# Measuring Annotator Agreement Generally across Complex Structured, Multi-object, and Free-text Annotation Tasks

Alexander Braylan
Dept. of Computer Science
University of Texas at Austin, USA
braylan@cs.utexas.edu

Omar Alonso
College of Computer Sciences
Northeastern University, USA
o.alonso@northeastern.edu

Matthew Lease
School of Information
University of Texas at Austin, USA
ml@utexas.edu

## ABSTRACT

When annotators label data, a key metric for quality assurance is inter-annotator agreement (IAA): the extent to which annotators agree on their labels. Though many IAA measures exist for simple categorical and ordinal labeling tasks, relatively little work has considered more complex labeling tasks, such as structured, multi-object, and free-text annotations. Krippendorff's $\alpha$, best known for use with simpler labeling tasks, does have a distance-based formulation with broader applicability, but little work has studied its efficacy and consistency across complex annotation tasks.

We investigate the design and evaluation of IAA measures for complex annotation tasks, with evaluation spanning seven diverse tasks: image bounding boxes, image keypoints, text sequence tagging, ranked lists, free text translations, numeric vectors, and syntax trees. We identify the difficulty of interpretability and the complexity of choosing a distance function as key obstacles in applying Krippendorff's $\alpha$ generally across these tasks. We propose two novel, more interpretable measures, showing they yield more consistent IAA measures across tasks and annotation distance functions.

## CCS CONCEPTS

• **Information systems** → *Crowdsourcing*; *Trust*; • **Computing methodologies** → Unsupervised learning; Learning in probabilistic graphical models.

## KEYWORDS

annotation, labeling, inter-annotator agreement, quality assurance

**ACM Reference Format:**
Alexander Braylan, Omar Alonso, and Matthew Lease. 2022. Measuring Annotator Agreement Generally across Complex Structured, Multi-object, and Free-text Annotation Tasks. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3485447.3512242

## 1 INTRODUCTION

Data annotations are often collected from human experts or crowdsourcing [2] as part of the process for training and evaluating models. As an early and crucial node of the machine learning pipeline, it is important both to have quality labels [5, 31] and to be able to measure label quality. In real-world applications that require gathering many labels, measures of *inter-annotator agreement* (IAA) [34] can be used to detect problems with data reliability stemming from task design, workers' performance, or other causes [1]. Due to the wide variety of different annotation tasks, there is typically no single method for measuring agreement that is suitable for every purpose, and sometimes the inappropriate use and interpretation of such statistics can lead to wasted effort and resources or misspecified and biased models. A more comprehensive approach for this problem is to understand the complexity of the labeling task, identify an agreement metric that is explainable and fits the specific project requirements, in combination with other quality control mechanisms that can quantify the quality of a dataset.

In this paper, we investigate the use of IAA measures for "complex" annotation tasks [10, 11] having large (finite or continuous) answer spaces, such as bounding boxes and keypoints in images, named entities in text, syntactic parse trees, free-text translations, ranked lists, and multi-dimensional numeric vectors. Most prior IAA studies assume relatively simple labeling tasks, such as classification or ordinal rating tasks.

One of the most versatile IAA measures, Krippendorff's $\alpha$ [18], can (in its most general form) be applied across diverse labeling tasks. As a baseline, we present empirical results for $\alpha$ across a variety of complex annotation tasks and task-specific distance functions for measuring annotation similarity. However, we observe two important limitations of Krippendorff's $\alpha$. First, $\alpha$ is difficult to interpret because its threshold for acceptable agreement varies greatly by task and distance function. This makes it confusing to understand when collected labels are of sufficient quality for use, especially with new tasks in which the task-specific $\alpha$ threshold is not yet known. Second, $\alpha$ requires selection of an appropriate distance function for the annotation task, and a poor choice can add noise, obscuring and underestimating agreement. This choice of distance function can be complicated as well. While prior work has relied on building simulators such as the Corpus Shuffle Tool (CST) [22] to evaluate distance functions, this requires creating an annotation simulator for each new labeling task of interest.

Our innovation is to propose new, distributional variants of Krippendorff's $\alpha$ that provide a conceptually and empirically more interpretable threshold for deciding that the data is "good enough", as well as clearer insight into selecting a distance function, without requiring either task-specific label noise simulation or gold data.

**Contributions** of our work include:

- We provide a guide to the considerations and techniques for specifying an annotation distance function, that generalizes across various complex annotation tasks.


*WWW '22, April 25–29, 2022, Virtual Event, Lyon, France*

ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3512242

- We identify two key limitations of the general form of Krippendorff's $\alpha$ and provide novel alternatives.
- A new IAA measure based on the Kolmogorov-Smirnov test [21] is shown to be particularly effective in evaluating distance functions for use with IAA, precluding need for either label noise simulation or gold data.
- A new IAA measure $\sigma$ provides a clear and task-general interpretation, with a lower bound to what fraction of observed label distances are significantly smaller than chance.
- To support reproducibility, we share our code and data[1].

## 2 RELATED WORK

### 2.1 Inter-annotator Agreement

In collecting labeled data, it is useful to distinguish between *objective* tasks (in which a single best response is presumed to exist for each item) vs. *subjective* labeling tasks [29] that expect diverse responses (e.g., soliciting personal preferences or opinions). Whereas high inter-annotator agreement (IAA) [34] is typically a goal with objective tasks, that is not the case with subjective tasks.

Even with objective tasks, annotator disagreement can still be a useful signal to model training and evaluation [6], indicating varying confidence of "ground truth" labels for across items (e.g., due to corner-cases in annotation guidelines or difficult instances such as blurry images, etc.). Annotators may also cluster into different *schools of thought* [44] in interpreting or executing annotation guidelines due to different personal backgrounds, task ambiguity, etc. In addition, there are further risks of data bias [26, 41]: a pool of homogeneous annotators may agree with one another yet miss important problems with task guidelines or specific items that may be apparent to more diverse and representative annotators.

The purpose of collecting multiple annotations per item is typically to assess and/or improve the quality of the labels, where the quality is presumed better when annotators agree (given the above caveats). Annotator agreement should not be confused with correctness; annotators can agree with one another yet share a systematic bias in collectively interpreting task guidelines differently than the author of those guidelines had intended. A common practice is thus to first check if annotators agree with one another (i.e., is the task clear?), then sample agreed-upon labels to ensure they are further consistent with what was actually desired from data collection.

Given the inherent variability in human judgement as well as complexity of the collection process, disagreements can arise for a large variety of reasons, such as annotator heterogeneity (which is often desirable). Beyond demographics, annotators may vary in their training or skill, or in the effort they apply. Their labels may change over time from fatigue [12] or calibration [38].

Inter-annotator disagreement may also arise from heterogeneous items. Some items may be more difficult to annotate than other items [45]. Even the definition of difficulty itself can be divided into multiple types. For example, items might be more *discriminating*, in that the more skilled annotators are much less prone to error than the less skilled ones, or more *ambiguous* in that both skilled and less skilled annotators are equally prone to error [8].

**Global sources of inter-annotator disagreement** include a random noise factor that may affect any given annotation, as well as systematic problems in the annotation process such as an unreliable platform or confusing instructions. These can stem from many ways in which annotations are collected, whether it be a crowdsourcing platform, an internal lab or team, managed workers, etc, with wide varieties in how tasks are designed and implemented.

Finally, one global source of measured inter-annotator disagreement is the method by which it is being measured. Much of the prior work on measuring IAA is around improving these measures so that they do not show more or less agreement than what arises from the aforementioned factors. One of the major innovations from prior work is the *chance correction*, or the separation of observed disagreements between annotations from what should be expected due to chance. Many such chance-corrected agreement measures exist, including Scott's $\pi$ [39], Cohen's $\kappa$ [15], and Fleiss' $\kappa$ [16]. The common approach for chance correction is to distinguish *observed disagreements* $D_o$ from *expected disagreements* $D_e$. Not performing such a correction can hinder the interpretability of the agreement measure, as the size of the possible and likely response spaces would heavily affect the magnitude of the measure.

