standard TED, which is much studied and well known in the field of geometry. Hence, the metric is not learned in this work, but such metric learning could be incorporated as a future work.

3 Sentence Tree Metric Space

3.1 Tree Representations

Let $t = (V, E)$ denote a rooted ordered directed tree, where V is the set of nodes and $E \subset V \times V$ is the set of edges. An ordered tree (also known as a plane tree) is a rooted tree with an ordering specified for the children of each node. Let T denote the set of finite rooted ordered directed trees. Let n be the number of nodes of a tree $t \in T$. Each node $v \in V$ is characterized by the order among its siblings, as given by a function order(v), and v can also be labeled, by label(v).

In our analysis of sentence structures, we consider universal dependency (UD) structures (Nivre *et al.*, 2020) as will be mentioned in Section 5. The root is the sentence head, and an edge represents a relation between two words, from the head to one of its modifiers. An example of size $n = 10$ is shown in Figure 1(a). A node corresponds to a word via label(v), and a branch runs from the head to each of its modifiers. The values of order(v) are the italicized numbers in each node.

The options for the ordered tree representation derive from the original dependency structure, with each having different label(v). Let ith(v) denote the offset of the word for node v in the sentence. For example, for the node of "board," ith$(v) = 5$. Furthermore, let v's head (i.e., its parent, or the modified)

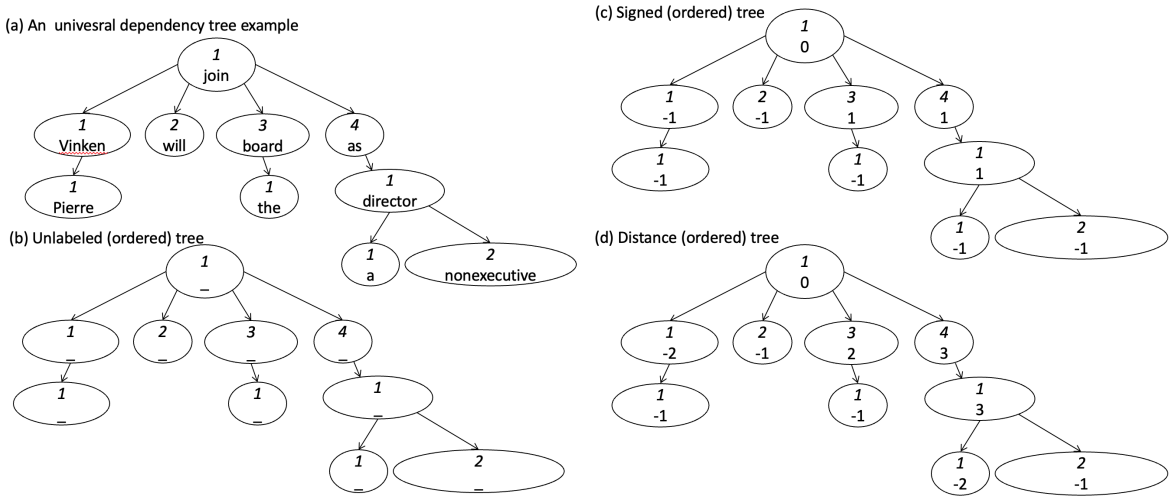

Figure 1: Dependency structure examples of (a) a sentence's dependency tree, and its (b) ordered unlabeled tree, (c) ordered signed tree, and (d) ordered word-distance-labeled tree.

be represented by head(v). For example, head("board") = "join." Then, two optional representations can be considered as follows:

(b) Unlabeled ordered tree: label$(v) \equiv$ _.

(c) Signed ordered tree: label$(v) \equiv$ sgn(ith$(v)-$ith(head(v))), where sgn(z) is defined as 1 when $z > 0$ or as -1 when $z < 0$, for $z \in \mathcal{Z}$ (integer). When v is the sentence head, label$(v) \equiv 0$.

For a fundamental comparison with random trees, as will be introduced later, (b) must be used, because a tree structure does not have any linear ordering of words, i.e., sentences. On the other hand, when there is such alignment, (c) captures the most basic structural differences.

There are many other possibilities beyond (c), such as capturing the distance in the modifier-modified structure as follows and as shown in Figure 1(d):

(d). Distance-ordered tree: label$(v) \equiv$ ith$(v) -$ ith(head(v)).

Furthermore, this representation can incorporate the word kind, and other attributes such as the part of speech. However, the more concrete a tree representation becomes, the sparser it becomes in the embedded space. Such an approach leads to another direction of metric learning, which aims to acquire the best metric, as introduced above. In contrast, this work considers a method to analyze real-world trees in a tree metric space.

Hence, in this work, we will use the most basic representations (b) and (c), where (b) is only used when random trees without linear alignment are unavailable.

3.2 Tree Edit Distance

A metric space is a pair of a set T and a distance function that defines a metric $m : T \times T \to [0, \infty)$. For the distance function to form a metric, it must fulfill the distance principles of nonnegativity, self-equality, discernibility, symmetry, and triangular inequality.

Under the theory of a tree metric space, the most basic distance between trees is the tree edit distance (TED) (Bille, 2005), which is defined as follows:

$$m(t_1, t_2) = \min_{(e_1,\ldots,e_j)\in\Gamma(t_1,t_2)} \sum_{i=1}^{j} c(e_i). \tag{1}$$

Here, $\Gamma(t_1, t_2)$ is the set of sequences of editing operations e_i, $i = 1, \ldots, j$, each of which is either a deletion, insertion, or replacement of a node. In addition, $c(e_i) \geq 0$ is a cost function for each editing operation e_i. Throughout this article, we use a cost function $c(e) \equiv 1$ for tree options (b) and (c). There is an open question of how to design $c(e)$. That question could lead to changing the metric itself, as with the Mahalanobis distance (Xing *et al.*, 2002) or pq-grams (Shindo *et al.*, 2020). As mentioned previously in Section 2, the highlight of such works becomes the performance of the metric itself.

The TED has seen possible extensions (Feragen *et al.*, 2013), especially in regard to whether a branch has a (continuous) specific length. See Bille (2005) for a survey on various TED versions and related problems and algorithms. Here, even for tree representations as defined above, the TED fulfills the distance principles, as briefly explained in Appendix A.

For example, consider two sentences, both with $n = 5$: t_1 = "Vinken will join the board"; t_2 = "Mary sings a beautiful song." As the sentences have entirely different words, one modifier structure (corresponding to the nodes "will" and "beautiful") must be reattached to the main verb, which requires one deletion and insertion. Therefore, $m_{(a)}(t_1, t_2)$ comprises 4 label replacements, 1 deletion of "will," and 1 insertion of "beautiful," totaling 6 operations.

However, the two sentences' structures share some common parts, and options (b) and (c) yield smaller distances. For (b), because no labels must be replaced, the unlabeled node "will" must be deleted and the unlabeled node "beautiful" inserted, thus giving $m_{(b)}(t_1, t_2) = 2$. As for (c), the TED value is 2.

In this article, the TED is calculated by ZSS (Zhang and Shasha, 1989), an efficient dynamic programming algorithm for ordered trees. Given two trees t_1 and t_2 of sizes n_1 and n_2, let their numbers of leaf nodes be l_1, l_2 and their depths be d_1, d_2, respectively. The computational efficiency of $m(t_1, t_2)$ is $O(n_1 n_2 \min(d_1, l_1) \min(d_2, l_2))$ (Zhang and Shasha, 1989). To analyze T, it is necessary to calculate a symmetric distance matrix M of dimension $|T|$, which records all possible distances between every pair of elements in T. Because of this quadratic complexity, it becomes necessary to sample from the dataset. In the following, the resampling size is denoted as K.

Furthermore, in calculating $m(t, t')$ in $t, t' \in T$, when t and t' have different lengths, a smaller t tends to be to closer to other t' or to the geometric median than to a larger t. This is because the distance between two large trees with different structures requires both elimination and insertion, whereas that between a small tree and a large tree mainly requires insertions to construct the long sentence. Therefore, the differences in sentence length (i.e., tree size) reflect the outcome of $m(t, t')$. Hence, to highlight the structural difference, sentences (trees) of the same fixed length N are compared in this work.

Hence, for the following analyses, T comprises I sets of trees, denoted as $T_1, T_2, \ldots, T_I$. The corpora used in this article will be introduced in Sections 5 and 6, for natural language and random sentences, respectively. From each set of trees, the statistical test involves K resamples, and the distances among the IK trees are calculated via the TED to obtain the distance matrix M.

4 Statistical Test for Two Sets of Trees in a Tree Metric Space

Unlike in a Euclidean space, the geodesic between two trees t_1 and t_2 is not unique. A nonunique geodesic essentially makes the concept of the *mean* nonunique, with the mean not necessarily corresponding to a particular tree. Instead, we use the concept of the *geometric median*, defined as the tree with the smallest sum of distances to all other nodes, which is also nonunique. When $m(t_1, t_2) > 1$ for $t_1, t_2 \in T$, there are multiple geodesics with respect to the order of editing the trees.

4.1 Distance Between Two Sets of Trees

For such a space, we define the distance between two sets of trees T_1, T_2, quantifying how close they are, and we thus define a distance $dis(T_1, T_2)$ between the two sets.

