

# Tool for the Automatic Analysis of Cohesion (TAACO)

## Quick Start Guide and User Manual for TAACO 1.5 (updated 7-16-2017)

Kristopher Kyle, University of Hawaii at Manoa  
Scott Crossley, Georgia State University

This document is intended to assist users of TAACO. It includes a brief explanation of how to get started with the tool, a description of how indices are calculated, and a section outlining “How to do a study with TAACO”. If this is the first time using TAACO, it is advisable to read this entire document, paying particular attention to the “How to Conduct a Study with TAACO” section. Additional information about TAACO is included in the supplementary Index Description Spreadsheet (available at [www.kristopherkyle.com](http://www.kristopherkyle.com)).

Please cite the following article when using TAALES in your work:

Crossley, S. A., Kyle, K., & McNamara, D. S. (2016). The tool for the automatic analysis of text cohesion (TAACO): Automatic assessment of local, global, and text cohesion. *Behavior Research Methods* 48(4), pp. 1227-1237. doi:10.3758/s13428-015-0651-7

### What is new in TAACO 1.5?

TAACO 1.5 represents a substantive update to TAACO 1.0. TAACO 1.5 is easier to install (no need to install Python or download NLTK) and features an updated, easy to use interface. TAACO 1.5 also includes a number of updates for the calculation of overlap indices. These updates provide more accurate and more intuitive calculations of cohesion. We suggest all TAACO users upgrade to version 1.5.

### Compatibility

TAACO 1.5 works on Mac OSX, 64-bit versions of Windows 7, 8, and 10, and 64-bit Linux Ubuntu. Compiled versions of TAACO are not supported on 32-bit versions of Windows or Linux Ubuntu<sup>1</sup>.

### Getting Started

First, download the version of TAALES that is appropriate for your operating system. Second, download and install the version of the [Java Development Kit \(JDK\)](#) that is appropriate for your operating system. Note that this is NOT the same version of Java that is installed by default on most operating systems.

---

<sup>1</sup> Advanced users with some programming knowledge should be able to use the Python 2.7 version of TAACO on 32-bit systems, but technical support is not offered.

### **Input Text Formatting**

TAACO is designed to process plain text (.txt) files with UTF-8 encoding. This is a very common encoding type, though not all word processing software will default to UTF-8 encoding. A common reason for TAACO to malfunction is text encoding. If problems occur during the use of TAACO (and particularly if an error message that includes ‘ascii codec can’t encode characters in position...’), then it is important to ensure that a) filenames only include “normal” English characters and b) texts are in UTF-8 format.

### **Options**

#### **Index Options**

TAACO 1.5 allows users to calculate all possible indices or choose to only calculate particular index types. Indices included in TAACO 1.5 are described in the “Indices” section below. Further details can be found in the TAACO 1.5 Index Description Spreadsheet, which is freely available at [www.kristopherkyle.com](http://www.kristopherkyle.com).

#### **Diagnostic Output Options**

TAACO 1.5 provides two types of diagnostic output designed to help users a) determine whether their data was processed correctly, b) understand how particular indices were calculated, and c) find examples of calculated linguistic features. The first (“diagnostic file”) includes basic information for each text such as the number of sentences, paragraphs, running word lemmas (tokens), and unique word lemmas (types) found in each input text. The second (“tagged files”) contains sentence breaks, paragraph breaks, and a variety of lemma and part of speech information for each word in the text.

If either of the diagnostic output options are selected, a folder is created that includes the user-selected output filename (e.g., “results”) with “\_diagnostic” appended to it (e.g., “results\_diagnostic”). All diagnostic files are written to this folder.

## TAACO 1.5 Indices

TAACO reports 150 indices in four main categories: TTR and Givenness, Sentence Overlap, Paragraph Overlap, and Connectives. The calculation of these indices is discussed below. A description of each index can be found in the Index Description Spreadsheet.

Note: All indices in TAACO 1.5 are calculated using the lemma form of each word (e.g., the lemma form of *is*, *am*, *are*, *was*, *were*, and *be* is *be*).

### TTR and Givenness

TAACO 1.5 calculates 14 indices related to TTR (11 indices) and Givenness (3 indices). These are briefly described below.