Krippendorff's $\alpha$ [18] is a measure that aims to generalize many others. Not only can it handle any number of annotators and missing values, it also allows plugging in a *distance function* that could in principle apply to any type of annotation for which such a function can be conceived. However, because of its design in the context of certain specific tasks such as content analysis, there is sometimes confusion around its definition. When the literature refers to Krippendorff's $\alpha$, it may refer to either: i) the general formula $\alpha = 1 - \frac{\hat{D}_o}{\hat{D}_e}$ for computing agreement given a distance function $D(a, b)$; or ii) a specific distance function $D(a, b)$ to use in this general formula.

Krippendorff [18] prescribes distance functions for several kinds of data, including nominal, ordinal, interval, ratio, polar, and circular. Sometimes alternatives to Krippendorff's $\alpha$ are actually alternatives to the prescribed distance functions rather than the general form. For example "weighted Krippendorff's $\alpha$" [3] distinguishes the use of a Euclidean distance function from a binary distance function, but relies on the same general form $1 - \frac{\hat{D}_o}{\hat{D}_e}$. In this paper, when we refer to Krippendorff's $\alpha$ we mean specifically the general formula and not any de facto prescriptions of distance functions. It is important to make this distinction in order to separate properties and criticisms of the general formula from properties and criticisms of specific distance functions that plug into it. This paper investigates challenges with both the general formula and with the variety and assessment of distance functions for complex annotations.

### 2.2 Complex Annotations

As practitioners seek to automate ever-more sophisticated tasks, annotation needed to train and evaluate models becomes increasingly complex. More complex tasks [4, 10, 11, 33] may involve open-ended answer spaces (e.g., translation, transcription, extraction) or structured responses (e.g., annotating ranked lists, linguistic syntax or co-reference), such as those shown in Figure 1.

In addition, while simple labeling tasks like classification may require only a single label for each input *item* (e.g., a document or image), many complex annotation tasks require annotators to label an unknown number of *objects* per item — such as demarcating named-entities in a sentence [37] or visual entities in an image [9],

[1] https://github.com/Praznat/annotationmodeling

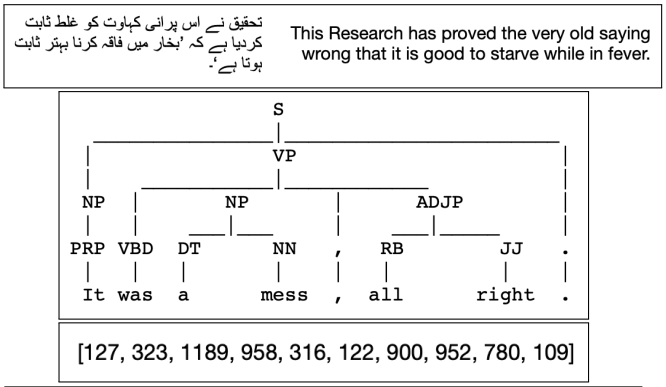



**Figure 1: Examples of complex annotations. Open-ended: language translation. Structured: a syntactic parse tree, and a ranked list of elements. Multi-object: text sequences for information extraction, image bounding boxes for object detection, and image keypoints for pose estimation.**

as shown in the last three examples of Figure 1. Mathet et al. [23] refer to this part of a task as *unitizing* – defining the boundaries objects of interest – in contrast to *categorizing*, or assigning a class to each object. Braylan and Lease [10] define *complex annotations* as anything that is not a categorical or simple numerical label – basically anything with a large to infinite response space. These tasks may also require greater cognitive effort by annotators.

Recent work has explored modeling distances between annotations as a general approach to *label aggregation* across diverse annotation types [10, 11, 19, 27]. A commonality among these is the use of a task-specific *distance function* to convert each specific complex domain into a much more general numeric one. This is the same trick Krippendorff's $\alpha$ takes advantage of to handle complex annotations, but this prior work recommends the use of common *evaluation functions* – measures of a predicted label against gold – which when inverted can be used as distance functions. An open question in this prior work is how to judge what distance function works best when there are multiple options.

## 2.3 Agreement for Complex Annotations

While theoretically Krippendorff's $\alpha$ is applicable to any complex annotation task that has an available distance function, the prior work investigating such applications is sparse. Skjærholt [42] investigates agreement metrics for dependency syntax trees, based on Krippendorff's $\alpha$. They evaluate different distance functions on syntax trees for use with Krippendorff's $\alpha$. Mathet et al. [23] propose a *Holistic* $\gamma$ measure of IAA for various linguistic annotation tasks such as Named Entity Recognition and Discourse Framing.

Both of the above works depend on the use of a *Corpus Shuffling Tool (CST)* [22] for evaluating IAA measures. The insight of CST is to use a simulator of noise applied to complex annotations to control the expected amount of disagreement. A proposed measure of IAA is evaluated by plotting the amount of simulated error in a dataset against the measured agreement. An ideal measure should span from 0 to 1 as simulated error spans from 1 to 0, and this response should be strictly decreasing. Many of the conclusions from this prior work come from using CST to judge and rank various candidate distance functions to plug into Krippendorff's $\alpha$.

The downside of CST is that it is taxing to build simulators of possible error. Furthermore, these simulators might deviate from reality in the types of errors they capture. These reasons make it difficult to apply CST to a wide range of different complex annotation tasks, and therefore make it difficult to choose an appropriate distance function for a given task.

Choosing an effective distance function is crucial for measuring IAA. As discussed in Section 2.1, a poor distance function can be a global contributor to total measured disagreement. That means the distance function competes as an explanation of disagreement with the other potential sources such as task difficulty, ambiguity, etc, that practitioners care about. The only way to isolate the effect of the distance function is to choose one that works better.

One desirable property of an IAA measure is thus its **absolute level to distinguish good distance functions from bad ones**. In Section 4 we argue that Krippendorff's $\alpha$ does not fulfill this need, which is why prior work has relied on CST experiments.

# 3 GENERALIZING TO COMPLEX TASKS

In this section we lay out the assumptions, requirements, and procedure for measuring IAA in a way that generalizes across many diverse complex annotation modalities.

The general formula for Krippendorff's $\alpha = 1 - \frac{\hat{D}_o}{\hat{D}_e}$, where $\hat{D}_o$ is the *average distance observed*, and $\hat{D}_e$ is the *average distance expected*. Observed distances $D_o$ are the set of *within-item* pairwise distances between annotations, given a distance function $D$.

$$D_o = \{D(a,b) \mid \text{ITEM}(a) = \text{ITEM}(b)\} \quad , \quad \hat{D}_o = \frac{1}{|D_o|} \sum_{d \in D_o} d$$

The standard method for getting the set of expected distances $D_e$ is to sample *inter-item* pairwise distances between annotations, which is generally applicable to complex annotations as well.

$$D_e = \{D(a,b) \mid \text{ITEM}(a) \neq \text{ITEM}(b)\} \quad , \quad \hat{D}_e = \frac{1}{|D_e|} \sum_{d \in D_e} d$$

Recall that expected distances are used for chance correction. That means that expected distances should represent what a measured distance might be between randomly-made annotations. For example, taking translations from two different items (source sentences) is a reasonable proxy for randomly generating translations.

