# NLPStatTest: A Toolkit for Comparing NLP System Performance

**Haotian Zhu** **Denise Mak** **Jesse Gioannini** **Fei Xia**
University of Washington, Seattle, USA
{haz060, dpm3, jessegio, fxia}@uw.edu

## Abstract

Statistical significance testing based on the $p$-value is commonly used to compare NLP system performance; however, $p$-values alone are insufficient because statistical significance is different from practical significance, the latter of which can be estimated by effect size. In this paper, we propose a three-stage procedure for comparing NLP system performance and build a toolkit, NLPStatTest, which automates the process. Users just need to upload NLP system evaluation scores; the toolkit will analyze these scores, run appropriate significance tests, estimate effect size, and conduct power analysis to estimate Type II error. Thus the toolkit provides a convenient and systematic way for comparing NLP system performance that goes beyond statistical significance testing.

## 1 Introduction

In the NLP field, to demonstrate that the improvement exhibited by a proposed system over the baseline is not due to mere happenstance, the common practice is to use statistical significance testing (Dror et al., 2018, 2020). However, as emphasized by the American Statistical Association, *"a p-value, or statistical significance, does not measure the size of an effect or the importance of a result"* (Wasserstein and Lazar, 2016); in other words, statistical significance is different from practical significance, but the latter is rarely discussed in the NLP field.

To address this issue, we propose a three-stage procedure for comparing NLP system performance as in Fig. 1. The first stage is to build an NLP system, and one can use *prospective power analysis* to compute an appropriate sample size when choosing a test corpus. The second stage is statistical hypothesis testing, where we stress the need of exploratory data analysis to verify test assumptions made by statistical significance tests and the importance of estimating the effect size and conducting power analysis. The last stage is to report various results produced by the second stage.

To automate the process, we build a toolkit, NLPStatTest. For the rest of this paper, we introduce the three-stage comparison procedure (§2), and then describe the the main components (§3) and implementation details (§4) of NLPStatTest.

## 2 Comparing NLP System Performance

In this section we briefly describe the three-stage comparison procedure and define terms that are relevant to NLPStatTest, and more detail of Stage 2 can be found in §3-§4.

### 2.1 Building an NLP System

The first stage is to build an NLP system, run it on a test set, and compare system output with gold standard. The output of this stage is a list of numerical values such as accuracy or F-scores.

**Definition 1 (Evaluation unit).** Let $(x_j, y_j)$ be a test instance. An evaluation unit (EU) $e = \{(x_j, y_j), j = 1, \cdots, m\}$ is a set of test instances on which an evaluation metric can be meaningfully defined. A test set is a set of EUs.

**Definition 2 (Evaluation metric).** Given an NLP system $A$, the evaluation metric $M$ is a function that maps an EU $e$ to a numerical value:

$$M_A(e) = M\Big( \{(\hat{y}_j, y_j),\ j = 1, \cdots, m\} \Big) \quad (1)$$

where $\hat{y}_j = A(x_j)$ is the system output of $A$ given $x_j$ and $m$ is the size of $e$ (i.e., the number of test instances in $e$).

An EU may contain one or more test instances. For example, a BLEU score can be computed on one or more sentences. The EU size affects sample size, $p$-value, sample standard deviation, effect size and so on, and it is thus one of the parameters that users can set when using NLPStatTest.