**TTR.** Type-token ratio (TTR) indices provide a measure of lexical diversity. TTR is calculated by dividing the number of unique lemmas in a text and dividing it by the number of total lemmas in a text. TTR indices are available for single lemmas, for bi-grams (two word phrases), tri-grams (three word phrases). See Table 1 for an example of how TTR scores are calculated. TAACO reports variants of the TTR index for various part of speech categories (e.g., nouns and verbs).

Table 1  
*TTR example*

---

Input text	This is a cool sentence. This sentence is even cooler. Those were both really cool sentences. This be a cool sentence.
Lemmatized text	This sentence be even cooler. Those were both really cool sentence.
Number of unique words (types)	10
Number of total words (tokens)	16
TTR score	<b><math>10/16 = .625</math></b>

---

**Givenness.** Givenness indices approximate the proportion of given information to new information by examining pronoun density, pronoun to noun ratios, and repeated content lemmas

and pronouns. TAACO 1.5 calculates four indices related to givenness. Each is described in Table X.

Table X  
*Givenness indices in TAACO 1.5*

Index	Calculation
pronoun density	number of third person pronouns divided by the total number of words
pronoun to noun ratio	the number third person pronouns divided by the total number of nouns
repeated content lemmas	the number of content words that are repeated at least once divided by the total number of words in the text
repeated content lemmas and pronouns	the number of content words and third person pronouns that are repeated at least once divided by the total number of words in the text

### Sentence Overlap

TAACO 1.5 calculates six basic types of sentence overlap indices. Each of these is described below. Table X includes a sample text at three stages of processing. First, a raw text is imported into TAACO. In the second stage, the text is lemmatized and separated into sentences. In the third stage, each sentence is refined so that it includes only unique lemmas (types). Overlap indices use the version of the text included in Step 3. Each description of the overlap indices will refer to the sample text in Table X.

Table X  
*Sentence overlap examples*

Step 1: Raw input text	Automatic text analysis can be very convenient. It is, however, important for users to understand how each index is calculated. In many cases, there are multiple ways that each index could be calculated. It is also important to understand what constructs an index is purported to measure.	
Step 2: Lemmatized text (separated into sentences)	LS 1	automatic text analysis can be very convenient
	LS 2	it be however important for user to understand how each index be calculate
	LS 3	in many case there be multiple way that each index could be calculate
	LS 4	it be also important to understand what construct a index be purported to measure
Step 3: Unique lemmas (tokens) in each sentence	LTS 1	automatic text analysis can be very convenient
	LTS 2	it be however important for user to understand how each index calculate
	LTS 3	in many case there be multiple way that each index could calculate
	LTS 4	it be also important to understand what construct a index purported measure

**Adjacent sentence overlap.** Adjacent sentence overlap calculates the average amount of words that are repeated between sentences. In each sentence that is followed by another sentence

(i.e., LTS 1, LTS 2, and LTS 3) each word is checked in order to determine whether it occurs in the following sentence. For each word that occurs in the following sentence at least once, the overlap count increases by one. In LTS 1, for example, only the word *be* occurs in the following sentence (LTS 2). The total index score is computed as the total number of words with adjacent sentence overlap divided by the number of words considered (i.e., the sum of the number lemma types in LTS 1, LTS 2, and LTS 3). Table X includes the calculation of adjacent sentence overlap for the example text.

Table X

*Adjacent sentence overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of lemma types
LTS 1 -> LTS 2	<i>be</i>	1	7
LTS 2 -> LTS 3	<i>be, calculate, each, index</i>	4	12
LTS 3 -> LTS 4	<i>be, index</i>	2	12
<b>Sum</b>		7	31
<b>Adjacent sentence overlap score</b>		$7 / 31 = 0.2258$	

**Adjacent sentence overlap (sentence normed).** Adjacent sentence overlap (sentence normed) is calculated in the same manner as adjacent sentence overlap, except that the denominator is the number of sentences considered. In short, the total index score is computed as the total number of words with adjacent sentence overlap divided by the number of sentences considered. Table X includes the calculation of adjacent sentence overlap (sentence normed) for the example text.