### 3.1 Distance Function Properties

This method for computing $D_e$ introduces one limitation: **distance functions must be applicable to pairs of annotations from different items**. As noted by Skjærholt [42], such a limitation precludes a number of candidate distance functions for syntactic parse trees, including EVALB [40], causing them to choose Tree Edit Distance (TED) which does not have this limitation.

Also noted in Mathet et al. [23] is that distance functions are *metrics* that must fulfill the requirements of **Non-negativity**, **Symmetry**, **Zero only for identical inputs**, and **Triangle inequality**. One may skip the triangle inequality requirement and call it a *dissimilarity*, as in Mathet et al. [23]. However, we will continue to use the term "distance" in this paper, assuming the triangle inequality requirement without studying whether it is truly needed.

### 3.2 Supporting Multi-object Items

With multi-object labeling tasks (e.g., labeling named-entities in a text or bounding boxes in an image), annotators or prediction models must locate (aka *unitize* [23]) and categorize 1-many objects in each input text/image. Evaluation metrics then score how well the set of annotated or predicted objects matches the true set of gold objects for the text/image. As Braylan and Lease [11] note, evaluation metrics already exist for many such multi-object labeling tasks and often can be directly applied as distance functions.

Another general strategy for supporting multi-object labeling tasks is to induce a multi-object distance function $D_m(A, B)$ from a single-object distance function $D_s(a, b)$ by finding the minimum distance from each object in $A$ to each object in $B$ and vice-versa.

$$\Delta(A, B) = \mathbb{E}\{\min(\{D_s(a, b) \mid b \in B\}) \mid a \in A\}$$

$$D_m(A, B) = \frac{\Delta(A, B) + \Delta(B, A)}{2}$$

For example, Mathet et al. [23] provide a comparable algorithm for computing an *alignment* that minimizes local disagreements between units in different annotations. The benefit of this kind of approach is that it abstracts away everything except the choice of single-object distance function $D_s$, which can be easier to provide.

Yet another general approach for dealing with complex annotations is to first decompose them into simpler annotations and operate on these instead [11, 33]. For example, Nguyen et al. [30] relax traditionally "strict" NER scoring of exact spans [37] by decomposing them into tokens and computing "partial-credit" scoring by token. Similarly, Jaccard index or Intersection-over-Union (IoU) decompose labeled image regions into pixels and measure partial-credit label distance based on area overlap. The benefit of decomposition is that since every annotator (implicitly) labels every low-level token/pixel, there is no need to align labels across annotators.

Krippendorff's $\alpha$ plus an appropriate task-specific distance function provides a very general approach for measuring IAA across diverse types of complex annotations. However, questions remain as to how to choose between distance functions and how to interpret the IAA measure. In the following section we will discuss why Krippendorff's $\alpha$ presents difficulties towards these ends, and we propose methods to address those difficulties.

## 4 ALTERNATIVES TO KRIPPENDORFF'S $\alpha$

So far we have assumed IAA should be measured for complex annotations using Krippendorff's $\alpha$, with the main question being what distance function to use. In this section, however, we argue that Krippendorff's $\alpha$ falls short when applied generally to complex annotations, for two reasons: its difficulty of interpretation and the complexity of using it to choose between distance functions. We provide two novel IAA measures to help address this.

### 4.1 Problem of Interpretability

IAA scores should aid interpetation of collected data. While a single agreement number does not provide all possible useful information about all possible sources of disagreement (such as annotator subjectivity, item ambiguity, etc.), this top level number should describe how much the overall data is **better than chance**. By looking at some examples we see that Krippendorff's $\alpha$ does not quite communicate that information sufficiently for complex data.

For example, noting the contrast in Figure 2 between observed distances $D_o$ on the diagonal blocks and expected distances $D_e$

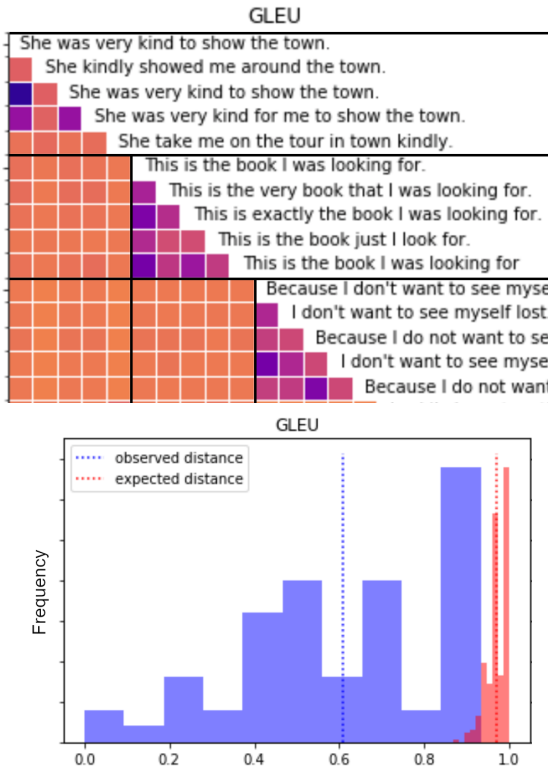



**Figure 2: Top: Heatmap of GLEU distance between translations for three sentences (darker being more similar). High contrast between within-item distances $D_o$ and inter-item distances $D_e$ implies annotators are much more in agreement than random. Bottom: Distribution of within-item GLEU distances $D_o$ and inter-item GLEU distances $D_e$ for same dataset. While the distribution for observed GLEU distances is quite wide, only a small portion of it overlaps the distribution of GLEU distances expected at random.**

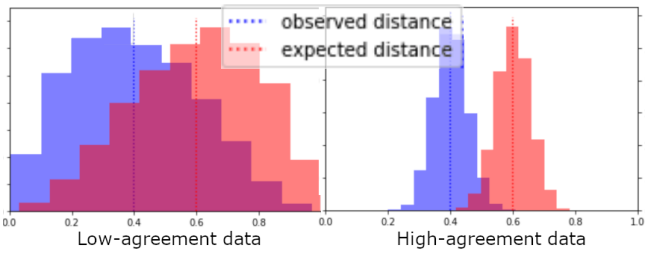



**Figure 3: Hypothetical datasets with low agreement (left) and high agreement (right). Purple-shaded regions denote area of overlap between observed annotation distances and distance expected from chance. Krippendorff's $\alpha$ is 0.33 for both datasets, whereas our measure $\sigma$ distinguishes 0.26 for the left and 0.98 for the right.**

outside those, it is surprising that the calculated Krippendorff's $\alpha$ is only 0.35. This number seems low given the obvious contrast between within-item and inter-item distances and the overall qualitatively acceptable level of agreement between translations. Krippendorff [18] stipulated 0.667 as the "lowest conceivable limit" of acceptable $\alpha$, although anticipating that such guidelines would not likely extrapolate as far as something like translation data. Still, one may ask what to do with such a low number in this example. Should we simply lower our expectations for acceptable $\alpha$ in Japanese-English translation tasks? What about other languages? What about other kinds of complex tasks? This difficulty is what we mean when we discuss the *interpretability* of an agreement measure. **To be *interpretable*, a measure must have a stable notion across new annotation tasks of when annotations are "good enough" to proceed from pilot to production.**

The reason for Krippendorff's $\alpha$'s low interpretability across tasks stems from the fact that it compares an *average* observed distance to an *average* expected distance. And depending on the task, there may be a whole *range* of acceptable distances that is lost when summarizing in the average. To clarify what we mean by this, consider the bottom of Figure 2, in which the $D_o$ (in blue) and $D_e$ (in red) values are rearranged into histograms. While the distribution for observed distances is quite wide, only a small portion of it overlaps the distribution of distances expected at random.