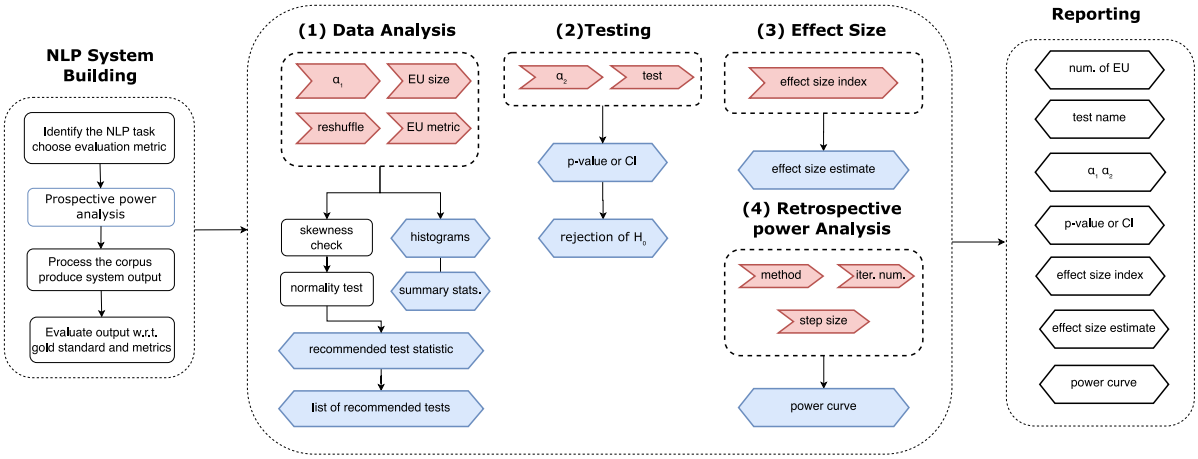


Figure 1: The three-stage procedure for comparing NLP system performance. The pink flag boxes are the parameters that users can either set or use the default values provided by `NLPStatTest`. The blue hexagons are system output of `NLPStatTest`. $\alpha_1$ and $\alpha_2$ are the significance levels for normality test and statistical significance test respectively. EU stands for evaluation unit.

## 2.2 The Comparison Stage

The second stage is the comparison stage which has four steps (see the big box in Fig. 1).

### 2.2.1 Data Analysis

When we compare two NLP systems $A$ and $B$, the output of Stage 1 is a set of pairs, $\{(M_A(e_i), M_B(e_i))\}$, where $e_i$ is the $i^{th}$ EU, and $M_A(e)$ (similarly $M_B(e)$) is defined in Eq 1.

Many statistics tests make certain assumptions about the sample (e.g., normality for $t$ test), thus it is important to conduct data analysis to determine whether those assumptions are met in order to choose significance tests that are appropriate for this particular sample. For instance, if the sample does not fall into any known distribution (e.g., normal), non-parametric tests should be used.

`NLPStatTest`will estimate sample skewness and run normality test, which will help users to choose an appropriate test statistic and a significance test.

### 2.2.2 Statistical Significance Testing

The second step in Stage 2 is statistical significance testing, using two mutually exclusive hypotheses: the null hypothesis $H_0$ and the alternative $H_1$. For comparing two NLP systems, usually a (paired) two-sample testing is used, which can be reduced to a one-sample testing by using pairwise difference. `NLPStatTest` currently only considers one sample testing cases for numerical data, where $i.i.d.$ within sample is assumed, with $H_0$ being the chosen test statistic $T$ (mean or median) is 0.

To run a statistical significance test, users first determine the test statistic $T$ (mean or median) and a significance test based on data analysis results (e.g., the sample skewness estimate and normality test results). Then users choose the significance level $\alpha$, which is often set to 0.05 or 0.01 in the NLP field. `NLPStatTest` will then calculate $p$-value and reject $H_0$ if and only if $p$-value $< \alpha$.

### 2.2.3 Effect Size Estimation

In most experimental NLP papers where statistical significance testing is employed, the $p$-value is the sole reported quantity. However, the $p$-value is often misused and misinterpreted. For instance, statistical significance is easily conflated with practical significance; as a result, NLP researchers often run statistical significance tests to show that the performances of two NLP systems are different (i.e., statistical significance), without measuring the degree or the importance of such a difference (i.e., practical significance).

Cohen (1990) noted *"the null hypothesis, if taken literally, is always false in the real world."* For instance, the evaluation metric values (e.g., F-scores) of two NLP systems on a test set are almost never exactly the same; thus, $H_0$, which says that two systems perform equally, is (almost) always false. When $H_0$ is false, the $p$-value will eventually approach zero in large samples (Lin et al., 2013). In other words, no matter how tiny the system performance difference is, there is always a large enough dataset on which the difference is statistically significant. Therefore, statistical significance

is markedly different from practical significance.

One way to measure practical significance is by estimating *effect size*, which is defined as the degree to which the 'phenomenon' is present in the population, or the degree to which the null hypothesis is false (Cohen, 1994). While the need to estimate and report effect size has long been recognized in other fields (Tomczak and Tomczak, 2014), the same is not true in the NLP field. We include several methods for estimating effect size in `NLPStatTest` (see §3.3).