Table X

*Adjacent sentence overlap (sentence normed) calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of sentences considered
LTS 1 -> LTS 2	<i>be</i>	1	1
LTS 2 -> LTS 3	<i>be, calculate, each, index</i>	4	1
LTS 3 -> LTS 4	<i>be, index</i>	2	1
<b>Sum</b>		7	3
<b>Adjacent sentence overlap score</b>		$7 / 3 = 2.333$	

**Binary adjacent sentence overlap.** Binary adjacent sentence overlap calculates how many adjacent sentences include any overlapping items. The total index score is computed as the total number of sentences that include words that occur in the following sentence divided by the number of sentences considered. Table X includes the calculation of binary adjacent sentence overlap for the example text.

Table X

*Binary adjacent sentence overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of sentences considered
LTS 1 -> LTS 2	<i>be</i>	1	1

LTS 2 -> LTS 3	<i>be, calculate, each, index</i>	1	1
LTS 3 -> LTS 4	<i>be, index</i>	1	1
<b>Sum</b>		3	3
<b>Adjacent sentence overlap score</b>		$3 / 3 = 1$	

**Adjacent two-sentence overlap.** Adjacent two-sentence overlap calculates the average amount of words that are repeated between sentences. In each sentence that is followed by at least two sentences (i.e., LTS 1 and LTS 2) each word is checked in order to determine whether it occurs in the following two sentences. For each word that occurs in the two following sentences at least once, the overlap count increases by one. In LTS 1, for example, only the word *be* occurs in the two following sentences (LTS 2). The total index score is computed as the total number of words with adjacent two-sentence overlap divided by the number of words considered (i.e., the sum of the number lemma types in LTS 1, LTS 2). Table X includes the calculation of adjacent two-sentence overlap for the example text.

Table X

*Adjacent two-sentence overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of lemma types
LTS 1 -> LTS 2 and LTS 3	<i>be</i>	1	7
LTS 2 -> LTS 3 and LTS 4	<i>it, be, important, to, understand, each, index, calculate</i>	8	12
<b>Sum</b>		9	19
<b>Adjacent sentence overlap score</b>		$9 / 19 = .4737$	

**Adjacent two-sentence overlap (sentence normed).** Adjacent two-sentence overlap (sentence normed) is calculated in the same manner as adjacent two-sentence overlap, except that the denominator is the number of sentences considered. In short, the total index score is computed as the total number of words with adjacent two-sentence overlap divided by the number of sentences considered. Table X includes the calculation of adjacent two-sentence overlap (sentence normed) for the example text.

Table X

*Adjacent two-sentence overlap (sentence normed) calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of sentences considered
LTS 1 -> LTS 2 and LTS 3	<i>be</i>	1	1
LTS 2 -> LTS 3 and LTS 4	<i>it, be, important, to, understand, each, index, calculate</i>	8	1
<b>Sum</b>		9	2
<b>Adjacent sentence overlap score</b>		$9 / 2 = 4.5$	

**Binary two-sentence overlap.** Binary adjacent two-sentence overlap calculates how many adjacent sentences include any overlapping items. The total index score is computed as the total number of sentences that include words that occur in the following two sentences divided by the number of sentences considered. Table X includes the calculation of binary adjacent two-sentence overlap for the example text.

Table X  
*Binary two-sentence overlap calculation*

Overlap check	Overlapping lemmas	Binary overlap Count	Number of sentences considered
LTS 1 -> LTS 2 and LTS 3	<i>be</i>	1	1
LTS 2 -> LTS 3 and LTS 4	<i>it, be, important, to, understand, each, index, calculate</i>	1	1
<b>Sum</b>		2	2
<b>Adjacent sentence overlap score</b>		$2 / 2 = 1$	

### **Paragraph Overlap**

Paragraph overlap indices are identical to sentence overlap indices, except that adjacent paragraphs are examined instead of sentences. Table X includes a sample text at three stages of processing. First, a raw text is imported into TAACO. In the second stage, the text is lemmatized and separated into paragraphs. In the third stage, each paragraph is refined so that it includes only unique lemmas (types). Paragraph overlap indices use the version of the text included in Step 3. Each description of the overlap indices will refer to the sample text in Table X.