To more generally see the shortcoming of comparing average $D_o$ and $D_e$, consider two hypothetical datasets illustrated in Figure 3 which have the same average but different scales of $D_o$ and likewise for $D_e$. The dataset with the smaller scales intuitively has more agreement, as observed distances between annotations are more discernible from chance expected distances. However, both datasets yield the same value for Krippendorff's $\alpha$.

## 4.2 Problem of Distance Function Choice

Section 2.3 discussed the CST method used for judging and choosing between candidate distance functions for complex annotations. To reiterate, the CST method requires designing a simulator of errors specific to the complex annotation task at hand, which is impractical for the common practitioner who just wants a good way to measure agreement for their collected data. It would be ideal to have an easily computable single number that could grade a candidate distance function against another. Alas, Krippendorff's $\alpha$ cannot itself be used this way. As seen in the previous section and Figure 3, $\alpha$ is sometimes unable to differentiate a distance function that better separates $D_o$ and $D_e$ distributions from one that produces more overlap. Furthermore, we will see in Section 6.2 that Krippendorff's $\alpha$ sometimes scores much higher for distance functions that we expect to be much worse, relative to other distance functions.

## 4.3 Methods Proposed

In order to better generalize to a wider set of tasks, we propose calculating agreement based on the difference in *distribution* of $D_o$ and $D_e$ rather than the difference in their averages.

For comparing distributions there are a number of options. To grade options we consider two of the shortcomings of Krippendorff's $\alpha$ on complex annotations that we would like to remedy: the need for stable interpretability of the agreement score and the need for using the score to choose between distance functions. The former also depends on the latter – agreement should be measured using the best distance function available in order to minimize the effect of the distance function as a source of disagreement.

One way to compare distributions is to first perform kernel density estimation to get a smooth probability distribution function (PDF) for the $D_e$ that can be evaluated for any individual observation from $D_o$. Then we can simply ask, "*for each observed distance in $D_o$ what is the probability that a random draw from the distribution of expected disagreements $D_e$ could be smaller than it?*" For any given observation $d$ from $D_o$, we can say that it is statistically significantly different from a random distance, if the cumulative distribution function (CDF, i.e. the integral of the PDF) of $D_e$ up to $d$ is smaller than $p = 0.05$ (this $p$ threshold parameter is flexible but should be used consistently). Finally, we note the fraction of $D_o$ deemed to have passed this one-sided significance test. This measure is easy to calculate and interpret: **the fraction of the observed distances that are unlikely to be drawn from random expected distances**. We denote this measure as $\sigma$:

$$\sigma = \mathop{\mathbb{E}}_{d \in D_o} [\, p > \int^{d} \mathrm{PDF}(D_e)]$$

A second option for comparing empirical distributions is the one-sided Kolmogorov-Smirnov (KS) test [21], used to determine whether a sample ($D_o$) could be a drawn from a reference distribution ($D_e$). While $\sigma$ compares each observation from $D_o$ independently against the $D_e$ distribution, KS compares the whole sample.

$$\mathrm{KS} = \max_x(\mathrm{CDF}(D_o \leq x) - \mathrm{CDF}(D_e \leq x))$$

The measure of separation between distributions as a means to judge distance functions is supported by analogy to the *metric learning* literature[47], in which the objective is to learn a good metric: one that distinguishes data of the same class from data of different classes. Another analogy is clustering, where the objective is to have the ratio of within-cluster distances to inter-cluster distances be as small as possible [25]. A key takeaway from these analogies is that **a better distance function will better separate distributions of different classes**, without having to compare across different scenarios of simulated noise as CST does. We will also show through experiments in Section 6.2 how our approach ranks distance functions appropriately according to how established literature would rank distance functions.

We calculate IAA KS measure as the complement of the statistical significance p-value ($1 - p$) returned by the KS test. While we recommend the KS measure as a means to compare distance functions, we argue that the $\sigma$ measure has a more useful interpretation for deciding whether there is sufficient agreement in the data. Once a distance function has been chosen according to the highest KS score, we use the corresponding $\sigma$ for that distance function as a **lower bound** for how much of the data differs from chance. Just as "indistinguishable from random" does not necessarily mean random, the observed distances inside the expected distance distribution are not necessarily made in bad faith or due to errors or ambiguity. Therefore, a low $\sigma$ score does not necessarily mean the data should be discarded, it only means further investigation is necessary.

On the other hand, as a lower bound measure, a high $\sigma$ score is a good indication that there is not a significant amount of confusion or spamming or other causes for random-seeming annotations. A high $\sigma$ does not necessarily mean there is nothing left to investigate, but as a summary measure it serves the purpose of comparing agreements overall against chance. What this does not guarantee is whether these annotations are useful enough to deploy in the real world. For that, it is still important to consider the needs and nuances that vary from task to task. Overall we recommend interpreting IAA by using these proposed measures, but not exclusively as there can be other sources of data reliability issues that these measures do not specifically identify, such as bias.

## 5 EXPERIMENTAL SETUP

To compare inter-annotator agreement (IAA) metrics across a broad range of complex annotation types, we now summarize the datasets and distance functions used. For each dataset, we compare IAA using Krippendorff's $\alpha$ vs. our two new IAA metrics: KS and $\sigma$.

**Vectors.** Dataset. Snow et al. [43] ask workers to score short text headlines according to six emotions on a [0-100] interval. Such data is typically modeled as a set of independent ordinal rating questions, neglecting that the same headline is being rated for all six emotions. In this work, we treat each headline as a single item to which the annotator assigns a complex, vector of six scalar values. Distance functions. We compare coarse exact match (binary) vs. finer-grained Euclidean distance; Antoine et al. [3] recommend the latter for use of Krippendorff's $\alpha$ with ordinal annotations.

**Translations.** Dataset. Li and Fukumoto [20]'s CrowdWSA2019 dataset of crowdsourced Japanese-to-English translations is drawn mostly from Japanese native speakers and non-native speakers of English. They encourage beginner English speakers to participate and collect a dataset of more diverse quality than usually used to train machine translation models. Distance functions. The four distance functions we compare are (in order of expected quality): Levenshtein, BLEU, GLEU, and BERTScore. Levenshtein is a general edit distance measure that ignores the finer nuances of natural language. BLEU [32] is a traditional baseline for evaluating translations. GLEU [46] is a variant and improvement on BLEU, which is specialized for comparisons between individual sentences. BERTScore [48] is the most modern approach, which takes advantages of the nuances of meaning and grammar baked into BERT embeddings. Zhang et al. [48] find it to correlate better with human judgements than quite a large assortment of competing measures.

**Bounding Boxes.** Dataset. Braylan and Lease [11] share an image bounding box dataset in which each box is defined by an upper-left and lower-right vertex. An image may contain several visual entities to annotate, resulting in one to many bounding boxes per image to annotate. Distance functions. We compare four distance functions: "Count Diff" measures only the difference in bounding box count between two annotations, L2 norm, Intersection Over Union (IoU), and Generalized IoU (GIoU). Rezatofighi et al. [35] propose and show GIoU to be an improvement over standard IoU, which is itself an improvement over the L2 norm.

**Named Entity Recognition.** Dataset. Sang and De Meulder [37] share a NER dataset in which annotators highlight and categorize multiple spans of text within news articles. Distance functions. We compare five distance functions varying in leniency. The coarsest "Count Diff" measures only the difference in named-entity count between two annotations. Leniency in the *range* gives partial credit for range overlap, while leniency in the *tag* simply ignores it. The strictest distance function requires both span and tag to be correct, while relaxations allow leniency in either or both span or tag.