There are multiple options for this function dis; because we defined a metric m between a pair of trees, dis should be based on m. The options include the distance between two geometric medians, or the Hausdorff distance, but these represent set distances between two particular trees, whereas the distribution of trees within the tree space should also be incorporated. Another candidate is the group average, but as argued in Appendix B, that function does not become zero when $T_1 = T_2$. Alternatively, a good function capturing the distributional difference among trees within the space is the Wasserstein distance, as defined in Appendix B.

4.2 Statistical Test of Equality Between Two Sets of Trees

The distance between two sets of trees depends on both the sample size K and the sentence lengths. Therefore, we seek a more conclusive way to evaluate the differences between sets of trees, considering the finiteness of K.

We propose to start from the p-value of the permutation test (Good, 2013; Pesarin, 2010), a natural test for considering the population equality of datasets, as an indicator for comparing the differences between two sets of trees. Because the p-value can be interpreted as a probability value, values may even be comparable for different datasets in different distance spaces.

Moreover, as frequently happens with nonparametric tests that do not reduce the test statistic's dimension, if the power is too large, the p-value will be zero in most cases, making it impossible to use as a comparison measure. Therefore, in this case, we fix K to a relatively small value and bootstrap resample from both tree sets.

Let T be the set of all possible trees in the given setting, and assume that two sets T_1 and T_2 are sampled i.i.d. from probability distributions P_1 and P_2 on T, respectively. The sizes of T_1 and T_2 need not be the same. Here, the null hypothesis H_0 is that the two probability distributions are equal, i.e., $H_0 : P_1 = P_2$. H_0 is tested by the following procedure.

1. (Re)sample $\tilde{T_1} \subset T_1$ and $\tilde{T_2} \subset T_2$, both with size K.
2. Calculate $d^* = dis(\tilde{T_1}, \tilde{T_2})$.
3. Merge, randomize, and separate the trees in the two sets to form new T_1' and T_2', each of size K, yielding $d = dis(T_1', T_2')$.
4. Randomize step 3 B times to yield B distances. Sort the distances in descending order to obtain $D = (d_1, \ldots, d_B)$, where $d_1 \geq \ldots \geq d_B$.
5. Find the rank of d^* in $D \cup \{d^*\}$, and denote it as r. If d^* equals one of the d_i, then the rank of d^* is set as $i + 1$.

By repeating the above procedure S times, we obtain the p-value as the average of $r/(B + 1)$ across S.

If the goal is to achieve a high-power test, a good strategy is to make K as large as possible. However, as our goal is a relative comparison of the differences between tree sets by using p-values, we do not want to have high power and too small p-values overall. Moreover, we use the bootstrap average of p-values over S iterations, because the results can vary depending on the resamples $\tilde{T_1}$ and $\tilde{T_2}$ if K is not large enough. As a result, the proposed method is a double resampling algorithm of bootstrap and permutation tests.

The computational cost for this algorithm is $O(BSK^2 \log K)$, because computation of the Wasserstein distance between samples of size K by Sinkhorn's algorithm costs $O(K^2 \log K)$ (Peyré and Cuturi, 2019). In contrast, the implementation of Liu *et al.* (2022) barely works in $O(K^3 BS)$ for our real-world case, and there lies our work's motivation.

4.3 Mathematical Characteristics of the Proposed Statistical Test

Let T be the set of all possible trees and P_1 and P_2 be two probability distributions on T. Mutually independent random sequences $X_1, \ldots, X_n \overset{\text{i.i.d.}}{\sim} P_1$ and $Y_1, \ldots, Y_m \overset{\text{i.i.d.}}{\sim} P_2$ represent sets of sampled trees T_1 and T_2, respectively. The s-th bootstrap resampling sets of size K are represented by $\tilde{X}_1^{(s)}, \ldots, \tilde{X}_K^{(s)} \overset{\text{i.i.d.}}{\sim} \hat{P}_X$ and $\tilde{Y}_1^{(s)}, \ldots, \tilde{Y}_K^{(s)} \overset{\text{i.i.d.}}{\sim} \hat{P}_Y$, where $\hat{P}_X$ and $\hat{P}_Y$ are the empirical distributions of $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$, respectively. For now, we focus on one resampling set and remove the indices $\cdot^{(s)}$ for simplicity.

The next step is to formulate the permutation test. We define it as

$$Z = (Z_1, \ldots, Z_{2K}) := (\tilde{X}_1, \ldots, \tilde{X}_K, \tilde{Y}_1, \ldots, \tilde{Y}_K), \tag{2}$$

and we consider a sequence of permutations, $\pi^{[b]} : \{1, \ldots, 2K\} \to \{1, \ldots, 2K\}$, for $b = 1, \ldots, B$. Then, we define $Z^{[b]} := (Z_{\pi^{[b]}(1)}, \ldots, Z_{\pi^{[b]}(2K)}) =: (V_1^{[b]}, \ldots, V_K^{[b]}, W_1^{[b]}, \ldots, W_K^{[b]})$, and we denote the empirical distributions of $V_1^{[b]}, \ldots, V_K^{[b]}$ and $W_1^{[b]}, \ldots, W_K^{[b]}$ as $\hat{P}_V^{[b]}$ and $\hat{P}_W^{[b]}$, respectively.

For simplicity, we set $\hat{P}_V^{[0]} := \hat{P}_{\tilde{X}}$ and $\hat{P}_W^{[0]} := \hat{P}_{\tilde{Y}}$. For the null hypothesis $H_0 : P_1 = P_2$, the estimator for the permutation test's p-value is represented as follows:

$$\hat{p} = \hat{p}(\pi^{[1]}, \ldots, \pi^{[B]}) := (B+1)^{-1} \sum_{b=0}^{B} \mathbf{1}\left(\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]}) \geq \mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}}) \right), \tag{3}$$

where $\mathbf{1}(\cdot)$ is the indicator function. For the exact permutation test, $B = \binom{2n}{n}$, and $\pi^{[1]}, \ldots, \pi^{[B]}$ comprise all the permutations. We denote the estimated p-values as $\hat{p}^*$ under this setting. Calculations for all possible permutations are typically infeasible, and Monte Carlo sampling is thus used for approximation with the random permutations $\pi^{[1]}, \ldots, \pi^{[B]}$.

The following proposition indicates that if n, m, and K are sufficiently large, then the difference between the p-Wasserstein distances $\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}})$ and $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]})$ has sufficient information on whether distributions P_1 and P_2 are equal or not.

Proposition 1. *Assume $n, m \geq K$ and $K \to \infty$.*
(1) If $P_1 = P_2$, then $\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}}) = O_p(K^{-1/2p})$ and $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]}) = O_p(K^{-1/2p})$ for each b.
(2) If $P_1 \neq P_2$, then $\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}}) = \mathcal{W}_p(P_1, P_2) + O_p(K^{-1/2})$ but $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]}) = O_p(K^{-1/2p})$ for each b.

Proof. (1) As the space T is finite, the results of Sommerfeld and Munk (2017) for empirical Wasserstein distances on finite spaces are applicable. By their Theorem 1(c),

$$\mathcal{W}_p(\hat{P}_X, \hat{P}_Y) = O_p\left(\{nm/(n+m)\}^{-1/2p} \right).$$

Meanwhile, because $\tilde{X}_1, \ldots, \tilde{X}_K \overset{\text{i.i.d.}}{\sim} \hat{P}_X$ and $\tilde{Y}_1, \ldots, \tilde{Y}_K \overset{\text{i.i.d.}}{\sim} \hat{P}_Y$, their Theorem 1(d) implies that

$$\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}}) = \mathcal{W}_p(\hat{P}_X, \hat{P}_Y) + O_p(K^{-1/2}).$$

These two equations and the fact that $nm/(n+m) \geq \min(n, m)/2 \geq K/2$ yield the first statement.

The second equation in statement (1) holds even without assuming $P_1 = P_2$. Because the space T is finite, it is possible to treat the probability functions as vectors. Hence, despite potential abuse of notation, both are represented with the same symbols. Let $\hat{H} := (\hat{P}_{\tilde{X}} + \hat{P}_{\tilde{Y}})/2$ and $H := (P_1 + P_2)/2$. Then, Theorem 3.8.2 of van der Vaart and Wellner (1996) implies that

$$\sqrt{K}(\hat{P}_V^{[b]} - \hat{H}) \rightsquigarrow 2^{-1/2} G_H \text{ for almost all } \tilde{X}, \tilde{Y},$$

where $\rightsquigarrow$ denotes convergence in distribution, and G_H is a Brownian bridge (on a finite space) corresponding to the measure H. That is, G_H follows a $|T|$-dimensional normal distribution $N(0, \Sigma(H))$ whose (co)variances are defined by $(\Sigma(H))_{ii} = H_i(1 - H_i)$, and $(\Sigma(H))_{ij} = -H_i H_j$ for $i \neq j$. By a similar argument to the proof of Theorem 1(a) in Sommerfeld and Munk (2017),

$$\sqrt{K}\mathcal{W}_p^p(\hat{P}_V^{[b]}, \hat{H}) \rightsquigarrow \text{(some distribution)}$$

as $K \to \infty$, owing to the Hadamard differentiability of $(v, w) \mapsto \mathcal{W}_p^p(v, w)$. This implies that $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{H}) = O_p(K^{-1/2p})$. Because the same holds for $\hat{P}_W^{[b]}$, we conclude that $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]}) = O_p(K^{-1/2p})$.