Table X

*Paragraph overlap examples*

<p>Step 1: Raw input text</p>	<p>Automatic text analysis can be very convenient. It is, however, important for users to understand how each index is calculated.</p> <p>In many cases, there are multiple ways that each index could be calculated. It is also important to understand what constructs an index is purported to measure.</p> <p>Conducting an analysis without understanding how indices are measured will at the very least make it difficult to interpret the results. Worse, this can also lead a researcher to make unwarranted conclusions about the data.</p> <p>This user manual has been created to make TAACO indices easier to understand. It is hoped that this will help both novices and experts conduct high impact studies.</p>
<p>Step 2: Lemmatized text (separated into paragraphs)</p>	<p>LP 1 automatic text analysis can be very convenient it be however important for user to understand how each index be calculate</p> <p>LP 2 in many case there be multiple way that each index could be calculate it be also important to understand what construct a index be purported to measure</p> <p>LP 3 conduct a analysis without understand how index be measure will at the very least make it difficult to interpret the result worse this can also lead a researcher to make unwarranted conclusion about the datum</p> <p>LP 4 this user manual have be create to make taaco index easier to understand it be hope that this will help both novice and expert conduct high impact study</p>
<p>Step 3: Unique lemmas (tokens) in each paragraph</p>	<p>LTP 1 automatic text analysis can be very convenient it however important for user to understand how each index calculate</p> <p>LTP 2 in many case there be multiple way that each index could calculate it also important to understand what construct a purported measure</p> <p>LTP 3 conduct a analysis without understand how index be measure will at the very least make it difficult to interpret result worse this can also lead researcher unwarranted conclusion about datum</p> <p>LTP 4 this user manual have be create to make taaco index easier understand it hope that this will help both novice and expert conduct high impact study</p>

**Adjacent paragraph overlap.** Adjacent paragraph overlap calculates the average amount of words that are repeated between paragraphs. In each paragraph that is followed by another paragraph (i.e., LTP 1, LTP 2, and LTP 3) each word is checked in order to determine whether it occurs in the following paragraph. For each word that occurs in the following paragraph at least once, the overlap count increases by one. In LTP 1, for example, eight words (*be, it, important, etc.*) occur in the following paragraph (LTP 2). The total index score is



computed as the total number of words with adjacent paragraph overlap divided by the number of words considered (i.e., the sum of the number lemma types in LTP 1, LTP 2, and LTP 3). Table X includes the calculation of adjacent paragraph overlap for the example text.

Table X

*Adjacent paragraph overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of lemma types
LTP 1 -> LTP 2	<i>be, it, important, to, understand, each, index, calculate</i>	8	18
LTP 2 -> LTP 3	<i>be, index, it, also, to, understand, a, measure</i>	8	22
LTP 3 -> LTP 4	<i>conduct, understand, index, be, will, make, it, to, this</i>	9	30
<b>Sum</b>		25	70
<b>Adjacent sentence overlap score</b>		$25 / 70 = .3571$	

**Adjacent paragraph overlap (paragraph normed).** Adjacent paragraph overlap (paragraph normed) is calculated in the same manner as adjacent paragraph overlap, except that the denominator is the number of paragraphs considered. In short, the total index score is computed as the total number of words with adjacent paragraph overlap divided by the number of paragraphs considered. Table X includes the calculation of adjacent paragraph overlap (paragraph normed) for the example text.

Table X

*Adjacent paragraph overlap (paragraph normed) calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of paragraphs considered
LTP 1 -> LTP 2	<i>be, it, important, to, understand, each, index, calculate</i>	8	1
LTP 2 -> LTP 3	<i>be, index, it, also, to, understand, a, measure</i>	8	1
LTP 3 -> LTP 4	<i>conduct, understand, index, be, will, make, it, to, this</i>	9	1
<b>Sum</b>		25	3
<b>Adjacent sentence overlap score</b>		$25 / 3 = 8.3333$	

**Binary adjacent paragraph overlap.** Binary adjacent paragraph overlap calculates how many adjacent paragraphs include any overlapping items. The total index score is computed as the total number of paragraphs that include words that occur in the following paragraph divided by the number of paragraphs considered. Table X includes the calculation of binary adjacent paragraph overlap for the example text.