#### 2.2.4 Power Analysis

There are two types of errors in hypothesis testing: Type I and Type II errors. The Type I error of a significance test, often denoted by $\alpha$, is the probability that, when $H_0$ is true, $H_0$ is rejected by the test. The Type II error of a significance test, usually denoted by $\beta$, is the probability that under $H_1$, $H_1$ is rejected by the test. While Type I error can be controlled by predetermining the significance level, Type II error can be controlled or estimated by power analysis.

**Definition 3 (Statistical power).** The power of a statistical significance test is the probability that under $H_1$, $H_0$ is correctly rejected by the test. The power of a test is $1 - \beta$.

Higher power means that statistical inference in a study is more correct and accurate (Perugini et al., 2018). While power analysis is rarely used in the NLP field, it is considered good standard practice in some other scientific fields such as psychology and clinical studies (Perugini et al., 2018); thus, we implement two methods of conducting power analysis in `NLPStatTest`(see §3.4).

### 2.3 Reporting Test Results

Beside $p$-value, it is important to report other quantities in order to make the studies reproducible and available for meta-analysis, including the name of statistical significance test, the predetermined significance level $\alpha$, effect size estimate/estimator, the sample size used, and statistical power.

## 3 System Design

`NLPStatTest` is a toolkit that automates the comparison procedure, and it has four main steps as shown in the big box in Fig. 1. To use `NLPStatTest`, users provide a data file with the NLP system performance scores produced in Stage 1. `NLPStatTest` will prompt users to either modify or use the default values for the parameters in the pink flags and then produce the output in the blue hexagons. The users can then report (some of) the output in Stage 3 of the comparison procedure.

### 3.1 Data Analysis

The first step of the comparison stage is data analysis, and a screenshot of this step is in Fig. 2. The top part (above the *Run* button) shows the input and parameters that the user needs to provide, and the bottom part (below the *Run* button) shows the output of the data analysis step.

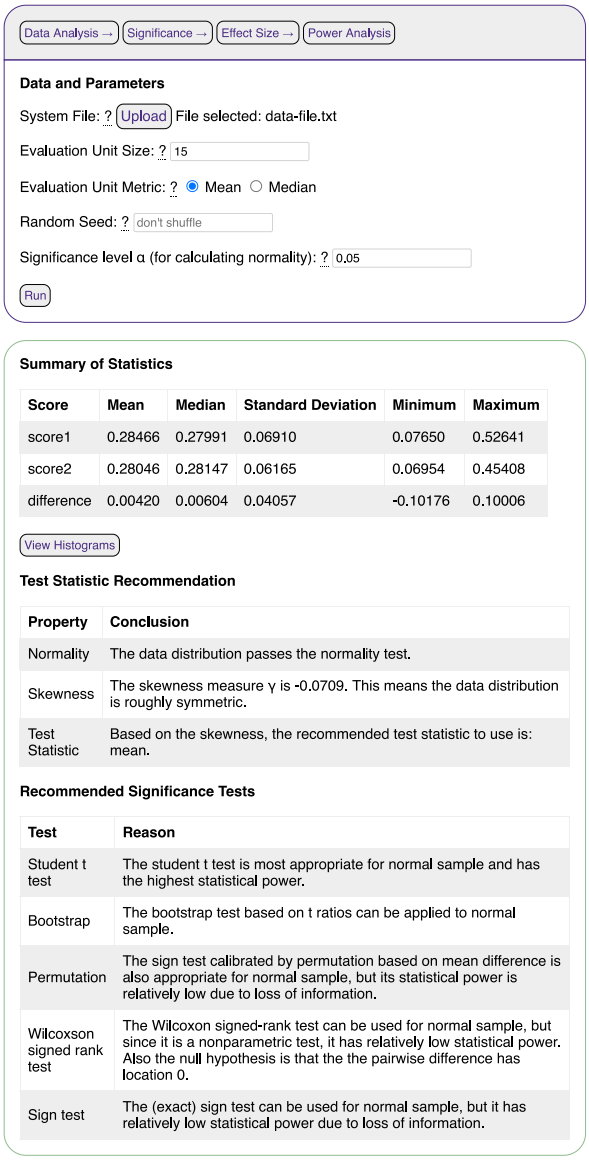


Figure 2: Screenshot of the data analysis step. The part above the *Run* button are parameters that users can set, and the part below is `NLPStatTest` output.

