Empowering ELAN with N-gram Analytics for Corpora

Larwan Berke
Gallaudet University
800 Florida Ave. NE
Washington, DC
larwan.berke@gallaudet.edu

ABSTRACT

American Sign Language, the preferred language of the Deaf community in the USA is its own language; complete with a rich collection of grammatical features. The DePaul University team has been working on an automatic English/ASL translator implemented as a 3D avatar in order to facilitate better communication between hearing and deaf people. Animations suffered from various timing inconsistencies and were awkward in appearance. This project attempts to address them by using corpus analysis to discern subtle features of ASL in order to improve the coarticulation model in the avatar.

1. INTRODUCTION

American Sign Language (ASL), the preferred language for members of the Deaf community in the United States of America, uses hand and body movement combined with facial expressions in order to communicate. ASL is its own distinct language, neither a gestural expression of English nor a derivative [1].

Due to the differences between ASL and English, a communication barrier has always existed between deaf and hearing communities even for short, spontaneous interactions. In order to facilitate communication and to overcome this barrier, the ASL research project at DePaul University [2] has been developing an automatic English to ASL translator named Paula that is displayed as a 3D avatar. A sample output of Paula interpreting a phrase is shown (Figure 1). Creating believable animations that are natural in appearance requires access to corpus data.

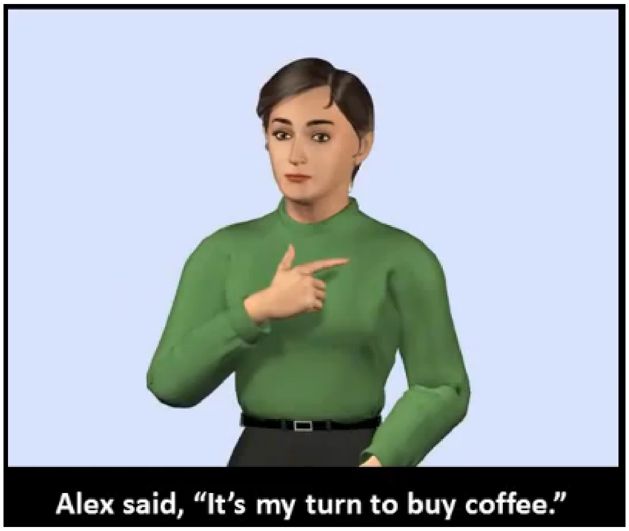
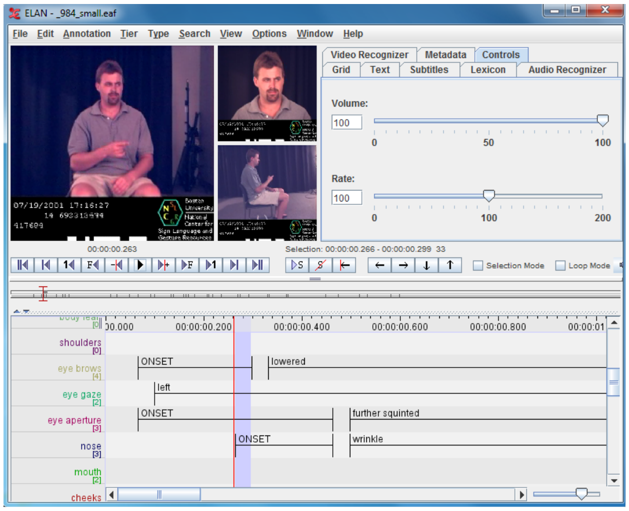


Figure 1. Paula from the demo website.

The team obtained a copy of the corpus published by the ASL Linguistic Research Project at Boston University [3]. The corpus data consists of videos and manually-encoded annotations that are synchronized with the video. Common annotations include the manual glosses (English equivalent of an ASL sign), nonmanual signals, and an English translation.

Originally, the corpus was in the SignStream format [4], but a custom software package constructed on a SignStream API [5] converted it to ELAN's XML-based format [6] to take advantage of the better user interface and search capabilities offered by ELAN.