Table X  
*Adjacent sentence overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of paragraphs considered
LTP 1 -> LTP 2	<i>be, it, important, to, understand, each, index, calculate</i>	1	1
LTP 2 -> LTP 3	<i>be, index, it, also, to, understand, a, measure</i>	1	1
LTP 3 -> LTP 4	<i>conduct, understand, index, be, will, make, it, to, this</i>	1	1
<b>Sum</b>		3	3
<b>Adjacent sentence overlap score</b>		3 / 3 = 1	

**Adjacent two-paragraph overlap.** Adjacent two-paragraph overlap calculates the average amount of words that are repeated between paragraphs. In each paragraph that is followed by at least two paragraphs (i.e., LTP 1 and LTP 2) each word is checked in order to determine whether it occurs in the following two paragraphs. For each word that occurs in the two following paragraphs at least once, the overlap count increases by one. In LTP 1, for example, twelve words occur, in the two following paragraphs (LTP 2). The total index score is computed as the total number of words with adjacent two-paragraph overlap divided by the number of words considered (i.e., the sum of the number lemma types in LTP 1, LTP 2). Table X includes the calculation of adjacent two-paragraph overlap for the example text.

Table X  
*Adjacent two-paragraph overlap calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of lemma types
LTP 1 -> LTP 2 and LTP 3	<i>analysis, can, be, very, it, important, to, understand, how, each, index, calculate</i>	12	18
LTP 2 -> LTP 3 and LTP 4	<i>be, that, index, it, also, to, understand, a, measure</i>	9	22
<b>Sum</b>		21	40
<b>Adjacent sentence overlap score</b>		21 / 40 = .5250	

**Adjacent two-paragraph overlap (paragraph normed).** Adjacent two-paragraph overlap (paragraph normed) is calculated in the same manner as adjacent two-paragraph overlap, except that the denominator is the number of paragraphs considered. In short, the total index score is computed as the total number of words with adjacent two-paragraph overlap divided by the number of paragraphs considered. Table X includes the calculation of adjacent two-paragraph overlap (paragraph normed) for the example text.

Table X

*Adjacent two-paragraph overlap (paragraph normed) calculation*

Overlap check	Overlapping lemmas	Overlap Count	Number of paragraphs considered
LTP 1 -> LTP 2 and LTP 3	<i>analysis, can, be, very, it, important, to, understand, how, each, index, calculate</i>	12	1
LTP 2 -> LTP 3 and LTP 4	<i>be, that, index, it, also, to, understand, a, measure</i>	9	1
	<b>Sum</b>	21	2
<b>Adjacent sentence overlap score</b>		$21 / 2 = 10.5$	

**Binary two-paragraph overlap.** Binary adjacent two-paragraph overlap calculates how many adjacent paragraphs include any overlapping items. The total index score is computed as the total number of paragraphs that include words that occur in the following two paragraphs divided by the number of paragraphs considered. Table X includes the calculation of binary adjacent two-paragraph overlap for the example text.

Table X

*Binary two-paragraph overlap*

Overlap check	Overlapping lemmas	Binary overlap Count	Number of paragraphs considered
LTP 1 -> LTP 2 and LTP 3	<i>analysis, can, be, very, it, important, to, understand, how, each, index, calculate</i>	1	1
LTP 2 -> LTP 3 and LTP 4	<i>be, that, index, it, also, to, understand, a, measure</i>	1	1
	<b>Sum</b>	2	2
<b>Adjacent sentence overlap score</b>		$2 / 2 = 1$	

**Connectives**

TAACO 1.5 calculates 23 connectives indices. For a particular connectives index list, the occurrence of each item in the list in the input text is counted. After checking the entire text for each list item, the sum of the list item is divided by the total number of words in the text. See Table X for an example. Each list of connectives is also included below.