#### 3.1.1 The Input Data File

To compare two NLP systems, *A* and *B*, the user needs to provide a data file where each line is a pair of numerical values. There are two scenarios. In the first scenario, the pair is $(u_i, v_i)$, where $u_i$=$M_A(e_i)$ is the evaluation metric value (e.g., accuracy or F-score) of an EU $e_i$ given System *A* (see Eq. 1), and $v_i$=$M_B(e_i)$.

In the second scenario, if $u_i$ and $v_i$ can be calculated as the mean or the median of the evaluation metric values of test instances in $e_i$, the user can upload a data file where each line is a pair of $(a_k, b_k)$, where $a_k$ and $b_k$ are the evaluation metric values of a test instance $t_k$ given System *A* and *B*, respectively. The user then chooses the EU size $m$ and specifies whether the EU metric value should be calculated as the mean or the median of the metric values of the instances in the EU. `NLPStatTest` will use $m$ adjacent lines in the file to calculate $u_i$ and $v_i$. If the user prefers to randomly shuffle the lines before calculating $u_i$ and $v_i$, he can provide a seed for random shuffling.

#### 3.1.2 Histograms and Summary Statistics

From the $(u_i, v_i)$ pairs, `NLPStatTest` generates descriptive summary statistics (e.g., mean, median, standard deviation) and histograms of three datasets, $\{u_i\}$, $\{v_i\}$, and $\{u_i - v_i\}$, as shown in the first table and the three histograms in Fig. 2.

#### 3.1.3 Central Tendency Measure

Most statistical hypothesis testing problems are based on testing for the mean as a measure for central tendency (average system performance). Many statistical tests are also based on means ($t$ test, bootstrap test based on $t$ ratios, etc). However, the distribution of the data is not necessarily symmetric, where the population mean does not measure the central tendency. In that case, the median should be used as a more robust measure. Another issue associated with mean is that if the distribution is heavy-tailed (e.g. $t$ distribution, Cauchy distribution), the sample mean oscillates dramatically.

In order to examine the symmetry of the underlying distribution, `NLPStatTest` checks the skewness of $\{u_i - v_i\}$ by estimating the sample skewness ($\gamma$). Based on the $\gamma$ value, we use the following rule of thumb (Bulmer, 1979) to determine whether `NLPStatTest` would recommend to use mean or median as the test statistic for statistical significance test:

- $|\gamma| \in [0, 0.5)$: roughly symmetric (use mean)
- $|\gamma| \in [0.5, 1)$: moderately skewed (use median)
- $|\gamma| \in [1, \infty)$: highly skewed (use median)

#### 3.1.4 Normality Test

To choose a good significance test for $\{u_i - v_i\}$, we need to determine whether the data is normally distributed. If normally distributed, the student $t$ test is the most appropriate (and powerful) test; if not, then nonparametric tests which do not rely on normality assumption might be more proper.

If a distribution is skewed according to $\gamma$, there is no need to run normality test as the data is not normally distributed. For a non-skewed distribution, `NLPStatTest` will run the Shapiro-Wilk normality test (Shapiro and Wilk, 1965), which is a statistical test itself and the user can choose the significance level for that ($\alpha_1$ in Fig 1).

#### 3.1.5 Recommended Significance Tests

Based on the skewness check and normality test result, `NLPStatTest` will recommend a test statistic (mean or median) and appropriate significance tests (e.g., $t$ test if $\{u_i - v_i\}$ is normally distributed).

### 3.2 Testing

In this step, the user sets the significance level ($\alpha_2$ in Fig 1) and chooses a significance test from the ones recommended in the previous step. If the test has any parameter (e.g., the number of trials for bootstrap test $B$), `NLPStatTest` will suggest a default value which can be changed by the user. `NLPStatTest` will then run the test, calculate $p$-value (or provide a confidence interval for the bootstrap test), and reject $H_0$ if $p$-value $< \alpha_2$ (or the value of test statistic under $H_0$ lies outside the confidence interval).

### 3.3 Effect Size

Effect size can be estimated in several ways (called *effect size indices*), depending on data types (numerical or categorical) and significance tests used. Dror et al. (2020) defined effect size as the (unstandardized) difference between system performance, while Hauch et al. (2012), Lauscher and Glavaš (2019) and Pimentel et al. (2019) used the standardized mean difference.

`NLPStatTest` implements the following four effect size indices. Once users select one or more indices, `NLPStatTest` will calculate effect size accordingly and display the results.