(2) By Theorem 1(d) of Sommerfeld and Munk (2017),

$$\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_{\tilde{Y}}) = \mathcal{W}_p(\hat{P}_X, \hat{P}_Y) + O_p(K^{-1/2}) = \mathcal{W}_p(P_1, P_2) + O_p(K^{-1/2}).$$

Finally, the second equation in statement (2) was already proved in part (1) above, because $P_1 = P_2$ was not assumed there. □

Note that our statement incorporates terms denoted by O_p, yet they can be explicitly expressed via Gaussian processes, similarly to Theorem 1 in Sommerfeld and Munk (2017). This ensures that the bound of this convergence order is tight. Such precision is particularly valuable because we seek to control the discrepancy between $\mathcal{W}_p(P_1, P_2)$ and $\mathcal{W}_p(\hat{P}_V^{[b]}, \hat{P}_W^{[b]})$ by carefully selecting the bootstrap sample size K.

Note also the significance of the finiteness of tree spaces for the order of probability convergence. As proved in Weed and Bach (2019) and summarized in Panaretos and Zemel (2019), for non-finite spaces such as a d-dimensional Euclidean space, or more generally for regular spaces, the convergence order of the empirical distribution to the true distribution based on the Wasserstein distance becomes $O_p(K^{-1/d})$. Therefore, for high-dimensional spaces, the convergence can be quite slow.

We now proceed to evaluate B, the size of Monte Carlo sampling in the permutation test. When conditioned on $\tilde{X}$ and $\tilde{Y}$, the value of the summation in (3), excepting $b = 0$, follows a binomial distribution with mean $\hat{p}^*$. Thus, for a sufficiently large B, a binomial proportional confidence interval for $\hat{p}$ at a confidence level γ can be approximated by

$$\left[\hat{p} - z\sqrt{\frac{\hat{p}(1-\hat{p})}{B}}, \hat{p} + z\sqrt{\frac{\hat{p}(1-\hat{p})}{B}}\right],$$

where z is the $(1+\gamma)/2$ quantile of a standard normal distribution. This confidence interval is valid even when the null hypothesis does not hold, and it serves as an indicator for determining the size B of the Monte Carlo sampling.

A final remark pertains to another reason for setting K smaller than n or m. According to Theorem 2 of Sommerfeld and Munk (2017), if $K = n$, the bootstrap subsamples $\tilde{X}$ cannot faithfully simulate the sampling distribution, in the sense that the law of $\mathcal{W}_p(\hat{P}_{\tilde{X}}, \hat{P}_X)$ may fail to converge to the law of $\mathcal{W}_p(\hat{P}_X, P_1)$ even after adequate rescaling. This limitation arises from the Wasserstein distance's lack of directional Hadamard differentiability, which distinguishes it from the conventional bootstrapping theory for the Euclidean norm.

However, such inconsistency can be resolved by setting $K/n \to 0$ as $K, n \to \infty$, as was originally shown by Dümbgen (1993). This fact implies that if we set K sufficiently smaller than n or m, then we can expect the distribution of bootstrap samples to be reasonable for approximating the original sampling distributions, even when using the Wasserstein distance to measure their discrepancies.

Table 1: Treebanks and the numbers of sentences used in this work, along with the numbers of sentences for $N = 10, 20$. The fifth and sixth columns give the average and standard deviation of the Wasserstein distance from English-EWT, for the unlabeled and signed cases, respectively, across $S = 10$ trials for $N = 20$ and $K = 200$. The last column gives the p-value of the statistical test for the null hypothesis H_0 with $S = 50$, $B = 100$, $K = 50$, and $N = 20$, for the signed case if available, or unlabeled otherwise.

treebank	# all	# (N=10)	# (N=20)	Unlabeled	Signed	Statistical Test p-value
English						
English-EWT	16621	633	427	2.911±0.107	3.783±0.073	0.958
English-Atis	5432	570	70	7.211±0.062	9.014±0.050	0.0
English-ESL	5124	213	207	4.978±0.078	6.365±0.071	0.19
English-EWT re-parsed by parsers						
English-EWT-Spacy	16621	633	427	6.262±0.124	8.409±0.154	0.048
English-EWT-UDPipe	16621	633	427	4.413±0.059	5.922±0.132	0.365
Non English						
Arabic-NYUAD	19738	274	385	6.413±0.044	10.326±0.122	0.0
Belarusian-HSE	25231	1437	435	5.818±0.055	7.211±0.062	0.031
Catalan-AnCora	16678	240	389	5.231±0.054	7.121±0.072	0.025
Czech-PDT	87913	3214	2771	5.776±0.098	7.655±0.088	0.065
Estonian-EDT	30972	1671	859	5.800±0.081	7.896±0.132	0.0
French-FTB	18535	307	478	5.212±0.072	7.405±0.079	0.021
German-HDT	189928	6127	7421	5.389±0.053	7.536±0.077	0.018
Hindi-HDTB	16647	529	719	7.410±0.113	11.940±0.160	0.0
Icelandic-IcePaHC	44029	1782	1197	6.320±0.086	9.380±0.126	0.0
Italian-ISDT	14167	595	375	5.237±0.062	6.977±0.077	0.057
Japanese-BCCWJ	57109	1968	1424	7.800±0.098	14.155±0.133	0.0
Korean-Kaist	27363	1988	932	9.780±0.130	13.419±0.101	0.0
Latvian-LVTB	15984	213	499	5.962±0.080	7.709±0.103	0.018
Norwegian-Bokmaal	20044	914	602	5.128±0.078	6.788±0.107	0.274
Persian-PerDT	29107	1676	952	7.138±0.104	10.571±0.113	0.0
Polish-PDB	22152	1305	536	5.898±0.078	8.220±0.106	0.0
Portuguese-GSD	12019	277	375	5.354±0.070	7.162±0.069	0.021
Romanian-Nonstandard	26225	703	983	5.279±0.066	7.269±0.115	0.013
Russian-SynTag	87336	3715	2668	5.893±0.107	7.705±0.092	0.012
Spanish-Ancora	17662	319	360	5.314±0.034	7.057±0.062	0.04
	1000	500	500	9.639±0.128	-	0.0
Rand-Markov (order=2)	1000	500	500	16.887±0.202	-	0.0
Rand-Markov (order=3)	1000	500	500	13.468±0.144	-	0.0
Rand-Markov (order=5)	1000	500	500	11.670±0.141	-	0.0
Rand-Markov (order=10)	1000	500	500	10.423±0.129	-	0.0
Rand-English-EWT-CF	100000	2560	935	6.491±0.058	8.944±0.098	0.0
Rand-ChatGPT	23882	446	1073	5.270±0.095	6.716±0.085	0.111

5 Universal Dependency Dataset

This work considers the Universal Dependency (UD) dataset (Nivre *et al.*, 2020), specifically the v. 2.9 dataset, which includes 218 corpora from 125 languages, totaling 1,839,131 sentences. Table 1 summarizes the data. Many UD corpora have training, development, and test subsets. Because this work does not take a usual machine learning approach to data, all three subsets are conjoined for the analysis here.

For the main data in this work, English-EWT is used because it comprises written text acquired from English Web data. This main treebank is compared with other kinds of English treebanks, which are listed in the first block of Table 1.

Later, in Section 9, corpora in multiple languages will be analyzed. Among the 218 corpora in the

Universal Dependency dataset, many are too small to acquire a sufficient number of samples, K, of a fixed length N. Because the size of English-EWT is 254 KB, we selected corpora by the following criteria:

- The corpus size must be larger than 250 KB.
- The corpus must consist of sentences in a written language (rather than speech data, parallel data, or another specific kind).
- The language must be in use today (rather than Latin or other barely used languages).
- One corpus is chosen for a particular language.

This filtering yielded the 20 corpora listed in the third block of Table 1. The second column in the table lists the number of sentences in each corpus, followed by the numbers of sentences of lengths $N = 10, 20$, which we consider the most in this article. The fifth and sixth columns list the Wasserstein distances from English-EWT for the unlabeled and signed cases, respectively. The fourth block shows the results for random trees. All of these corpora and results are explained and discussed below.

6 Random Trees

Comparison with synthetic random trees serves to illuminate the nature of tree structures. Let $g(n-1)$ denote the number of ordered unlabeled trees of size $n-1$. Then, for size n, an additional node could be attached to any of the $n-1$ nodes, thereby giving $ng(n-1)$ possibilities. As $g(2) = 1$, the total number of unlabeled ordered trees is $g(n) = (n-1)!$. The two extreme cases of unordered trees are a *linear* tree and a *hub* tree. In a linear tree, each node except the leaf node always has a single child, whereas a *hub* tree has a common root node with all other nodes as its children.

In the following two sections, we consider five different sets of random trees. In each set, the order of the child nodes is the order of their generation.

6.1 Basic Random Trees

An unlabeled ordered tree has a structure between the structures of linear and hub trees of the same size. The following two basic trees are naturally considered:

Rand-Uniform: The nth node is attached uniformly randomly to one of the tree's $n-1$ nodes. This generates a random tree among unlabeled ordered trees.

Rand-Markov: For $j > 0$, the nth node is attached to one of the j previous nodes inserted in the tree. The case of $j = 1$ is a linear tree, while the cases of $j = 2, 3, 5, 10$ are examined in this article.