**Keypoints.** Dataset. Braylan and Lease [11] share a synthetic dataset for image annotation using keypoints, generated by simulating various types of annotator noise over a base dataset from COCO [14]. An image may contain multiple visual entities to be annotated with keypoints. Distance functions. We compare three distance functions: the coarsest "Count Diff" of objects annotated, a coarse measurement of the IoU of the smallest boxes containing the keypoints, and the most commonly used function for comparing keypoints: Object Keypoint Similarity (OKS) [36].

**Parse Trees.** Dataset. Braylan and Lease [10] share a dataset of simulated syntactic parse data using the Brown corpus [17] and the Charniak parser [24]. This dataset provides an alternate test for agreement measures on syntactic parse trees compared to Skjærholt [42]'s syntactic parse data. Its source of simulated noise comes from error in sub-optimal machine parses, rather than random relabeling or reattaching of nodes. Distance functions. Following Skjærholt [42], we compare three variants of Tree Edit Distance (TED): $\alpha_{\text{plain}}$, $\alpha_{\text{norm}}$, and $\alpha_{\text{diff}}$, the latter two being different ways to normalize TED by the compared tree sizes. Skjærholt [42] finds $\alpha_{\text{plain}}$ to be the best distance function based on a CST analysis.

**Ranked Lists.** Dataset. Braylan and Lease [10] share a dataset of simulated ranked lists of elements. Distance functions. We compare three distance functions: a coarse Kendall's $\tau$ over only the top-5 ranks, and $\tau$ and Spearman's $\rho$ calculated over the full ranking.

## 6 RESULTS

We evaluate our methods with consideration of the following two objectives. First, the interpretation of our measures of IAA should be useful and general across a wide variety of different complex annotation tasks, with minimal need for domain-specific nuance. Second, our methods should help determine better distance functions, not only for measuring agreement but potentially also for aggregation and evaluation against gold.

### 6.1 Score Interpretability

Recall that an IAA score for a dataset should describe how much better the observed distances are from chance. However, there are several methods for describing such difference from chance, and it

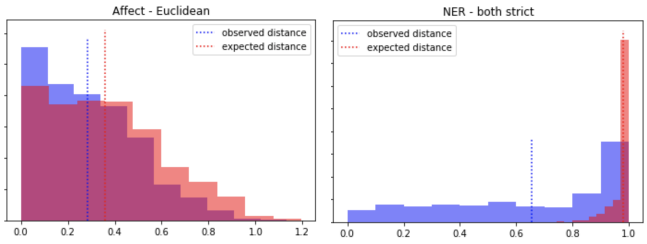
| Dataset | Distance f(x) | $\alpha$ | KS | $\sigma$ |
|---|---|---|---|---|
| Vector | Binary | 0.1277 | 0.5011 | 0.1151 |
| | Euclidean [3] | 0.2146 | 0.5885 | 0.1593 |
| Translations | Levenshtein | 0.2762 | 0.7735 | 0.5373 |
| | BLEU [32] | 0.1816 | 0.8532 | 0.5791 |
| | GLEU [46] | 0.1656 | 0.8758 | 0.8100 |
| | BERTScore [48] | 0.4534 | 0.9085 | 0.8952 |
| Bounding Boxes | Count Diff | 0.4365 | 0.6169 | 0.3736 |
| | L2 | 0.6873 | 0.9130 | 0.7640 |
| | IoU Score | 0.5046 | 0.9543 | 0.8418 |
| | GIoU Score [35] | 0.5069 | 0.9615 | 0.8711 |
| NER | Count Diff | 0.3900 | 0.6205 | 0.1969 |
| | both lenient | 0.4054 | 0.7816 | 0.6324 |
| | both strict [37] | 0.3324 | 0.7340 | 0.6620 |
| | strict range | 0.3776 | 0.7688 | 0.6520 |
| | strict tag | 0.3605 | 0.7848 | 0.6735 |
| Keypoints | Count Diff | 0.0419 | 0.3871 | 0.4007 |
| | IoU Score | 0.2924 | 0.7989 | 0.6278 |
| | OKS Score | 0.6726 | 0.8715 | 0.5666 |
| Parse Trees [42] | $\alpha_{\text{diff}}$ | 0.8422 | 0.9815 | 0.9181 |
| | $\alpha_{\text{plain}}$ | 0.8768 | 0.9909 | 0.9601 |
| | $\alpha_{\text{norm}}$ | 0.8626 | 0.9987 | 1.0000 |
| Ranked Lists | Kendall's $\tau$ @5 | 0.2005 | 0.6099 | 0.6158 |
| | Kendall's $\tau$ | 0.4915 | 0.9893 | 1.0000 |
| | Spearman's $\rho$ | 0.5413 | 0.9867 | 1.0000 |

**Table 1: IAA metrics for different distance functions across datasets. Best $\alpha$ varies greatly between datasets, sometimes reaching very low levels despite these being mostly reliable datasets. For distinguishing between distance functions, $\alpha$ is also unreliable, making questionable preferences such as L2 > GIoU and Levenshtein > GLEU. Our measure KS's ordering of distance functions is more in line with expectations.**

can be difficult to compare these methods when there is no ground truth for how useful each one is. One way to compare methods is to understand why they conflict with one another. Table 1 shows the different agreement scores for Krippendorff's $\alpha$, KS, and $\sigma$ across different distance functions for each dataset.

The top Krippendorff's $\alpha$ across distance functions varies drastically from dataset to dataset. This is expected because it is well known that $\alpha$ cannot be interpreted the same way across very different domains or across different distance functions [7]. In order to get a "good" $\alpha$ score, $D_o$ and $D_e$ should cluster heavily around 0 and 1, respectively. For complex annotations, such results are very rare, as the distributions of $D_o$ and $D_e$ can be quite wide.

On the other hand, KS seems to vary more from the choice in distance functions than from the task. The top KS score for each dataset is consistently high, with a single exceptionally low score for the Vector dataset. One explanation is that these are all fairly "good" datasets that were released publicly or generated with simulators. The entire sample of observed distances would need to be hard to differentiate from chance in order to produce a low KS score. Other than a serious problem in annotation task design, the only




**(a) $D_e$ concentrated on the low end for vector interval labels.** **(b) $D_o$ with a tall mode on the high end for NER exact-match.**

**Figure 4: Examples of IAA underestimated by $\sigma$ due to a smaller response space than ideal for using $\sigma$. This can be remedied for NER by using a more lenient distance function.**

other ways KS can receive a very low score are if 1) a sub-optimal distance function obscures the signal separating the observed from the expected distributions, or 2) the space of possible responses is small enough to make the expected and observed distributions overlap significantly. Case 1) we discuss in Section 6.2 and is the main contributor to variation in KS for complex annotations with large response spaces. Case 2) seems to contribute to the low KS score for the Vector dataset, as Figure 4a exemplifies how the observed and expected distances mostly overlap. Notably, most of the mass of the expected distance distribution falls on the low side, even more than would be expected for random independent draws from a six-dimensional uniform or Gaussian distribution, implying that the *likely* annotation space across all items is much smaller than the *possible* annotation space. This problem with interval data is pointed out by Checco et al. [13], whose alternative to Krippendorff's $\alpha$ is better suited to bounded numerical tasks like this one, though not applicable to more complex annotation types.

Our agreement measure $\sigma$ has the added value of interpretability over $\alpha$ and KS. A common struggle with $\alpha$ is that different distance functions result in very different values [7], and the relative order of $\alpha$ between distance functions we see is difficult to explain as well. While KS yields a favorable ordering of distance functions, $\sigma$ actually has a relatively simple natural language explanation.