**Cohen's** $d$ estimates the standardized mean difference by

$$d = \frac{\hat{u} - \hat{v}}{\hat{\sigma}} \tag{2}$$

where $\hat{v}$ and $\hat{u}$ are the sample means and $\hat{\sigma}$ denote standard deviation of $u - v$. Cohen's $d$ assumes normality and is one of the most frequently used effect size indices. If Cohen's $d$, or any other effect size indices depending on $\hat{\sigma}$, is used to estimate effect size, the EU size will affect the standard deviation and thus effect size.

**Hedges'** $g$ adjusts the bias brought by Cohen's $d$ in small samples by the following:

$$g = d \cdot \left(1 - \frac{3}{4n - 9}\right) \tag{3}$$

where $n$ is the size of $\{u_i - v_i\}$.

**Wilcoxon** $r$ is an effect size index for the Wilcoxon signed rank test, calculated as $r = \frac{Z}{\sqrt{n}}$, where

$$Z = \frac{W - n(n+1)/4}{\sqrt{\frac{n(n+1)(2n+1)}{24} - \frac{\sum_{t \in T} t^3 - \sum_{t \in T} t}{48}}} \tag{4}$$

Here, $W$ is the test statistic for Wilcoxon signed rank test and $T$ is the set of tied ranks.

**Hodges-Lehmann Estimator** (Hodges and Lehmann, 1963) is an estimator for the median. Let $w_i = u_i - v_i$. The $HL$ estimator for one-sample testing is given by

$$HL = \text{median}\left(\{(w_i + w_j)/2,\ i \neq j\}\right) \tag{5}$$

### 3.4 Power Analysis

Power (as in Definition 3) is associated with sample size, effect size and the significance level $\alpha$. In particular, power increases with larger sample size, effect size, and $\alpha$.

#### 3.4.1 Two Common Types of Power Analysis

There are two common types of power analysis. One is *prospective power analysis*, which is used when planning a study (usually in clinical trials) in order to decide how many subjects are needed. In the NLP field, we can conduct this type of power analysis to determine how big a test corpus needs to be in order to ensure that the significance test reaches the desired power level.

The other type is called *retrospective* or *post-hoc power analysis*, usually done after a significance test to determine the relation between sample size and power. `NLPStatTest` implements this type of power analysis because the test corpora for NLP tasks are often predetermined.

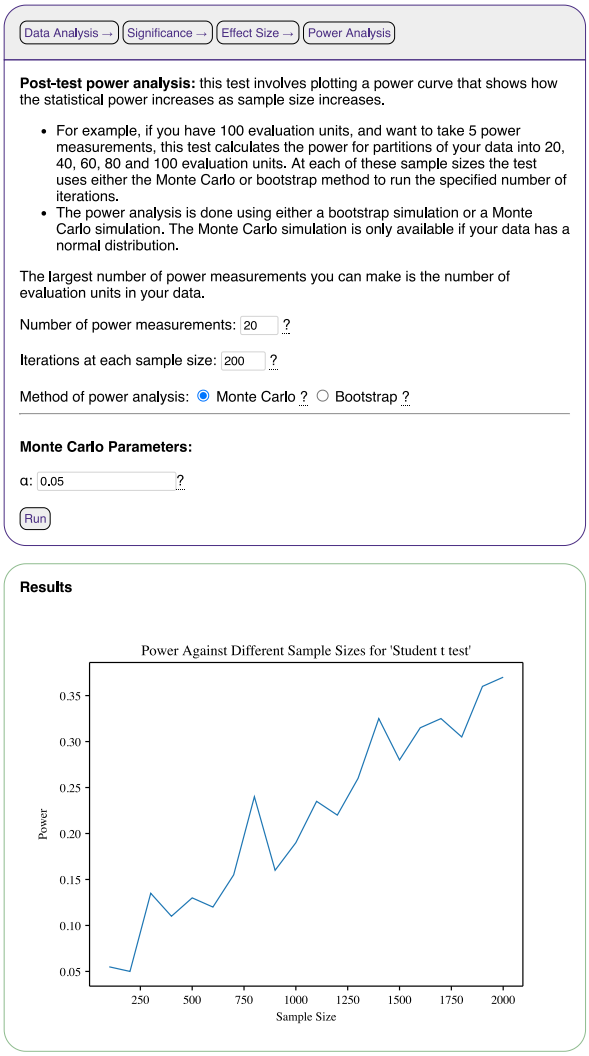


Figure 3: Screenshot for power analysis step.