Table X  
*Connectives example using the basic connectives list*

---

input text	Jack and Jill went up the hill to fetch a pail of water. They put the bucket in the well, but there was no water. Jack and Jill then decided to hike down the hill to a stream instead.
basic connectives	for, and, nor, but, or, yet, so
input text with basic connectives identified	Jack <b>and</b> Jill went up the hill to fetch a pail of water. They put the bucket in the well, <b>but</b> there was no water <b>so</b> Jack <b>and</b> Jill decided to hike down the hill to a stream instead.
number of basic connectives	4
number of total words	39
<b>basic connectives index score</b>	<b>4 / 39 = .1026</b>

---

**Basic connectives.** *for, and, nor, but, or, yet, so*

**Conjunctions.** *and, but*

**Disjunctions.** *or*

**Simple subordinators.** *after, although, as, because, before, if, once, since, that, though, till, unless, until, whenever, wherever, whereas, whereupon, while*

**Coordinating conjuncts.** *yet, so, nor, however, therefore*

**Addition.** *and, also, besides, further, furthermore, too, moreover, in addition, then, another, indeed, likewise*

**Sentence linking.** *nonetheless, therefore, although, furthermore, whereas, nevertheless, whatever, for, however, besides, henceforth, then, yet, if, while, so, but, until, because, alternatively, meanwhile, when, and, since, notwithstanding, whenever, moreover, as, with, consequently, after*

**Order.** *to begin with, next, first, firstly, second, secondly, finally, in conclusion, above all, before, after, then*

**Reason and purpose.** *therefore, that is why, for this reason, for that reason, hence, because, so, since, as, because of, on account of, so that, consequently*

**Opposition.** *but, however, nevertheless, otherwise, on the other hand, on the contrary, yet, still, maybe, perhaps, instead, except for, in spite of, despite, nonetheless, apart from, unlike, whereas*

**Determiners.** *a, an, the, this, that, these, those*

**Demonstratives.** *this, that, these, those*

**All additive connectives.** *after all, again, all in all, also, alternatively, and, anyhow, as a final point, as well, at least, besides, but, by contrast, by the way, contrasted with, correspondingly, except that, finally, first, for example, for instance, fortunately, further, furthermore, however, in actual fact, in addition, in contrast, in fact, in other words, in sum, incidentally, instead, it follows, moreover, next, notwithstanding that, on one hand, on the contrary, on the one hand, on the other hand, or, otherwise, rather, secondly, similarly, summarizing, summing up, that is, thereupon, to conclude, to return to, to sum up, to summarize, to take an example, to these ends, to this end, too, well at any rate, whereas, yet*

**All causal connectives.** *although, arise, arises, arising, arose, because, cause, caused, causes, causing, condition, conditions, consequence, consequences, consequent, consequently, due to, enable, enabled, enables, enabling, even then, follow that, follow the, follow this, followed that, followed the, followed this, following that, follows the, follows this, hence, made, make, makes, making, nevertheless, nonetheless, only if, provided that, result, results, since, so, therefore, though, thus, unless, whenever*

**Positive causal connectives.** *arise, arises, arising, arose, because, cause, caused, causes, causing, condition, conditions, consequence, consequences, consequent, consequently, due, enable, enabled, enables, enabling, even, follow, followed, following, follows, hence, if, made, make, makes, making, only, provided, result, results, since, so, that, the, then, therefore, this, thus, to*

**All logical connectives.** *actually, admittedly, after all, all in all, also, alternatively, although, and conversely, anyhow, anyway, arise from, arise out of, arises from, arises out of, arising from, arising out of, arose from, arose out of, as a final point, as a result, as well, at least, at this point, because, besides, but, by contrast, cause, caused, causes, causing, conditional upon, consequence, consequences, consequently, contrasted with, correspondingly, despite the fact that, due to, enable, enabled, enables, except that, finally, follow that, follow the, follow this, followed that, followed the, followed this, following that, follows the, follows this, for, fortunately, further, furthermore, hence, however, if, in actual fact, in any case, in any event, in case, in conclusion, in contrast, in fact, in order that, in other words, in short, in sum, incidentally, instead, it followed that, it follows, it follows that, likewise, moreover, nevertheless, next, nonetheless, nor, notwithstanding that, on condition that, on one hand, on the condition that, on the contrary, on the one hand, on the other hand, once again, or, otherwise, provided that, purpose of which, pursuant to, rather, secondly, similarly, since, so, summarizing, summing up, that is, that is to say, then, therefore, thereupon, though, thus, to conclude, to return to, to sum up, to summarize, to take an example, to that end, to these ends, to this end, to those ends, unless, well at any rate, whereas, while*