ELAN is the “Eudico Linguistic ANnotator”, an open-source program created by the “EUropean DIstributed COrpora” project [7]. ELAN is widely used in linguistic studies as the tool of choice to annotate media clips containing utterances of the language under study. A graphical interface enables linguists to view the clips and annotate them at the same time, speeding up productivity. An example of the GUI is provided in (Figure 2). The annotations are organized under specific tiers that the user can define or import as part of a template. Present in the tiers are annotations arranged with respect to time. The annotated corpus with ELAN contained a wealth of information that the team wanted to analyze.


ELAN - _984_small.eaf

Volume: 100

Rate: 100

00:00:00.263

Selection: 00:00:00.266 - 00:00:00.299 33

Figure 2. Example of the ELAN GUI.

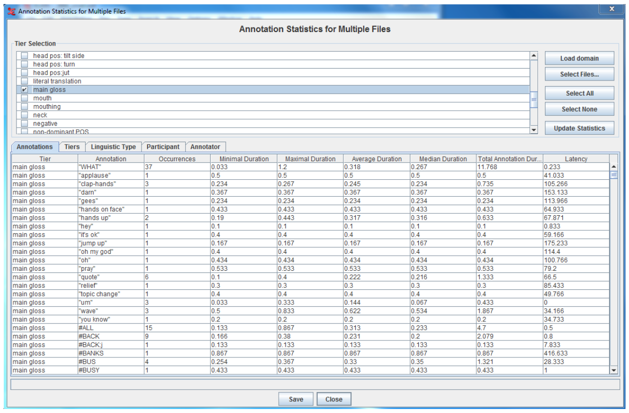
2. STUDYING COARTICULATION

What the team desired was to better understand the relationship between glosses in the corpus to further their research on coarticulating signs. Coarticulation is the manner in which the characteristics of signs performed in sequence are modified by the preceding and succeeding signs. In ASL signs are characterized by their location, orientation, handshape, movement, and nonmanual signals [8].

In a signing avatar, modeling coarticulation and automatically incorporating it into generated utterances of ASL is necessary to create realistic movement. Without coarticulation the transitions between signs tend to be stiff and awkward. By incorporating a model of coarticulation we hope to improve the acceptability of the animation, and possibly the clarity of the communication. To achieve this goal, the avatar needed a database of coarticulated signs in order to fine-tune its timing.

3. EXISTING TOOLS

ELAN contains the ability to generate some useful statistics on the corpus as a whole via the "Annotation Statistics for Multiple Files" command. An example of what it looks like is provided in (Figure 3). It was able to collate information on the glosses over the corpus and provide some descriptive statistics like the mean and median duration.



Annotation Statistics for Multiple Files

Tier Selection

☐ head pos: tilt side
☐ head pos: turn
☐ head pos:jut
☐ literal translation
☑ main gloss
☐ mouth
☐ mouthing
☐ neck
☐ negative
☐ non-dominant POS

Tier	Annotation	Occurrences	Minimal Duration	Maximal Duration	Average Duration	Median Duration	Total Annotation Dur...	Latency
main gloss	"WHAT"	37	0.033	1.2	0.318	0.267	11.768	0.233
main gloss	"applause"	1	0.5	0.5	0.5	0.5	0.5	41.033
main gloss	"clap-hands"	3	0.234	0.267	0.245	0.234	0.735	105.266
main gloss	"darn"	1	0.367	0.367	0.367	0.367	0.367	153.133
main gloss	"gees"	1	0.234	0.234	0.234	0.234	0.234	113.966
main gloss	"hands on face"	1	0.433	0.433	0.433	0.433	0.433	64.933
main gloss	"hands up"	2	0.19	0.443	0.317	0.316	0.633	67.871
main gloss	"hey"	1	0.1	0.1	0.1	0.1	0.1	0.833
main gloss	"it's ok"	1	0.4	0.4	0.4	0.4	0.4	59.166
main gloss	"jump up"	1	0.167	0.167	0.167	0.167	0.167	175.233
main gloss	"oh my god"	1	0.4	0.4	0.4	0.4	0.4	114.4
main gloss	"oh"	1	0.434	0.434	0.434	0.434	0.434	100.766
main gloss	"pray"	1	0.533	0.533	0.533	0.533	0.533	79.2
main gloss	"quote"	6	0.1	0.4	0.222	0.216	1.333	66.5
main gloss	"relief"	1	0.3	0.3	0.3	0.3	0.3	85.433
main gloss	"topic change"	1	0.4	0.4	0.4	0.4	0.4	49.766
main gloss	"um"	3	0.033	0.333	0.144	0.067	0.433	0
main gloss	"wave"	3	0.5	0.833	0.622	0.534	1.867	34.166
main gloss	"you know"	1	0.2	0.2	0.2	0.2	0.2	34.733
main gloss	#ALL	15	0.133	0.867	0.313	0.233	4.7	0.5
main gloss	#BACK	9	0.166	0.38	0.231	0.2	2.079	0.8
main gloss	#BACK:j	1	0.133	0.133	0.133	0.133	0.133	7.833
main gloss	#BANKS	1	0.867	0.867	0.867	0.867	0.867	416.633
main gloss	#BUS	4	0.254	0.367	0.33	0.35	1.321	28.333
main gloss	#BUSY	1	0.433	0.433	0.433	0.433	0.433	1