These trees are constructed recursively by inserting the nth node in a tree of size $n-1$. Here, 500 samples were generated for each required size. These trees appear in the first five rows in the last block of Table 1.

6.2 Random Context-Free/Sensitive Trees

Another type of random tree is generated by sampling some aspect of a real corpus. We consider two kinds of corpora: context free and context sensitive.

6.2.1 Random Context-Free Trees

A tree is *context free* if a node samples children *independently* of its siblings (if any exist).

Our set of context-free random trees was generated by using a grammar acquired from English-EWT. The procedure comprises grammar construction and tree generation, as follows:

1. **Grammar Construction:** A dependency grammar is built for English-EWT by examining every word and recording what other words modify it. As a word w can be modified by words before or after itself, the sets of modifying words *before* and *after* w are recorded with their frequency counts. All the numbers of modifiers occurring *before* and *after* are also recorded for each w. The resulting sets of modifier-modified relations before and after w are denoted as Gb and Ga, respectively. In other words, for a word w, Gb and Ga each comprise the following:

   X a list of the numbers of modifiers, and

   Y pairs of a modifier word and its frequency.

   Furthermore, all main predicate words are collected from the corpus together with their frequencies; the resulting set is denoted as H.

2. **Tree Generation:** A main predicate is sampled from H in proportion to its frequency. Then, a random sentence is generated recursively by using a function F, starting with the main predicate w as the target word. For every target word w, from each of Gb and Ga, F first samples the number of modifiers (from X above), and it then samples the corresponding number of modifiers in proportion to the frequency of the number of occurrences (from Y above). For each modifier generated in this way as a new target, the function F is called recursively. This recursive procedure stops when the number of modifiers (sampled from X) is zero.

In the grammar construction, the list X from Gb and Ga for every word w has many zeros, each of which corresponds to w occurring in the tree without a modifier. The recursive procedure thus terminates when a zero is sampled. Accordingly, the procedure can terminate without introducing any arbitrary setting such as a stop probability, as was introduced in Klein and Manning (2004).

Because the generated trees have words and dependency directions, any of the tree representations described in Section 3.1 can be applied to Rand-English-EWT-CF. As indicated in the sixth row in the last block of Table 1, 100,000 trees were generated.

6.2.2 Random Context-Sensitive Trees

By changing Rand-English-EWT-CF slightly, context-sensitive trees can be generated[1]. None of these simplest context-sensitive trees, however, is any closer to sentences via our framework, and we thus do not report any results for those trees.

In contrast, recent generative AI produces sophisticated random data, with almost quasi-human-level sentences. If the output's tree structure is statistically tested for equality with natural language sentences,

[1] This entails certain choices, as follows:

- A child is generated if its parent has a particular sibling, found in English-EWT.
- The numbers of branches are sampled for each child node.
- The tree shape is incorporated: for the English case, the last branch tends to recursively bifurcate; therefore, the position among the siblings is considered when generating random trees.

These options would position the sampled trees in the tree metric space slightly closer to those of English-EWT than to those of Rand-English-EWT-CF. Although we tested all of these improvements, none of them made the set significantly closer to English-EWT than to Rand-English-EWT-CF. Because we already have multiple options, we do not deal any further with these possibilities in this article.

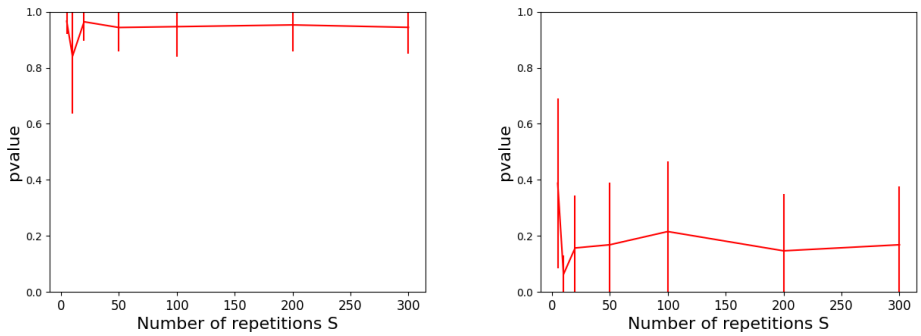

Figure 2: Mean and standard deviation of the p-value with respect to S, for $B = 100$, $N = 20$, $K = 50$.

generative AI can be judged as capturing the characteristics of natural language sentences, from a tree-structure perspective. We believe that this is important, because recent neural models capture long-range dependence, which partially derives from tree structure, in contrast to previous language models that were based on Markov models.

The well-known ChatGPT by OpenAI was constructed with a transformer (GPT) technique (Brown *et al.*, 2020). ChatGPT rephrases a given text or generates an answer for a given question. Because rephrasing produces texts that are limited by the original input phrase structure given by humans, we generated texts by prompting ChatGPT with questions.

To construct texts from ChatGPT, we needed questions. We thus downloaded questions from SQuAD (Rajpurkar *et al.*, 2016), a dataset for question answering in the field of natural language processing. Only the questions were fed to ChatGPT via the OpenAI API, resulting in one to multiple sentences per question. Some examples are given in Appendix C. The resulting sentences were automatically parsed by UDPipe (Straka, 2018; Straka and Straková, 2020)[2]

The parser quality influences the results. We thus tested two parsers for English, as described in Appendix D, and we chose UDPipe, which parsed the English-EWT sentences into parsed trees that were not distinguishable from the original English-EWT according to our statistical test method; see the second block of Table 1 (last column). Appendix D extends this preliminary discussion with more detail. The generated text had around 24,000 sentences.

All data that was generated by random models is prefixed by "Rand-" in this article.

7 Empirical Characteristics of the Statistical Test

From the next section, we compare multiple sets of trees by applying the proposed statistical test described in Section 4.2. Thus far, there are four hyper-parameters: S, the number of repetitions; B, the bootstrap sample size; K, the number of (re)sampled trees for each repetition; and N, the sentence length. As mentioned previously, the distance between two sets depends on K and N. Therefore, although the distance captures the relative relations among sets well for the comparable setting, a statistical test would serve to make a stabler judgment on whether the compared sets are similar. Hence, this section reports the basic empirical characteristics of the test for the four hyper-parameters.

We compare two pairs of sets: first, English-EWT against itself, for which the null hypothesis of the two sets' equality must be accepted; and second, English-EWT against English-ESL, where the latter set comprises English sentences written by non-natives. In the strictest sense, the null hypothesis must be rejected in the second comparison, but English-ESL has some closeness to English-EWT.

[2]https://lindat.mff.cuni.cz/services/udpipe.

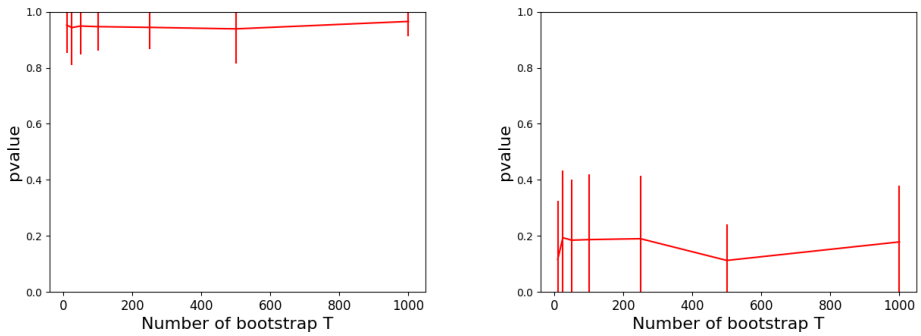

Figure 3: Mean and standard deviation of the p-value with respect to B, for $S = 10$, $K = 50$, $N = 20$.

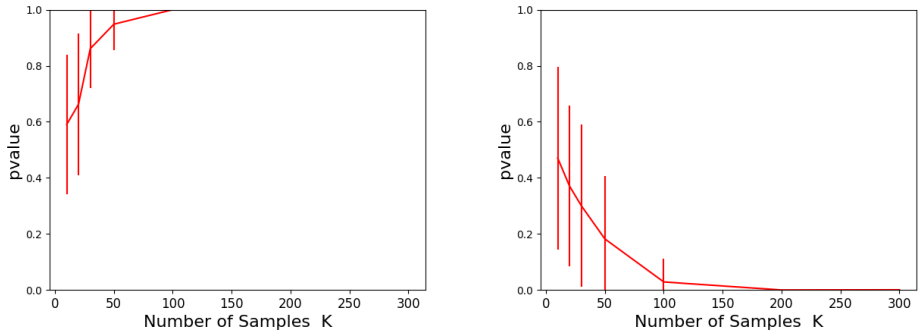

Figure 4: Mean and standard deviation of the p-value with respect to K, for $S = 50$, $B = 100$, $N = 20$.

Figures 2 and 3 show the mean p-value and its standard deviation for varying values of S and B, respectively. Specifically, for $S = [5, 10, 20, 50, 100, 200, 300]$, we used $B = 100$, $K = 50$, and $N = 20$, whereas for $B = [10, 25, 50, 100, 250, 500, 1000]$, we used $S = 1$, $K = 50$, and $N = 20$. There are not many differences in the results across the settings, except for the fluctuation seen for small $S = 5, 10$. Hence, we use $S = 50$ and $B = 100$ below, unless mentioned otherwise.