#### 3.4.2 Methods for Power Analysis

There are two scenarios for conducting power analysis. When the values in $\{u_i - v_i\}$ are from a known distribution, one can use *Monte Carlo simulation* to directly simulate from this known distribution. To do this, one has to have an informed guess of the desired effect size (i.e., mean difference) via meta-analysis of previous studies.

When the distribution of the sample is unknown a priori, one can resample with replacement from the empirical distribution of the sample (a.k.a. the *bootstrap* method (Efron and Tibshirani, 1993)) to estimate the power. This method works well

for retrospective power analysis in the NLP setting because we already have the test corpus and evaluation metric values at hand.

`NLPStatTest` implements both methods. Users can employ one or both methods, `NLPStatTest` will produce a figure that shows the relation between sample size and power, as in Fig. 3.

## 4 Implementation Details

There are two ways to run `NLPStatTest`. The first way is to run a graphical user interface (GUI) locally or on the Web. Users go through the four steps and then download the system output. The web interface was implemented using HTML and JavaScript for the front end and Python for the back end. The Python code uses the `scipy` and `statsmodels` libraries for implementing statistical tests for significance, and `matplotlib` for generating the histograms and graphics. Alternatively, users can download the Python code and run it directly as a command line. The whole package is open-source and available at https://github.com/nlp-stat-test.

Running the demo at AACL-2021 requires a modern web browser (and an internet connection to access the website), and a screencast of the demo is available at https://vimeo.com/443074846.

## 5 Conclusion

While statistical significance testing has been commonly used to compare NLP system performance, a small $p$-value alone is not sufficient because statistical significance is different from practical significance. To measure practical significance, we recommend estimating and reporting of effect size. It is also necessary to conduct power analysis to ensure that the test corpus is large enough to achieve a desirable power level. Hence, we propose a three-stage procedure for comparing NLP system performance, and build a toolkit, `NLPStatTest`, to automate the testing stage of the procedure. For future work, we will extend this work to hypothesis testing with multiple datasets or multiple metrics.

## References


M. G. Bulmer. 1979. *Principles of Statistics*, page 57. Dover, New York.

J. Cohen. 1990. Things i have learned (so far). *American Psychologist*, 45(12):1304 – 1312.

J. Cohen. 1994. The earth is round (p $<$ .05). *American Psychologist*, pages 997–1003.

R. Dror, G. Baumer, S. Shlomov, and R. Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of ACL-2018 (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia.

R. Dror, L. Peled-Cohen, S. Shlomov, and R. Reichart. 2020. *Statistical Significance Testing for Natural Language Processing*. Morgan & Claypool.

B. Efron and R. J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman & Hall, New York, NY.

Valerie Hauch, Iris Blandón-Gitlin, Jaume Masip, and Siegfried Ludwig Sporer. 2012. Linguistic cues to deception assessed by computer programs: A meta-analysis. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 1–4, Avignon, France.

J. L. Hodges and E. L. Lehmann. 1963. Estimates of location based on rank tests. *Annals of Mathematical Statistics*, 34(2):598–611.

Anne Lauscher and Goran Glavaš. 2019. Are we consistently biased? multidimensional analysis of biases in distributional word vectors. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 85–91, Minneapolis, Minnesota.

Mingfeng Lin, Henry Lucas, and Galit Shmueli. 2013. Too big to fail: Large samples and the p-value problem. *Information Systems Research*, 24:906–917.

Marco Perugini, Marcello Gallucci, and Giulio Costantini. 2018. A practical primer to power analysis for simple experimental designs. *International Review of Social Psychology*, 31.

Tiago Pimentel, Arya D. McCarthy, Damian Blasi, Brian Roark, and Ryan Cotterell. 2019. Meaning to form: Measuring systematicity as information. In *Proceedings of ACL-2019*, pages 1751–1764, Florence, Italy.

S. S. Shapiro and M. B. Wilk. 1965. An analysis of variance test for normality (complete samples)†. *Biometrika*, 52(3-4):591–611.

Maciej Tomczak and Ewa Tomczak. 2014. The need to report effect size estimates revisited. an overview of some recommended measures of effect size. *Trends in Sport Sciences*, 21:19–25.

R. L. Wasserstein and N. A. Lazar. 2016. The asa statement on p-values: Context, process, and purpose. *The American Statistician*, 70(2):129–133.