Figure 3. ELAN Annotation Statistics for Multiple Files.

One useful tool the team already had was the ability to generate histograms of a gloss' length [9]. A sample histogram is shown (Figure 4).

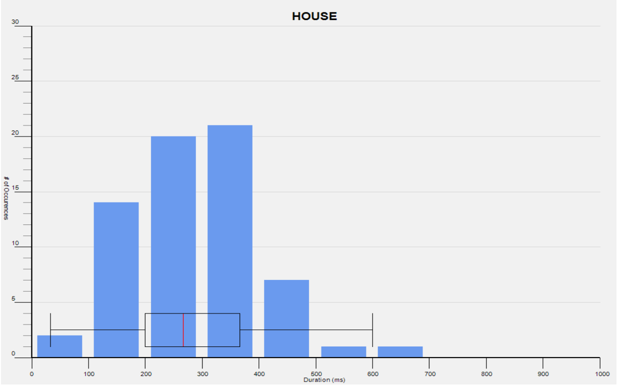


Figure 4. Histogram of the gloss HOUSE with the x-axis representing duration in milliseconds and the y-axis representing occurrences in the corpus.

Other tools outside the realm of ELAN exist to analyze corpus in varying annotation formats are available. Some of them are: iLex [10], Anvil [11], SignStream [4], and ATLAS [12]. All of them are well-utilized software with the ability to analyze corpus files. However answering questions regarding coarticulation created a challenge. Existing tools was sufficient to study individual glosses in the entire corpus and the information assisted in the team's effort to normalize the signs being produced in the avatar. A new approach was required to go further than analyzing just the gloss.

4. CORPUS ANALYSIS APPROACHES

It was necessary to develop alternative tools for this purpose and existing corpus analysis literature outlined several approaches that were possible. Most of the published papers focused solely on text-based analysis and did not address sign language corpus. Others made superficial analyses and only used the data to drive their avatar models which the team already incorporated in Paula.

One popular approach was to focus on the frequencies of individual glosses and discover correlations in parallel corpora. An extensive study was done by Johnston [13] that addressed this method. Other authors have done the same thing to corpora in different languages and their papers outlined methods similar to what Johnston employed. However, gloss frequencies revealed very little on coarticulation behaviors and were similar to the histograms the team already generated.

Other authors have preferred to use a completely different system to annotating sign languages: SiGML [14] and HamNoSys [15]. The software associated with their systems can execute corpus analyses similar to those available in ELAN. However, this would require writing an additional conversion package, so it was better to develop software that was usable on the existing ELAN-based corpus.

In related work on machine translation, Zhao et al. [16] broke the corpus down into parse trees. It was then possible to discern relationships in the corpus data. However, like many other authors who worked on machine translation it was mostly focused on analyzing the English corpus, not the ASL corpus.