The dependencies on K and N are more important. Our objective is to judge the similarity of the two sets in the tree metric space. The similarity, if directly considered by examining the Wasserstein distances between sets, depends largely on the parameters of the set samples. We want to use the test method's p-value to provide integrated judgment across settings. In our test method, K is the number of (re)sampled sentences (and random trees), and it determines the method's power. A large K yields a result that is too strong, with the p-value converging to 0 or 1. Therefore, we suppress K in order to judge the two sets' similarity via the p-value, where the judgment will stay stable.

Figure 4 shows the results for $K = [10, 20, 30, 50, 100, 200, 300]$ and $N = 20$. We can see that when K increases (horizontally), the left graph goes to $p = 1$, whereas the right graph goes to $p = 0$, although the convergence is faster on the left. These graphs show how the test makes correct judgments with increasing K. However, they also suggest that for a large K, the p-value cannot be used to judge the closeness of the two sources even for ESL-English, because p becomes 0, indicating that the two sets are different. Hence, we use a small value of $K = 50$ below. If $p = 0$ for even this K, then the null hypothesis is completely rejected.

Lastly, we consider the dependence on N, the sentence length. Figure 5 shows the results for $N = [10, 15, 20, 25, 30, 35]$ and $K = 50$: as mentioned above, a larger K would cause the p-values to converge. When N is small, the sentence structures become less varied, whereas long sentences better show the difference between two sets; however, the corpora include fewer long sentences, and the result

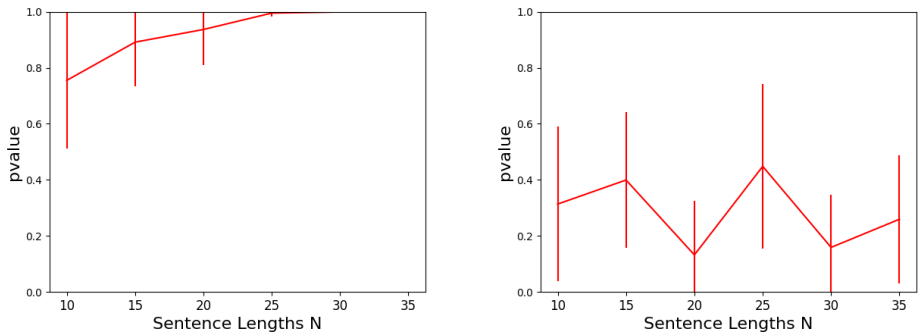

Figure 5: Mean and standard deviation of the p-value with respect to N, for $K = 50$, $S = 10$, $B = 100$.

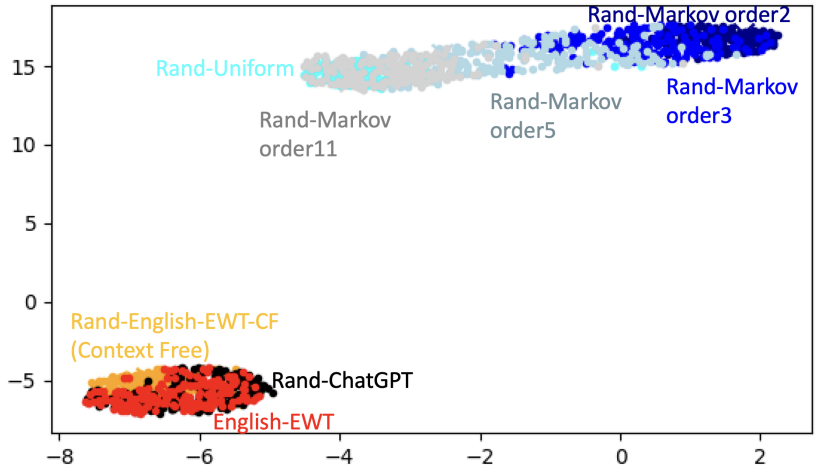

Figure 6: Tree metric space plotted via UMAP for $K = 200$ samples and sentences of length $N = 20$, for English EWT (red), Rand-Uniform (cyan), Rand-Markov (order 2 in dark blue, order 3 in blue, order 5 in light blue, order 10 in grey), Rand-English-EWT-CF (i.e., context free, yellow), and Rand-ChatGPT (black).

becomes more unstable. Therefore, N should be large enough to capture the data's characteristics, but with a sufficient number of samples. Appendix E gives the statistical characteristics of sentence lengths for English-EWT. From this discussion, we use $N = 20$ below.

8 Comparison of English-EWT with Random Trees

Appendix E reports a preliminary experiment showing the basic differences in sentence lengths, bifurcation ratios, and diameters for English-EWT in comparison with some random trees[3]. In Section 6, four different kinds of random data were introduced, as summarized in the bottom block of Table 1. The closeness of a set of random trees to the point clouds of English-EWT would indicate how well the random trees capture English's sentence structure. Because this section involves basic random trees, the comparison is conducted solely via unlabeled ordered trees, unless mentioned otherwise.

We first examine the 2-dimensional mapping results via the UMAP topological visualization technique (Mclnnes *et al.*, 2018)[4]. We chose UMAP because the tree metric space that we consider is nonlin-

[3]We placed these simple statistics in an appendix because they are not this paper's main point. Most importantly, analysis of the diameters showed that English-EWT is scattered among less than one third of all the trees.

[4]We used n_neighbors=30 and min_dist=0.1.

ear[5] Rand-Uniform, Rand-Markov (order $2, 3, 5, 10$), Rand-English-EWT-CF, and Rand-ChatGPT were compared with English-EWT by sampling sentences with length $N = 20$ for $K = 200$ trees.

Figure 6 shows the results. Each point represents a sampled tree, in red for English-EWT, cyan for Rand-Uniform, dark blue to gray for Rand-Markov, yellow for Rand-English-EWT-CF, and black for Rand-ChatGPT. It can be seen that the Markov models are outside the natural language points for English-EWT. The larger the Markov order, the more the set of trees approaches those of natural language. The uniform trees are close to the order-10 Markov trees, which is much better than for lower-order Markov models. Historically, natural language models have improved from lower- to higher-order Markov models, and this figure visualizes how that technical shift has made these models closer to natural language.

The extent to which natural language is context free or context sensitive has been debated (Pullum and Gazdar, 1982). In Figure 6, the points for English-EWT overlap those for Rand-English-EWT-CF. ChatGPT is deemed the best context-sensitive model possible, and its sentences are on the opposite side of English-EWT from Rand-English-EWT-CF. This probably suggests that ChatGPT sentences have a different nature from the simple Rand-English-EWT-CF sentences.

The averages and standard deviations of the Wasserstein distances from English-EWT are given in the fifth column of Table 1, for $N = 20$, $K = 200$, and $S = 10$ trials, while the sixth column gives the signed distance results only for Rand-English-EWT-CF and Rand-ChatGPT. The distances are consistent with the understanding gained from Figure 6. Curiously, the distance from English-EWT to Rand-ChatGPT is much smaller than that to Rand-English-EWT-CF, thus showing how Rand-ChatGPT captures English sentence structure much better.

According to our statistical test, the p-values are zero even with a weak test of $K = 50$, except for Rand-ChatGPT, as seen in the last column of Table 1 for $N = 20$, $S = 50$, and $B = 100$. For Rand-ChatGPT, the p-value is 0.11. This result provides evidence that Rand-ChatGPT sentences have similar tree structures to natural English. It also suggests that a language model, Rand-ChatGPT, has successfully performed unsupervised parsing, i.e., the acquisition of grammatical tree structure from plain texts, without grammatical annotation.

Rand-ChatGPT thus produces text with sentence structure similar to that of natural language, although there is still a discrepancy from real sentence structure. An increased K would reject the null hypothesis, and English authored by non-natives produces a larger p-value. Future language models could improve even further, however, and our method provides a means to measure such improvement.

9 Comparison Among Natural Language Data

Finally, we examine the metric space for various natural language trees. Following the previous section, the trees are visualized with UMAP and are signed, unless mentioned otherwise.

The UD dataset for one language often comprises corpora with varied kinds of data. First, as listed in the first block of Table 1, we compare English-EWT with English-Atis, a speech corpus, and English-ESL, a corpus of English for second-language learners, together with Rand-English-EWT-CF. We sampled $K = 200$ sentences from each treebank. Figure 7 shows the results for sentences with length $N = 20$. There is a total of 800 points, in red for English-EWT, blue for English-Atis, green for English-ESL, and orange for Rand-English-EWT-CF. Each$\times$ mark represents the geometric mean among the K points.

The Atis points (blue) are mostly separated from the EWT (red) and ESL (green) points, which suggests that the structure of speech is different from the written structures in EWT and ESL. Nevertheless,

[5]There is the other topological method of t-SNE, which yields almost similar results. A linear method of MDS (multi-dimensional scaling) would cause known problems, such as separating points which actually are not separated in the metric space.