In Section 4 we argue that the $\sigma$ score serves as a good lower bound measure of non-chance agreement. The reason it is a lower bound is that some of the real-life agreement can be occluded and underestimated due to the choice of distance function or nature of the annotation task. For example, using the Count Diff distance for bounding boxes, we see that the $\sigma$ of 0.3736 only considers agreement on the number of boxes and ignores the tendency for annotators to agree on their boundaries. Diagnosing a low $\sigma$ more generally can involve looking at histograms of the $D_o$ and $D_e$ distributions. Examining the behavior of $\sigma$ across various datasets and distance functions, we find some common pathologies:

► **Mode of expected distances near the low boundary.** As seen in the example of numeric vectors, this case can be common for lower-dimensional interval spaces. As $\sigma$ is developed for more complex annotation tasks with large response spaces, something like Checco et al. [13]'s $\Phi$ may be more appropriate for interval annotations. This is also seen for certain distance functions that greatly constrain the response space, such as Count Diff.

► **Mode of observed distances near the high boundary.** This case can occur when there is a common cause of zero-credit comparison between annotations. For example, in the NER task, the stricter the distance function (e.g. ranges and tags must match exactly) the greater the density at the high-boundary mode becomes for observed distances (see Figure 4b).

► **Multi-modal observed or expected distances.** Both of the previous two cases can also be instances where the distance distributions exhibit multiple modes. A general concern is whether the complex response space is actually contracted in some way due to distance function or task design. For example, the task may have a first step asking whether an object exists in an image (binary response), followed by an image annotation if it exists at all. This example could result in a very tall mode for the instances where the first step precludes any following complex annotation.

Absent these problems, a low $\sigma$ score might just mean low agreement. For example, the Keypoints dataset is simulated with a large amount of noise in the location, rotation, and magnitude of the keypoint annotations, perhaps justifying its low $\sigma = 0.5666$.

## 6.2 Comparing Distance Functions

To assess how well our KS approach for choosing a distance function works, we compare the KS rankings of distance functions against "expected" orderings. The intuition behind these expected orderings stems from two lines of prior work. First, for each specific task or domain, prior work has often established current state-of-the-art distance functions (or evaluation metrics) for each task. Second, prior CST work [22] also induced an ordering of distance functions across different tasks, showing that that weaker distance functions were less sensitive to detecting label error, with less correlation observed between annotator agreement and simulated errors. During development we also conducted CST experiments and confirmed (unsurprisingly) that our agreement measures correlate negatively with increasing injected noise for stronger distance functions (Figure 5 in Appendix). Furthermore, our KS approach yields consistent distance function orderings with prior work and without requiring a task-specific noise simulator, which is particularly useful for novel annotation tasks lacking prior work. One interesting finding is that "fine-grained" distance functions that capture more meaningful information than "coarse" ones tend to perform better.

We now discuss the resulting IAA scores seen in Table 1.

**Numeric Vector.** All three measures show the finer-grained Euclidean distance outperforming mean element-wise binary exact match, consistent with Antoine et al. [3]'s recommendation of Euclidean distance over binary for use on ordinal annotations.

**Translations.** Both of our measures yield the expected order of distance functions (from best to worst): BERTScore, GLEU, BLEU, Levenshtein. Note that using Krippendorff's $\alpha$ here to compare the means rather than the full distributions of expected and observed differences ranks Levenshtein above BLEU and GLEU.

**Bounding boxes.** Krippendorff's $\alpha$ is actually highest by far for L2, again indicating its inappropriateness for comparing distance functions simply in terms of levels of disagreement. Both of our measures on the other hand yield the expected order of distance functions (from best to worst): GIoU, IoU, L2, Count Diff.

**Named Entity Recognition.** As expected, the coarsest "Count Diff" distance function is the least discriminating. We see lenient distance functions slightly outperform stricter ones, particularly leniency in the range, according to KS. Whereas leniency in range creates finer-granularity partial-credit in measuring distance, leniency in tag (i.e., ignoring tags) actually makes the distance measure coarser since categorical tags do not have any obvious notion of "nearby" for awarding partial credit, and ignoring tags entirely makes the distance function less discriminating. While in prior work, models trained on this dataset were evaluated using a strict metric [37], a more lenient metric that gives partial credit for range overlap provides finer-granularity of distance for calculating IAA. Our KS findings are consistent with CST findings [23] on other datasets: finer-grained distance measures beat coarse, binary ones.

**Keypoints.** The order of distance functions under KS from best to worst matches expectations: OKS, IoU, and Count Diff.

**Parse Trees.** Compared to Skjærholt [42], our dataset shows similar KS and $\sigma$ scores across all three distance functions. The KS score for $\alpha_{\text{diff}}$ is slightly worse than the others, also consistent.

**Ranked lists.** As expected, the coarser measure of Kendall's $\tau$ over only the top-5 ranks performs worst. Over the full ranking, both distance functions yield similar values for all agreement metrics. This is unsurprising because both correlation functions tend to return similar p-values for the same given data [28].

# 7 CONCLUSION

Because human annotations are pivotal in training and testing machine learning systems, it is important to have both reliable labels and effective ways to assess label quality. This is challenging due to the many possible sources of label disagreement and great variation in the nature of annotation across different labeling tasks, especially with "complex" labeling tasks [10, 11] having large (finite or continuous) answer spaces. A common approach, inter-annotator agreement (IAA), supports assessment label quality on the basis of agreement between annotators, without assumption of any oracle "gold standard". However, most IAA methods do not generalize to complex labeling tasks. While the most general (and less known) form of Krippendorff's $\alpha$ [18] can be used, we showed two key limitations of it: difficulty identifying suitable distance functions and interpreting $\alpha$ across tasks and distance functions.

To address this, we described two novel IAA measures that offer greater conceptual and empirical interpretability than $\alpha$ for assessing when human annotations for complex labeling tasks are "good enough" to be used. Empirical testing across seven diverse complex annotation tasks shows how these measures add great value toward assessing IAA for complex annotations.

Various limitations remain for future work. For example, once we have isolated as best as possible the *global* sources of disagreement from noise and distance function, how do we go further in diagnosing the contributions from annotator and item heterogeneity, without which we cannot fully understand IAA? How do we use IAA to predict how useful annotations will be after aggregation?

# ACKNOWLEDGMENTS

We thank the many talented Amazon Mechanical Turk workers who contributed to our study. This research was supported in part by the Knight Foundation, the Micron Foundation, and Good Systems (https://goodsystems.utexas.edu), a UT Austin Grand Challenge to develop responsible AI technologies. Our opinions are entirely our own.

# REFERENCES


[1] Omar Alonso. 2013. Implementing crowdsourcing-based relevance experimentation: an industrial perspective. *Information retrieval* 16, 2 (2013), 101–120.

[2] Omar Alonso. 2019. The practice of crowdsourcing. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 11, 1 (2019), 1–149.

[3] Jean-Yves Antoine, Jeanne Villaneau, and Anaïs Lefeuvre. 2014. Weighted Krippendorff's alpha is a more reliable metrics for multi-coders ordinal annotations: experimental studies on emotion, opinion and coreference annotation.. In *EACL 2014*. 10–p.

[4] Ines Arous, Jie Yang, Mourad Khayati, and Philippe Cudré-Mauroux. 2020. Opencrowd: A human-ai collaborative approach for finding social influencers via open-ended answers aggregation. In *Proceedings of The Web Conference 2020*. 1851–1862.

[5] Lora Aroyo, Matthew Lease, Praveen Paritosh, and Mike Schaekermann. 2022. Data Excellence for AI: Why Should You Care. *ACM Interactions* 29, 2 (2022). March-April.

[6] Lora Aroyo and Chris Welty. 2013. Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. *WebSci2013. ACM* 2013, 2013 (2013).

[7] Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics* 34, 4 (2008), 555–596.

[8] Frank B Baker and Seock-Ho Kim. 2004. *Item response theory: Parameter estimation techniques*. CRC press.