Extensive studies have been conducted worldwide on sign language recognition systems which employed automatic machine identification algorithms. That approach was surveyed by Ong & Ranganath [17] and showed that there was still a lot of work to be done in order to achieve satisfactory performance. Delving into the different systems gave us insights on what algorithms were used. However, most of them are related to video recognition and contained no algorithms we could use to analyze our corpus.

As a survey of the literature showed, there existed very few tools that were able to analyze the ELAN-based ASL corpus to provide sufficient detail to develop new coarticulation models. The task fell on the team to utilize a previously unexplored approach to corpus analysis.

5. DISCOVERING N-GRAMS

The concept of bigrams is often used to support Markov models [18] or associated models for prediction and machine translation. Immediately the team realized that bigrams would be useful for discovering insights into patterns in coarticulation.

One influential paper in linguistic analysis related to bigrams described work by Dunning [19]. It is an approach not utilized in sign language studies and from there we became aware of how powerful contingency tables were in processing corpora. It was possible to extract voluminous amounts of data from a single table as Pecina [20] showed in the survey of collocation methods. The list is shown in (Appendix 1) and contains 84 different formulas. The team was now prepared to apply N-gram analytics on the ELAN corpus. However, with a huge list of methods it dawned on us that we had to find relevant formulas for our research.

6. NGRAM STATISTICS PACKAGE

A resource that proved immensely valuable in this regard was the Ngram Statistics Package (NSP), a Perl program used to process corpora for varying statistics [21]. The developers had carefully chosen several algorithms and implemented them in the program. It made our search scope much narrower as we just had to study the algorithms they selected and understand the basis of their applications. The NSP contained 13 formulas for analyzing bigrams and we immediately saw the value of some of them in answering our questions. Utilizing the precise formulas outlined in NSP, it was now possible to directly compare corpora and immediately see the differences. Furthermore, the set of data analysis tools would help us gain a better understanding of the behaviors of coarticulation.

The next step was to understand the algorithms and to apply them to our corpus. We set out to implement the N-gram analytics capabilities in ELAN. It was decided that augmenting ELAN was the best course of action as we could publish the new features and benefit the sign language linguistics community with enhanced capabilities. It was straightforward to study the code contained in NSP and the relevant publications and port them over to ELAN's Java codebase.

7. ELAN's N-GRAM ANALYTICS

Implementing the N-gram capabilities in ELAN was relatively easier than wading through the dense mathematical publications outlining the contingency table algorithms. Fortunately, reviewing literature from linguistic authors helped us understand the applications of the algorithms. What follows is an outline of the new capability in ELAN and an extensive treatment of the resulting data. It is the aim of this paper to serve as a documentation of the new functionality and to help users understand how to use the data.

The N-gram analysis can be reached through the “Multiple File Processing” submenu through the File menu. From there the “N-gram Analysis” method will be available to start the analysis. A picture showing the location of the new functionality is in (Figure 5).

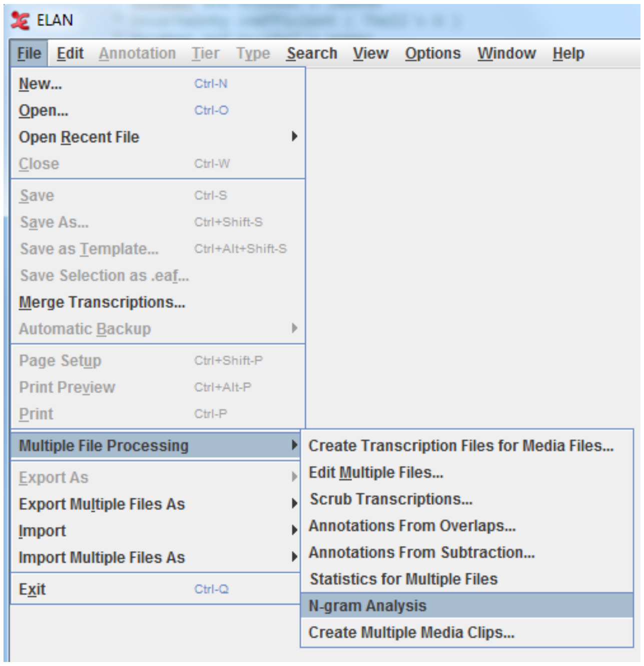


Figure 5. Location of the N-gram analysis in ELAN.