**Positive logical connectives.** *actually, after all, all in all, also, anyway, arise from, arise out of, arises from, arises out of, arising from, arising out of, arose from, arose out of, as a final point, as a result, as well, at least, at this point, because, besides, cause, caused, causes, causing, conditional upon, consequence, consequences, consequently, correspondingly, due to, enable, enabled, enables, finally, follow that, follow the, follow this, followed that, followed the, followed this, following that, follows the, follows this, for, fortunately, further, furthermore, hence, if, in actual fact, in any case, in any event, in case, in conclusion, in fact, in order that, in other words, in short, in sum, incidentally, instead, it followed that, it follows, it follows that, likewise, moreover, next, on condition that, on one hand, on the condition that, on the one hand, once again, provided that, purpose of which, pursuant to, secondly, similarly, since, so, summarizing, summing up, that is, that is to say, then, therefore, thereupon, thus, to conclude, to return to, to sum up, to summarize, to take an example, to that end, to these ends, to this end, to those ends, well at any rate, while*

**Negative logical connectives.** *admittedly, alternatively, although, and conversely, anyhow, but, by contrast, contrasted with, despite the fact that, except that, however, in contrast, nevertheless, nonetheless, nor, notwithstanding that, on the contrary, on the other hand, or else, otherwise, rather, though, unless, whereas, yet*

**Temporal connectives.** *a consequence of, after, again, all this time, as, at last, at once, at the same time, at this moment, at this point, before, by this time, earlier, finally, first, follow that, following that, from now on, further, immediately, in the meantime, instantly, it followed that, it follows, it follows that, later, meanwhile, next, now that, on another occasion, once more, presently, previously, secondly, simultaneously, since, so far, soon, suddenly, the consequence of, the consequences of, the last time, the previous moment, then, this time, throughout, to that end, until, up till that time, up to now, when, whenever, while*

**Positive intentional connectives.** *by, desire, desired, desires, desiring, for, goal, goals, in order, made, make, makes, making, purpose, purposes, so, to that end, to these ends, to this end, to those ends, want, wanted, wanting, wants*

**All positive connectives.** *actually, after, again, all in all, also, and, anyway, arise, arises, arising, arose, as, at last, at least, at once, because, before, besides, by, cause, caused, causes, causing, condition, conditional upon, conditions, consequence, consequences, consequent, consequently, correspondingly, desire, desired, desires, desiring, due, earlier, enable, enabled, enables, enabling, even, finally, first, follow, followed, following, follows, for, fortunately, from now on, further, furthermore, goal, goals, hence, if, immediately, in actual fact, in addition, in any case, in any event, in case, in conclusion, in fact, in order, in other words, in short, in sum, incidentally, instantly, instead, later, likewise, made, make, makes, making, meanwhile, moreover, next, on another occasion, on one hand, once more, only, provided, presently, previously, purpose, purposes, result, results, secondly, similarly, simultaneously, since, so, soon, suddenly, summarizing, summing up, that, the, then, therefore, thereupon, this, throughout, thus, to, too, want, wanted, wanting, wants, well at any rate, when, whenever, while*

**All negative connectives.** *admittedly, alternatively, although, and conversely, anyhow, but, by contrast, contrasted with, despite the fact that, except that, however, in contrast,*

*nevertheless, nonetheless, nor, notwithstanding that, on the contrary, on the other hand, or, otherwise, rather, though, unless, until, whenever, whereas, yet*