[9] Steve Branson, Grant Van Horn, and Pietro Perona. 2017. Lean crowdsourcing: Combining humans and machines in an online system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7474–7483.

[10] Alexander Braylan and Matthew Lease. 2020. Modeling and Aggregation of Complex Annotations via Annotation Distances. In *Proceedings of The Web Conference 2020*. 1807–1818.

[11] Alexander Braylan and Matthew Lease. 2021. Aggregating Complex Annotations via Merging and Matching. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 86–94.

[12] Ben Carterette and Ian Soboroff. 2010. The effect of assessor error on IR system evaluation. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. 539–546.

[13] Alessandro Checco, Kevin Roitero, Eddy Maddalena, Stefano Mizzaro, and Gianluca Demartini. 2017. Let's agree to disagree: Fixing agreement measures for crowdsourcing. In *Fifth AAAI Conference on Human Computation and Crowdsourcing*.

[14] COCO. 2020. Common Objects in Context (COCO). https://cocodataset.org/ Accessed: 2021-10-18.

[15] Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20, 1 (1960), 37–46.

[16] Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76, 5 (1971), 378.

[17] W Nelson Francis and Henry Kucera. 1979. Brown corpus manual. *Letters to the Editor* 5, 2 (1979), 7.

[18] Klaus Krippendorff. 2004. Reliability in content analysis: Some common misconceptions and recommendations. *Human communication research* 30, 3 (2004), 411–433.

[19] Jiyi Li. 2020. Crowdsourced Text Sequence Aggregation Based on Hybrid Reliability and Representation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1761–1764. https://doi.org/10.1145/3397271.3401239

[20] Jiyi Li and Fumiyo Fukumoto. 2019. A Dataset of Crowdsourced Word Sequences: Collections and Answer Aggregation for Ground Truth Creation. In *Proceedings of the First Workshop on Aggregating and Analysing Crowdsourced Annotations for NLP*. 24–28.

[21] Frank J Massey Jr. 1951. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association* 46, 253 (1951), 68–78.

[22] Yann Mathet, Antoine Widlöcher, Karën Fort, Claire François, Olivier Galibert, Cyril Grouin, Juliette Kahn, Sophie Rosset, and Pierre Zweigenbaum. 2012. Manual corpus annotation: Giving meaning to the evaluation metrics. In *International Conference on Computational Linguistics*. 809–818.

[23] Yann Mathet, Antoine Widlöcher, and Jean-Philippe Métivier. 2015. The unified and holistic method gamma ($\gamma$) for inter-annotator agreement measure and alignment. *Computational Linguistics* 41, 3 (2015), 437–479.

[24] David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proceedings of the main conference on human language technology conference of the North American Chapter of the Association of Computational Linguistics*. Association for Computational Linguistics, 152–159.

[25] Arshad Muhammad Mehar, Kenan Matawie, and Anthony Maeder. 2013. Determining an optimal value of K in K-means clustering. In *2013 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, 51–55.

[26] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)* 54, 6 (2021), 1–35.

[27] Reshef Meir, Ofra Amir, Gal Cohensius, Omer Ben-Porat, Tsviel Ben-Shabat, and Lirong Xia. 2020. Truth Discovery via Average Proximity. arXiv:1905.00629 [cs.AI]

[28] Jerome L Myers, Arnold D Well, and Robert F Lorch Jr. 2013. *Research design and statistical analysis*. Routledge.

[29] An Thanh Nguyen, Matthew Halpern, Byron C. Wallace, and Matthew Lease. 2016. Probabilistic Modeling for Crowdsourcing Partially-Subjective Ratings. In *Proceedings of the 4th AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. 149–158.

[30] An T Nguyen, Byron C Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. 2017. Aggregating and predicting sequence labels from crowd annotations. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, Vol. 2017. NIH Public Access, 299.

[31] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. 2021. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. In *Thirty-fifth Conference on Neural Information Processing Systems: Datasets and Benchmarks Track*.

[32] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 311–318.

[33] Aditya Parameswaran, Akash Das Sarma, and Vipul Venkataraman. 2016. Optimizing open-ended crowdsourcing: The next frontier in crowdsourced data management. *Bulletin of the Technical Committee on Data Engineering* 39 (2016).

[34] Silviu Paun, Ron Artstein, and Massimo Poesio. 2022. Statistical Methods for Annotation Analysis. *Synthesis Lectures on Human Language Technologies* 15, 1 (2022), 1–217.

[35] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 658–666.

[36] Matteo Ruggero Ronchi and Pietro Perona. 2017. Benchmarking and error diagnosis in multi-instance pose estimation. In *Proceedings of the IEEE international conference on computer vision*. 369–378.

[37] Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*. 142–147.

[38] Falk Scholer, Diane Kelly, Wan-Ching Wu, Hanseul S Lee, and William Webber. 2013. The effect of threshold priming and need for cognition on relevance calibration and assessment. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 623–632.

[39] William A Scott. 1955. Reliability of content analysis: The case of nominal scale coding. *Public opinion quarterly* (1955), 321–325.

[40] Satoshi Sekine and Michael Collins. 1997. EvalB: a bracket scoring program. http://nlp.cs.nyu.edu/evalb/

[41] Shilad Sen, Margaret E Giesel, Rebecca Gold, Benjamin Hillmann, Matt Lesicko, Samuel Naden, Jesse Russell, Zixiao Wang, and Brent Hecht. 2015. Turkers, scholars," arafat" and" peace" cultural communities and algorithmic gold standards. In *Proceedings of the 18th acm conference on computer supported cooperative work & social computing*. 826–838.

[42] Arne Skjærholt. 2014. A chance-corrected measure of inter-annotator agreement for syntax. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 934–944.

[43] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 254–263.

[44] Yuandong Tian and Jun Zhu. 2012. Learning from crowds in the presence of schools of thought. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 226–234.

[45] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems* 22 (2009).

[46] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016).

[47] Liu Yang and Rong Jin. 2006. Distance metric learning: A comprehensive survey. *Michigan State Universiy* 2, 2 (2006), 4.

[48] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).

# 8 APPENDIX

**Distance functions**

**Vectors: Binary**

$$D(a,b) = 1 - \frac{1}{N}\sum_{i}^{N} 1_{a_i=b_i}$$

**Vectors: Euclidean**

$$D(a,b) = 1 - \text{RMSE}(a,b)$$

$$\text{RMSE}(a,b) = \sqrt{\frac{1}{N}\sum_{i}^{N}(a_i - b_i)^2}$$

**Translations: Levenshtein**
https://en.wikipedia.org/wiki/Levenshtein_distance
(on tokens not characters)

**Translations: BLEU**

$$D(a,b) = \frac{\text{bleu}(a,b) + \text{bleu}(b,a)}{2}$$

bleu: nltk version 3.4.5 sentence_bleu smoothing method 4

**Translations: GLEU**

$$D(a,b) = \frac{\text{gleu}(a,b) + \text{gleu}(b,a)}{2}$$

gleu: nltk version 3.4.5 sentence_gleu smoothing method 4

**Translations: BERTScore**

$D(a,b) = 1 - F1$ using BERTScorer.score(a, b) from bert_score version 0.3.10

**Various: Count Diff**

$$D_m(A,B) = ||A| - |B||$$

**Bounding Box and Keypoints: single to multi**

$$D_m(A,B) = \frac{\Delta(A,B) + \Delta(B,A)}{2}$$

$$\Delta(A,B) = \mathbb{E}\{\min(\{D_s(a,b) \mid b \in B\}) \mid a \in A\}$$

**Bounding Box: L2**

$$D_s(a,b) = 1 - \frac{\text{RMSE}(a_0,b_0) + \text{RMSE}(a_F,b_F)}{20}$$

**Bounding Box: IoU Score**

$$D_s(a,b) = \frac{\cap(a,b)}{\text{AREA}(a) + \text{AREA}(b) - \cap(a,b)}$$

**Bounding Box: GIoU Score** Adjustment to IoU described in Rezatofighi et al. [35].

**Keypoints: OKS Score** OKS distance function described in Ruggero Ronchi and Perona [36].

**NER: single to multi**

$$D_m(A,B) = 1 - \frac{2\Delta(A,B)\Delta(B,A)}{\Delta(A,B) + \Delta(B,A)}$$

**NER: both lenient**

$$\Delta(A,B) = \mathbb{E}\{\frac{\sum_{t\in a}\sum_{s\in b|b\in B} 1_{s=t}}{|\{t \in a\}|} \mid a \in A\}$$