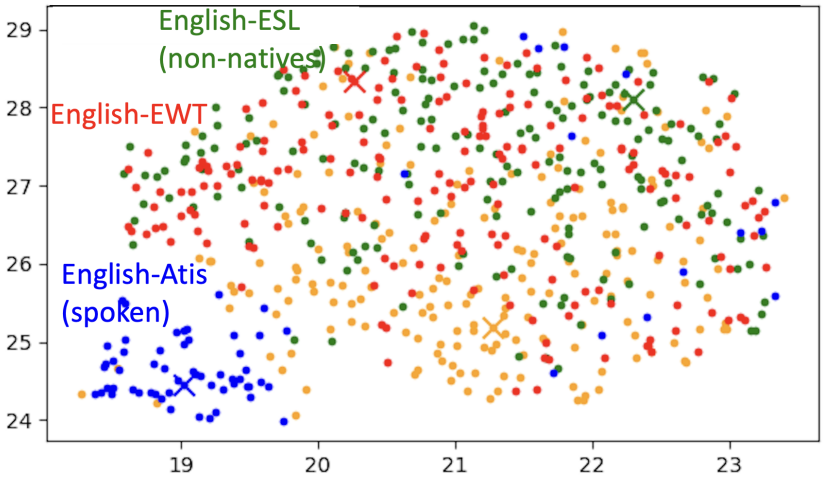

Figure 7: Tree metric space with $K = 200$ for English-EWT (red), English-Atis (blue), English-ESL (green), and Rand-English-EWT-CF (orange), as visualized by UMAP for $N = 20$.

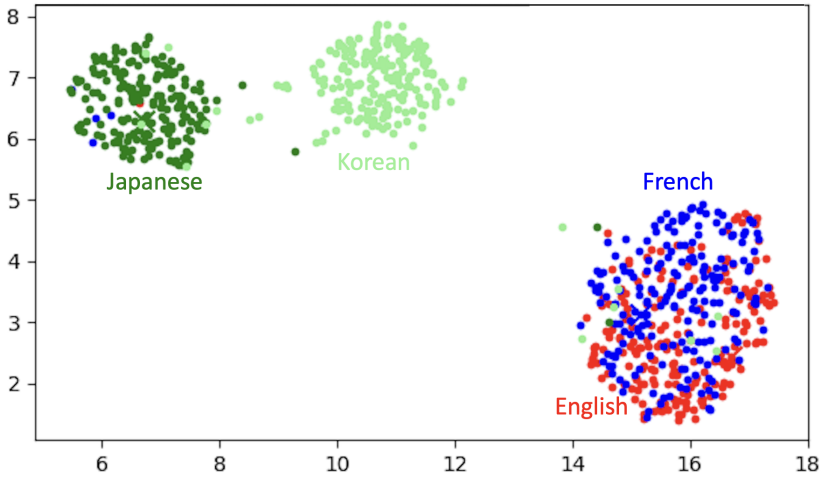

Figure 8: Tree metric space with $K = 200$ for English-EWT (red), French-GSD (blue), Japanese-BCCWJ (green), and Korean-Kaist (light green), as visualized by UMAP for $N = 20$.

some blue points appear within the region of red and green points; therefore, this mode separation is not complete, indicating that there are writing-like sentences in Atis.

On the other hand, the figure suggests that EWT and ESL are not so far apart. Many green points are located close to red points, which may indicate a small lack or insertion of words, possibly because of ESL including sentences authored by non-natives.

The concrete Wasserstein distances from English-EWT are shown in the fifth and sixth columns of Table 1, for unlabeled trees and signed trees, respectively, from $S = 10$ trials with $N = 20$, $K = 200$. For the English-EWT case, the distance is measured against itself, thus showing how there is a certain distance from a tree to a sample of itself. Obviously, the signed distance is consistently larger than the unsigned distance. The differences from English-Atis and English-ESL summarize the figure well. By conducting our statistical test for $S = 50$, $B = 100$, $K = 50$, and $N = 20$, we find that the null hypothesis is obviously accepted for English-EWT itself, yielding a p-value of almost 1. The null hypothesis is rejected for English-Atis, but not for English-ESL, as mentioned above in Section 4.2, with p=0.18. Therefore, the null hypothesis is accepted for English-ESL with a confidence interval of 0.1.

The same analysis can be conducted across different languages. Figure 8 shows the results with $K = 200$ and $N = 20$ for the English-EWT (red), French-FTB (blue), Japanese-BCCWJ (green), and Korean-Kaist (light green) datasets. The English and French sentence structures are mixed together, whereas Korean and Japanese are separated. Some of the points are intermixed, but this is deemed to derive from UMAP's limitation. Overall, the tree space captures differences in grammatical constructions.

It is natural that English and French appear together, as they are both SVO languages and have many similarities. On the other hand, Japanese and Korean are known to have similar sentence structures (Haspelmath *et al.*, 2005) as well, yet they are separated in the graph. Within the UD research community, this difference between the Japanese and Korean UD treebanks is a well-known fact. These treebanks were constructed with completely different segmentation strategies, despite Korean and Japanese largely sharing their sentence structures. Such qualitative observation would not be changed even by changing the experimental settings by differentiating the distance or having different N.

The distances between English-EWT and multiple languages are given in the fifth and sixth columns in the second block of Table 1. The values well represent the distances between languages. The last column gives the p-values for equality with English-EWT. Norwegian was accepted as similar to English, and Norwegian's word order is known to be very similar to that of EnglishHaspelmath *et al.* (2005) among Northern European languages, possibly because of their historical relation. Because this tree metric space does not consider words, such similarities are highlighted. Other neighboring languages, including French and German, yielded small p-values but were all rejected at the 10% significance level even with $K = 50$.

10 Conclusion

This article has proposed a lightweight statistical test method to analyze real-world trees in a tree metric space. In particular, we have analyzed sentences in comparison with random trees, including outputs from recent large language models. As tree comparison is already costly and sentence data is large in scale, the recent two-sample tests using metrics are too computationally expensive. Accordingly, we proposed a more lightweight, simpler method by naturally incorporating previous test methods in a double resampling algorithm comprising bootstrap and permutation tests. The test was shown theoretically to have good power, and empirical analysis showed that the method does not depend on the data's parameters.

We applied the method to examine sentence trees with respect to random trees. The random trees included those from various language sentence models that have been developed over time, including Markov models, context-free models, and the most recent large language models. Among them, only the large language models showed similarity with real sentences. This finding is important because it indicates that, for the first time in history, language technology has achieved unsupervised acquisition of sentence structure. Furthermore, we showed that our method can be applied to compare corpora within and across languages, indicating the characteristics of both the corpora and the languages.

The p-value shows that there is still a discrepancy between the sentence structure produced by large language models and that of real sentences. Our method will enable measurement of this difference for future large language models. Because the proposed method evaluates sentence characteristics, it has other possible engineering applications, such as rephrasing of sentences and comparison of sentence complexity. Finally, our method is also applicable to other kinds of data, which remains for our future work.

Appendix

A Explanation of How TED Follows Distance Principles

For the TED used here, leaf deletion and insertion are performed, in addition to label replacement. Because a leaf is a vertex of degree 1, this special version is sometimes called the 1-degree TED. There have been studies on both the labeled 1-degree TED (Selkow, 1977; Chawathe, 1999) and the unlabeled TED (Micheli and Rossin, 2005). As those references provide the detailed theory, we only briefly explain here that such a special TED also fulfills the metric axioms of nonnegativity, self-equality, discernibility, symmetry, and triangular inequality. Although only the unlabeled case is discussed here, the same argument holds for the labeled case.

Let T denote the set of finite rooted ordered directed trees. Let $\mathcal{G}_T = (T, E_T)$ be an undirected graph such that the vertices are the elements of T and two vertices are connected by an undirected edge of length 1 if they can be transformed by insertion or deletion of a leaf. Note that $\mathcal{G}_T$ is a path-connected graph because any tree can be transformed into another arbitrary tree by a finite number of repeated deletions to leave only the root node, followed by a finite number of repeated insertions. In this case, the TED we use is equivalent to the shortest path length in $\mathcal{G}_T$. Given that the shortest path length for each pair of vertices in a path-connected graph with positive edge weights becomes a metric, we can confirm that the unlabeled 1-degree TED also defines a metric on T. Therefore, if we denote this metric on T as d, then (T, d) is a metric space. If only trees of size N are considered, then we restrict (T, d) to $T_N \subset T$, the set of trees of size N.

B Wasserstein Distance Between Two Sets of Trees

Given a tree metric space (T, m) and two finite sets of trees $T_1, T_2 \subset T$, there are several measures to compute the degree of difference between them. Here, we use the p-Wasserstein metric on the empirical distributions:

$$m_{W_p}(T_1, T_2) := W_p\left(\frac{1}{n_1}\sum_{i=1}^{n_1}\delta(t_i), \frac{1}{n_2}\sum_{j=1}^{n_2}\delta(t'_j)\right), \tag{4}$$

where $T_1 = \{t_i\}_{i=1}^{n_1}$, $T_2 = \{t'_j\}_{j=1}^{n_2} \subset T$, and $\delta(t)$ is the Dirac measure at $t \in T$. Here, $\frac{1}{n_1}\sum_{i=1}^{n_1}\delta(t_i)$ and $\frac{1}{n_2}\sum_{j=1}^{n_2}\delta(t_j)$ become probability measures. The pth Wasserstein metric between two probability measures μ, ν on T with a finite pth moment is defined as

$$W_p(\mu, \nu) := \left(\inf_{\gamma\in\Gamma(\mu,\nu)}\int_{T\times T} m(t, t')^p \,\mathrm{d}\gamma(t, t')\right)^{1/p}, \tag{5}$$

where $\Gamma(\mu, \nu)$ is the set of all couplings of μ and ν, i.e., measures on $T \times T$ with marginals μ and ν on the first and second factors, respectively. The Wasserstein metric can be recognized as a generalization of the optimal transport cost. See, for example, Villani (2008) and Peyré and Cuturi (2019) for more details on the theory, computation, and application of the Wasserstein metric.