A new dialog window will pop up that contains the various options for the search and the resulting table showcasing a few statistics. It looks like this (Figure 6).

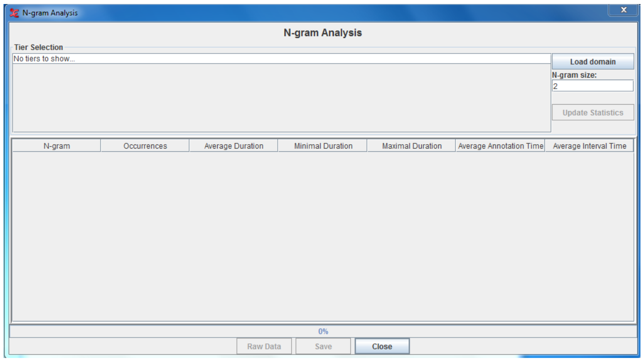
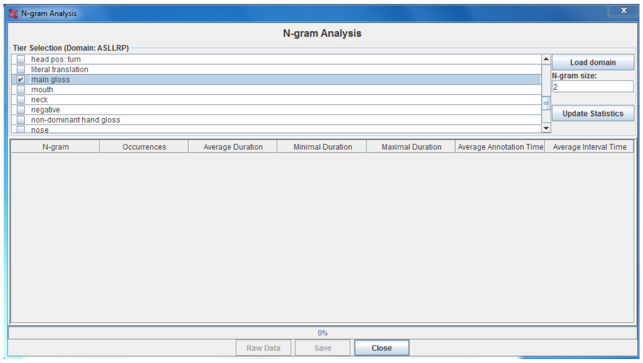


Figure 6. Main N-gram analysis window.

The first step is to select the search domain. It is the standard ELAN “Load domain” window where the user can then specify a list of files or directories to search in. Once that is done a list of tiers seen in the domain will be shown (Figure 7). A note of caution: the code assumes that all files in the domain will contain the same tiers. It then loads the first file in the domain to extract the tiers and display it in the window.



N-gram Analysis

Tier Selection (Domain: ASLLRP)

- □ head pos: turn
- □ literal translation
- ☑ main gloss
- □ mouth
- □ neck
- □ negative
- □ non-dominant hand gloss
- □ nose

N-gram size: 2

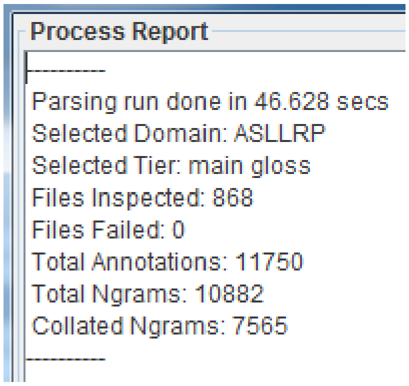
N-gram	Occurrences	Average Duration	Minimal Duration	Maximal Duration	Average Annotation Time	Average Interval Time

0%

Figure 7. N-gram analysis window displaying possible tiers to search on.

The next step is to then define the N-gram size in the textbox. The software can handle any positive size greater than 1. However, the powerful contingency table analysis can only be done on bigrams and will not be done on unigrams or trigrams and bigger N-grams. Once that is defined clicking the "Update Statistics" button will start the search and depending on the machine, take a while to calculate the statistics. The annotations are extracted from the files, N-grams created from them, and finally collated into groupings of same N-grams for statistical analysis.

When the search is done a report window will pop up displaying some information. If there were errors during the search it will be displayed in this window so it is important to double-check the validity of the search. A sample report is seen in (Figure 8).