**NER: strict tag**

$$\Delta(A,B) = \mathbb{E}\{\frac{\sum_{t\in a}\sum_{s\in b|b\in B} 1_{s=t \wedge \text{TAG}(a)=\text{TAG}(b)}}{|\{t \in a\}|} \mid a \in A\}$$

**NER: strict range**

$$\Delta(A,B) = \frac{\sum_{a\in A}\sum_{b\in B} 1_{a=b}}{|A|}$$

**NER: both strict**

$$\Delta(A,B) = \frac{\sum_{a\in A}\sum_{b\in B} 1_{a=b \wedge \text{TAG}(a)=\text{TAG}(b)}}{|A|}$$

**Parse Trees: $\alpha_{\text{plain}}$**

$$D(a,b) = \text{TED}(a,b)$$

TED: zss version 1.2.0 simple_distance

**Parse Trees: $\alpha_{\text{diff}}$**

$$D(a,b) = \text{TED}(a,b) - |\text{NLEAVES(a)} - \text{NLEAVES(b)}|$$

**Parse Trees: $\alpha_{\text{norm}}$**

$$D(a,b) = \frac{\text{TED}(a,b)}{\text{NLEAVES(a)} + \text{NLEAVES(b)}}$$

**Ranked Lists: Kendall's $\tau$**
scipy version 1.3.1 stats.kendalltau

**Ranked Lists: Spearmans' $\rho$**
scipy version 1.3.1 stats.spearmanr

| Dataset | Annotators | Items | Annotations |
|---|---|---|---|
| Vectors | 38 | 100 | 1000 |
| Translations | 70 | 250 | 2490 |
| Bounding Box | 196 | 200 | 1723 |
| NER | 46 | 199 | 982 |
| Keypoints | 100 | 199 | 1000 |
| Parse Trees | 24 | 128 | 512 |
| Ranked Lists | 30 | 100 | 600 |

**Table 2: Datasets used and summary statistics**

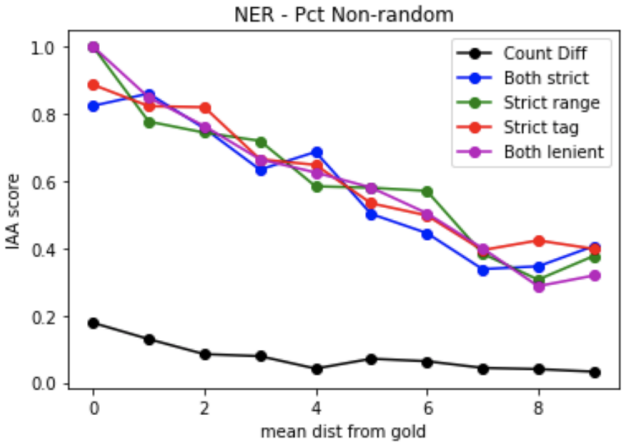



**Figure 5: Unsurprisingly, $\sigma$ measure decreases with increased noise (expected distance from gold), NER example.**

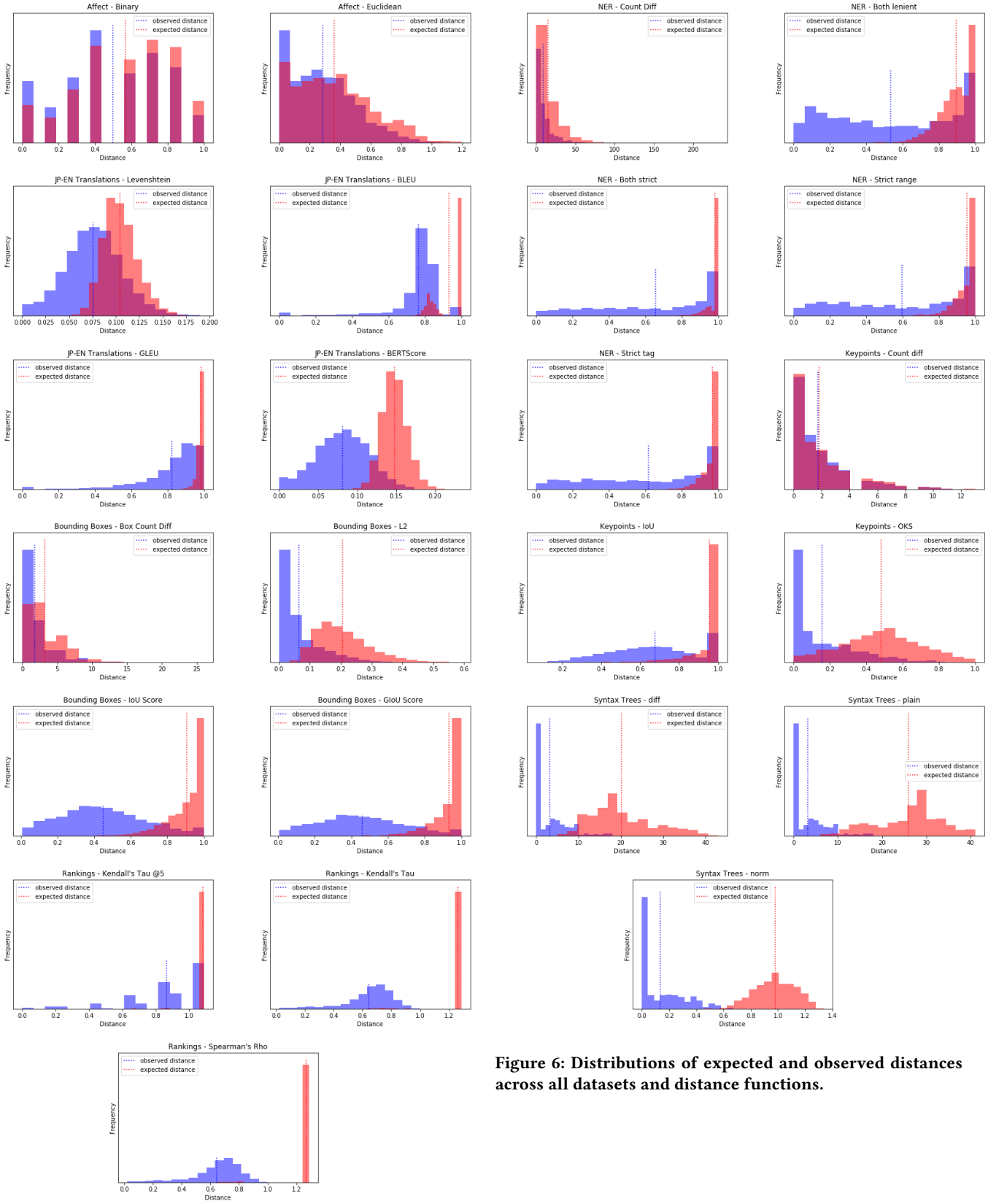



**Figure 6: Distributions of expected and observed distances across all datasets and distance functions.**