The motivations of using the Wasserstein metric for our tree metric space are that (1) it can use the metric space's information and (2) the Wasserstein metric is itself a metric. Here, we explain these points in comparison with other measures of the difference between sets of trees.

If the problem is formulated as the distance between probability measures, the Kullback-Leibler divergence or the total variation distance might be candidate metrics, for example. However, both of those use almost no information in the metric space. For example, if there is a tree t such that $t \in T_1$ but $t \notin T_2$, then neither measure's value changes no matter how far t is from the elements of T_2.

Alternatively, we could consider the distance between T_1 and T_2 as a dissimilarity of clusters. One

of the most popular criteria for clustering analysis is the group average:

$$\frac{1}{|T_1||T_2|} \sum_{t \in T_1, t' \in T_2} m(t, t').$$

However, this is not a distance because it fails to be 0 even when $T_1 = T_2$; thus, it may overestimate the distance between clusters with large variances.

C Examples of Questions from SQuad and Answers by ChatGPT

This section gives two examples, short and long, of answers by ChatGPT to two SQuad questions.
Question: During what years were the bones moved from Cimitière des Innocents to the catacombs?

> The bones from Cimetière des Innocents were moved to the catacombs in Paris, France from 1785 to 1786.

Question: In political federalism what does Presbyterian ecclesiology do?

> Presbyterian ecclesiology refers to the organizational structure and governance of Presbyterian churches. In the context of political federalism, Presbyterian ecclesiology focuses on the principle of decentralization and the autonomy of local church communities within a broader Presbyterian denomination.
>
> Presbyterian churches are governed by a system of representative democracy, where decisions are made by representatives elected by the congregation. This system is known as the Presbyterian polity. The key feature of Presbyterian ecclesiology is the hierarchical connection between local churches, presbyteries, and general assemblies.
>
> In political federalism, Presbyterian ecclesiology can demonstrate the importance of local autonomy and regional decision-making. It emphasizes the idea that power is shared and distributed among various levels of authority, mirroring the concept of federalism in a political system. This decentralized structure allows for diverse opinions and practices within the broader Presbyterian denomination while ensuring a level of cohesion and connection among member churches.
>
> Overall, Presbyterian ecclesiology supports the idea of federalism by promoting the local autonomy of churches within a larger denomination and facilitating decision-making through representative bodies at different hierarchical levels.

D Effect of Parsing

Figure 9 shows UMAP visualizations of English-EWT and the original English-EWT sentences as parsed by two parsing software applications: UDPipe in blue (Straka, 2018; Straka and Straková, 2020)[6] and Spacy in green[7]. Both UDPipe and Spacy are excellent state-of-the art applications.

Even though the three sets are based on the same set of sentences, the scattering of trees is not the same in via UMAP. Spacy appears especially far from English-EWT. As seen in Table 1, the Wasserstein distances for both the unsigned and signed TED were by far larger, and our proposed statistical test rejected the null hypothesis. In contrast, UDPipe accepted the null hypothesis, giving the largest p-value of any set, which is expected because the original sentences were the same.

[6] We used the dictionary english-partut-ud-2.6-200830.

[7] We downloaded it from www.spacy.io, with the dictionary en_core_web_sm.

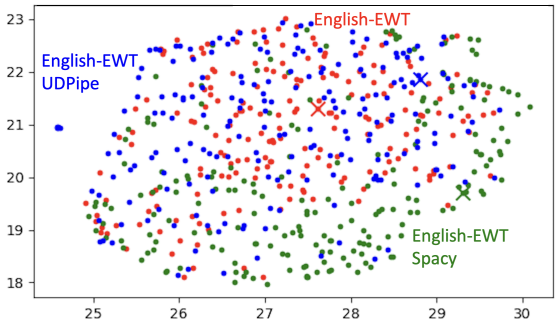

Figure 9: UMAP visualization of English-EWT (red), in comparison with two reparsed results obtained with UDPipe (blue) and Spacy (green).

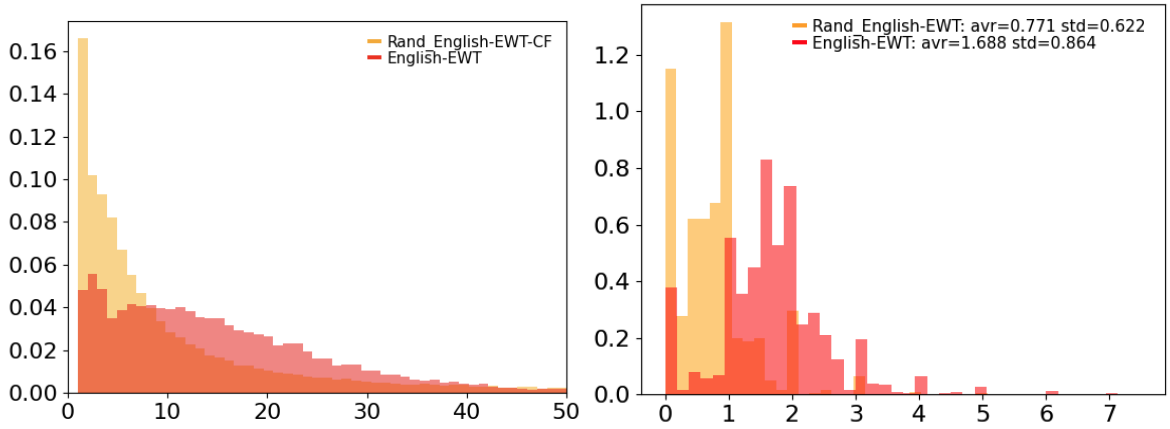

Figure 10: Distributions of sentence lengths (left) and bifurcation ratios (right) for English-EWT (red) and Rand-English-EWT-CF (orange).

We believe that this appendix provides further evidence of our method's utility by evaluating the parsing quality through the tree structure's topology, which is different from evaluating a parser only by the precision.

Given this result, we used UDPipe to parse the ChatGPT text and examine whether it was different from English-EWT.

E English Sentence Characteristics Through Simple Statistics

E.1 Distribution of Sentence Lengths

The left graph in Figure 10 shows the distributions of sentence lengths for the English-EWT and Rand-English-EWT-CF datasets in red and orange, respectively.

For Rand, the length distribution drops monotonically. This is a natural consequence, because short sentences are more probable to occur among random context-free sentences. In contrast, for EWT, there is a split at length 5 such that for longer sentences, the lengths have a distribution within a certain range. Since Yule, studies on sentence length distributions have modeled them by log-normal or gamma functions (Yule, 1939, 1968; Sichel, 1974), and the distribution for EWT does seem to have a similar shape for $n > 5$.

In addition, we can see a peak at lengths around 10. In contrast, the distribution's peak for other corpora is often not at $N = 10$, as seen in the third column in the second block of Table 1: eight corpora

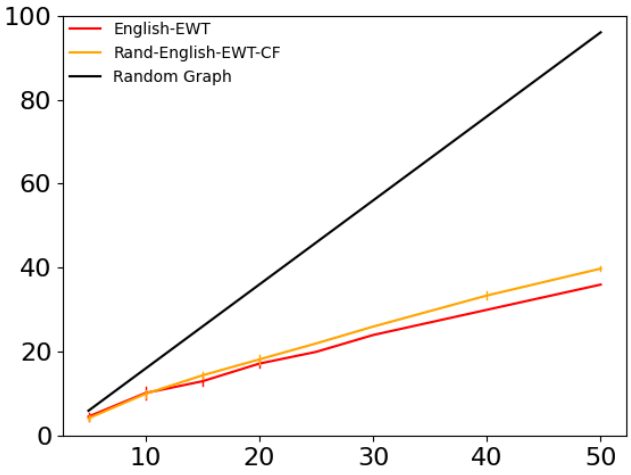

Figure 11: Diameter with respect to N (horizontal axis) for English-EWT, Rand-English-EWT-CF, and a random graph.

have more samples at $N = 20$ than at $N = 10$. This already illustrates some of the varied nature of treebanks.

E.2 Diameter

The *diameter* of a set T is defined as follows:

$$\text{diameter}(T) \equiv \max_{t_i, t_j \in T} m(t_i, t_j). \tag{6}$$

The diameter of an unlabeled ordered random tree of size n is $2(n-2)$. This is because $n-2$ children, i.e., all but one child, must be edited by insertion and deletion. There is a question here of how much smaller the diameters of natural language sentences are as compared to the diameters of a set of general trees.

Figure 11 shows the diameter's growth with respect to N (horizontal axis) as calculated with $K = 200$ for English-EWT (red) and Rand-English-EWT-CF (orange). For $N > 20$, when the sample size was less than $K = 200$, all of the samples were used for this figure; for $N \leq 20$, however, $K = 10$ different samples were acquired to calculate the diameter's standard deviation, as shown by the vertical bars at each point. In addition, the black line represents the diameter among all unlabeled ordered dependency trees of size N (i.e., $2(N-2)$; see Section 6). For comparison with this theoretical result, Rand-English-EWT-CF and English-EWT were also processed as unlabeled ordered trees.