**All connectives.** *actually, admittedly, after, again, all in all, all this time, also, alternatively, although, and, anyhow, anyway, arise, arises, arising, arose, as, at last, at least, at once, at the same time, at this moment, at this point, because, before, besides, but, by, cause, caused, causes, causing, condition, conditional upon, conditions, consequence, consequences, consequent, consequently, contrasted with, correspondingly, desire, desired, desires, desiring, despite the fact that, due to, earlier, enable, enabled, enables, enabling, except that, finally, first, follow that, follow the, follow this, followed that, followed the, followed this, following that, follows the, follows this, for, fortunately, from now on, further, furthermore, goal, goals, hence, however, if, immediately, in actual fact, in addition, in any case, in any event, in case, in conclusion, in contrast, in fact, in order, in other words, in short, in sum, in the end, in the meantime, incidentally, instantly, instead, it followed that, it follows, it follows that, later, likewise, made, make, makes, making, meanwhile, moreover, nevertheless, next, nonetheless, nor, notwithstanding that, now that, on another occasion, on one hand, on the contrary, on the one hand, on the other hand, once more, or, otherwise, presently, previously, provided that, purpose of which, pursuant to, rather, secondly, similarly, simultaneously, since, so, soon, suddenly, summarizing, summing up, that is, the last time, the previous moment, then, therefore, thereupon, this time, though, throughout, thus, to conclude, to return to, to sum up, to summarize, to take an example, to that end, to these ends, to this end, to those ends, too, unless, until, up till that time, up to now, well at any rate, when, whenever, whereas, while, yet*

## **How to Conduct a Study with TAACO**

This section comprises an overview of how to conduct a study using TAACO. This is not meant to be comprehensive, but it should provide researchers an introduction into text analysis using automated tools.

### **1. Determine Research Questions**

First and foremost, an automatic analysis tool such as TAACO is only useful if it helps a researcher address specific research questions. Therefore, having at least one concrete research question is vital to conducting meaningful research using a text analysis tool.

### **2. Learn About Indices**

Before processing data with a text processing tool, it is important to determine what indices the tool measures and whether the calculated indices meets the needs of the researcher. For TAACO 1.5, this information can be found in the TAACO 1.5 Indices section above and the TAACO 1.5 Index Description Spreadsheet, which is available at [www.kristopherkyle.com](http://www.kristopherkyle.com).

### **3. Choose Indices**

After learning about the indices, a researcher should determine which indices will be informative for investigating the research questions. Avoiding the use of indices that are not clearly related to one's research questions will help make interpreting results more straightforward.

### **4. Format Input Texts**

Before running a text analysis program, it is important to ensure that the texts to be analyzed are in the appropriate format (see the Input Text Formatting section above) and are largely free from spelling and punctuation errors. The algorithms that detect sentence boundaries, for example, will not work correctly if there is no white space (e.g., a space or new line character) between sentence ending punctuation and the beginning of the new sentence. For example, TAACO will count "This is a sentence.This is a second sentence." as a single sentence, and will also count "sentence.This" as a single word.

### **5. Process Data**

After completing the steps above, data should be processed using TAACO 1.5's interface.

### **6. Check Data Processing Results**

TAACO 1.5 provides two types of diagnostic output to help users ensure that their data was processed accurately. See the "Options" section above for more details. Before beginning a statistical analysis, it is important to spot check the data to ensure that a) words, sentences and paragraphs are being counted accurately and b) the lemmatizer and part of speech tagger is producing accurate results. Both of these can be checked using the diagnostic output features. If major problems are discovered with regard to a) or b), the likely problem is the format of the input texts.

### **7. Conduct Preliminary Statistical Tests to Meet Assumptions**

With any statistical analysis, it is important for researchers to ensure that their data meets the assumptions of the statistical test(s) that are chosen. In many cases, analyses with TAACO



indices involve statistical tests such as multiple regression, discriminant function analysis, and factor analysis. In addition to other tests of assumptions, it is particularly important to check for normality and multicollinearity. Some TAACO features may be particularly rare (or conversely particularly common) in certain datasets, leading to heavily skewed distributions. Furthermore, some TAACO indices measure the same construct in slightly different ways, which may result in a number of strongly correlated indices.

## **8. Interpret and Report Results**

After conducting preliminary assumptions tests and (provided those assumptions are met) primary statistical tests are conducted, it is time to interpret the results. Interpreting the results should be (relatively) easy provided the researcher followed steps 1, 2, and 3. Researchers who neglect to follow these steps may find themselves having difficulty understanding and interpreting their results.