The figure shows that the space's size increases almost linearly with N for both datasets, though the slopes are dramatically different. However, the growth for English-EWT and Rand-English-EWT-CF seems to slow down slightly as N increases. At $N = 50$, the diameter is 37.5% for English-EWT with all unlabeled ordered trees, whereas it is 41.7% for Rand-English-EWT-CF. Obviously, human UD structures cover only a fraction of the whole tree space. The difference between English-EWT and Rand-English-EWT-CF indicates how real sentences differ from context-free trees only by structure. The following section considers this difference in more detail.

The vertical ticks at the points with smaller N show the diameter's standard deviation among 10 samples. For larger N, the deviation is small or even zero because of the limited number of samples. This does not signify, however, that two sampled sets of size K do not have any difference. The first line

in Table 1 indicates the average Wasserstein distance among 10 pairs for English-EWT. The mean is 3.00 for unlabeled ordered trees, but this distance has a very small standard deviation of 0.085. Furthermore, a pair of sets of point clouds for English-EWT obtained by MDS embedding would not show any visible difference at all.

E.3 Distribution of Bifurcation Ratios

Lastly, we report a measure related to the diameter, the degree of branching of each tree. Let $\mathrm{node}(t)$ be the set of all nodes of $t \in T$, and for $v \in \mathrm{node}(t)$, let $\mathrm{ch}(v)$ be the number of v's children. The *number of bifurcations*, $\mathrm{nb}(t)$, of t is then defined as follows:

$$\mathrm{nb}(t) \equiv \sum_{v \in \mathrm{node}(t), \mathrm{ch}(v) > 0} (\mathrm{ch}(v) - 1). \tag{7}$$

In other words, $\mathrm{nb}(t)$ equals the total number of edges leaving all the inner nodes minus the number of inner nodes. Note that for unlabeled ordered trees, $\mathrm{nb}(t)$ is exactly half of the TED between t and a linear tree of the same size, i.e., a tree whose nodes all have one child, except the leaf. This is because one branching entails a distance of 2, i.e., a deletion and an insertion, so that the tree becomes linear.

As this $\mathrm{nb}(t)$ obviously increases with the tree size n, $\mathrm{nb}(t)$ should be considered in proportion to the number of inner nodes; accordingly, the bifurcation ratio $\mathrm{br}(t)$ is defined as follows:

$$\mathrm{br}(t) \equiv \frac{\mathrm{nb}(t)}{\#(v \in \mathrm{node}(t), \mathrm{ch}(v) > 0)}. \tag{8}$$

The bifurcation ratio is 0 for a linear tree. Among complete trees, the ratio is 1 for a complete binary tree, 2 for a ternary tree, and so on.

Among the five inner nodes in the example of Figure 1, two have more than one child (the root "join" and "director"). The total number of edges leaving all the inner nodes minus the number of inner nodes is 4, so $\mathrm{nb}(t) = 4$, and $\mathrm{br}(t) = 4/5 = 0.8$.

The right graph in Figure 10 shows the bifurcation ratios calculated from Formula (8) for English-EWT and Rand-English-EWT-CF. For Rand, the average bifurcation ratio is 0.77, even though it was generated by sampling child nodes uniformly following English-EWT. On the other hand, for EWT, the average bifurcation ratio is 1.69, with a standard deviation of 0.86. This suggests that the average branching of English sentences is more than binary and less than ternary. Thus, from a statistical viewpoint, natural language sentences are more bifurcated than complete binary trees when averaging across all inner nodes.

Acknowledgment

This work was supported by JST, CREST Grant Number JPMJCR2114, Japan.

References

Bernard, M., Habrard, A., and Sebban, M. (2006). Learning stochastic tree edit distance. In *Machine Learning: ECML 2006*, pages 42–53. Springer Berlin Heidelberg.

Bille, P. (2005). A survey on tree edit distance and related problems. *Theor. Comput. Sci.*, **337**(1), 217–239.

Billera, L., Holmes, S., and Vogtmann, K. (2001). Geometry of the space of phylogenic trees. *Advances in Applied Mathematics*, **27**, 733–767.

Boyer, L., Habrard, A., and Sebban, M. (2007). Learning metrics between tree structured data: Application to image recognition. In *Machine Learning: ECML 2007*, pages 54–66. Springer Berlin Heidelberg.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., *et al.* (2020). Language models are few-shot learners. *Advances in neural information processing systems*, **33**, 1877–1901.

Buneman, P. (1971). The recovery of trees from measures of dissimilarity. *Mathematics in the archaeological and historical sciences.*

Buneman, P. (1974). A note on the metric properties of trees. *Journal of Combinatorail Theory, Series B*, **17**(1), 48–50.

Chawathe, S. (1999). Comparing hierarchical data in external memory. *VLDB J.*

Dryer, M. S. (1992). The Greenbergian word order correlations. *Language*, **68**(1), 81–138.

Dümbgen, L. (1993). On nondifferentiable functions and the bootstrap. *Probab. Theory Related Fields*, **95**(1), 125–140.

Feragen, A., Lo, P., Bruijne, M. d., Nielsen, M., and Lauze, F. (2013). Toward a theory of statistical tree-shape analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**(8).

Good, P. (2013). *Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypotheses*. Springer Science & Business Media.

Greenberg, J. H. (1963). *Universals of Language*. The MIT Press. In the chapter entitled "Some universals of grammar with particular reference to the order of meaningful elements", pages 73–113.

Haspelmath, M., Dryer, M. S., Gil, D., and Comrie, B. (2005). *The World Atlas of Language Structures*. Oxford University Press.

Klein, D. and Manning, C. (2004). Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 478–485.

Krishna, T. and Yusu, W. (2022). *Computational Topology for Data Analysis*. Cambridge University Press.

Liu, H. (2010). Dependency direction as a means of word-order typology: A method based on dependency treebanks. *Lingua*, **120**(6), 1567–1578.

Liu, H., Xu, C., and Liang, J. (2017). Dependency distsance: a new perspective on syntactic patterns in natural languages. *Physics of Life Reviews*, **21**, 171–193.

Liu, J., Shuangge, M., Xu, W., and Zhu, L. (2022). A generalized wilcoxon-mann-whiteney type test for multivariate data through pairwise distance. *Journal of Multivariate Analysis*, **190**(104946).

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, **19**(2), 313–330.

Mclnnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *ArXiv e-prints*, (1802.03426).

Micheli, A. and Rossin, D. (2005). Edit distance between unlabeled ordered trees. *RAIRO - Theoretical Informatics and Applications*, **40**(4).

Mielke, P. W. and Berry, K. J. (2007). *Permutation Methods*. Springer New York.

Nivre, J., de Marneffe, M.-C., Ginter, F., Hajič, J., Manning, C. D., Pyysalo, S., Schuster, S., Tyers, F., and Zeman, D. (2020). Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Panaretos, V. M. and Zemel, Y. (2019). Statistical aspects of wasserstein distances. *Annu. Rev. Stat. Appl.*, **6**(1), 405–431.

Parikh, A. P., Cohen, S. B., and Xing, E. P. (2014). Spectral unsupervised parsing with additive tree metrics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1062–1072, Baltimore, Maryland. Association for Computational Linguistics.

Pesarin, F. (2010). *Permutation Tests for Complex Data: Theory, Applications and Software*. Wiley & Sons, Limited, John.

Peyré, G. and Cuturi, M. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, **11**(5-6), 355–607.

Pullum, K. G. and Gazdar, G. (1982). Natural languages and context-free languages. *Linguistics and Philosophy*, **4**(4), 471–504.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Selkow, S. M. (1977). The tree-to-tree editing problem. *Inf. Process. Lett.*, **6**(6), 184–186.

Shindo, H., Nishino, M., Kobayashi, Y., and Akihiro, Y. (2020). Metric learning for ordered labeled trees with pq-grams. In *Proceedings of European Conference of Artifial Intelligence*.

Sichel, H. S. (1974). On a distribution representing sentence-length in written prose. *Journal of the Royal Statistical Society: Series A*, **137**(1), 25–34.

Sommerfeld, M. and Munk, A. (2017). Inference for empirical wasserstein distances on finite spaces. *J. R. Stat. Soc. Series B Stat. Methodol.*, **80**(1), 219–238.

Straka, M. (2018). UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.

Straka, M. and Straková, J. (2020). Udpipe at evalatin 202: Contextualized embeddings and treebank embeddings. *ArXiv.org*.

van der Vaart, A. and Wellner, J. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer Science & Business Media.

Villani, C. (2008). *Optimal Transport: Old and New*. Springer Science & Business Media.

Wang, J., Gao, R., and Xie, Y. (2022). Two-Sample test with kernel projected wasserstein distance. In G. Camps-Valls, F. J. R. Ruiz, and I. Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8022–8055. PMLR.

Weed, J. and Bach, F. (2019). Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. *BJOG*, **25**(4A), 2620–2648.

Xing, E. P., Y. Ng, A., Jourdan, M. I., and Russell, S. (2002). Distance metric learning, with application to clustering with side-information. In *Proceedings of Neural Information Processing Systems*.

Yule, U. G. (1939). On sentence length as a statistical characteristic of style in prose: With application to two cases of disputed authorship. *Biometrika,*, **30**, 363–390.

Yule, U. G. (1968). *The Statistical Study of Literary Vocabulary*. Archon Books.

Zhang, K. and Shasha, D. (1989). Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal of Computing*, **18**, 1245–1262.