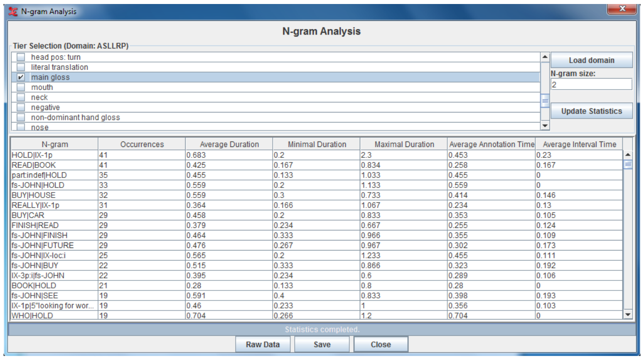


Process Report

```
----------
Parsing run done in 46.628 secs
Selected Domain: ASLLRP
Selected Tier: main gloss
Files Inspected: 868
Files Failed: 0
Total Annotations: 11750
Total Ngrams: 10882
Collated Ngrams: 7565
----------
```

Figure 8. N-gram analysis report window.

When the search is done, the result table will be displayed in the main window as shown in (Figure 9). Some of the columns from the data are visible here, however only a small subset is displayed simultaneously to avoid overcrowding the GUI. The visible columns are: N-gram, Occurrences, Average Duration, Minimal Duration, Maximal Duration, Average Annotation Time, and Average Interval Time.



N-gram Analysis

Tier Selection (Domain: ASLLRP)

- □ head pos: turn
- □ literal translation
- ☑ main gloss
- □ mouth
- □ neck
- □ negative
- □ non-dominant hand gloss
- □ nose

N-gram size: 2

N-gram	Occurrences	Average Duration	Minimal Duration	Maximal Duration	Average Annotation Time	Average Interval Time
HOLD|IX-1p	41	0.683	0.2	2.3	0.453	0.23
READ|BOOK	41	0.425	0.167	0.834	0.258	0.167
part:indef|HOLD	35	0.455	0.133	1.033	0.455	0
fs-JOHN|HOLD	33	0.559	0.2	1.133	0.559	0
BUY|HOUSE	32	0.559	0.3	0.733	0.414	0.146
REALLY|IX-1p	31	0.364	0.166	1.067	0.234	0.13
BUY|CAR	29	0.458	0.2	0.833	0.353	0.105
FINISH|READ	29	0.379	0.234	0.667	0.255	0.124
fs-JOHN|FINISH	29	0.464	0.333	0.966	0.355	0.109
fs-JOHN|FUTURE	29	0.476	0.267	0.967	0.302	0.173
fs-JOHN|IX-loc:i	25	0.565	0.2	1.233	0.455	0.111
fs-JOHN|BUY	22	0.515	0.333	0.866	0.323	0.192
IX-3p:i|fs-JOHN	22	0.395	0.234	0.6	0.289	0.106
BOOK|HOLD	21	0.28	0.133	0.8	0.28	0
fs-JOHN|SEE	19	0.591	0.4	0.833	0.398	0.193
IX-1p|5"looking for wor...	19	0.46	0.233	1	0.356	0.103
WHO|HOLD	19	0.704	0.266	1.2	0.704	0

Statistics completed.

Figure 9. The result of the N-gram analysis.

It is obvious that the displayed columns closely mirror the data shown in ELAN's existing statistical output as shown in (Figure 3). It is important to note how the N-grams are displayed in (Figure 9), and the first row contains the N-gram "HOLD|IX-1p". The vertical marker "|" separates the annotations contained in the bigram. For example, if a trigram was selected it would show something similar to "FINISH|READ|BOOK" and so on for larger N-gram sizes.

Finally, in order to see the entire data that was produced it is necessary to export the results into a text file for further processing. This is done by clicking on the "Save" button and a dialog will pop up asking the user where to save the data. It is exported in a CSV-like format (Comma-Separated Values). Based on the existing statistics code in ELAN, the CSV file uses tabs "\t" as the delimiters and newlines "\n" as the record separators to avoid ambiguity with the values. A sample row is: "HOLD|IX-1p\t7.9934\t0.348\t0.13754 ..." and contains numerous columns. A full listing of a row for bigrams can be seen in (Appendix 2).

It is then possible to import that data into the user's preferred spreadsheet program to further analyze the result. For example, using Microsoft Excel or LibreOffice Calc it is possible to open the file and by specifying tabs as the delimiter then letting it import the data. There might be issues with the text qualifier as the N-gram could contain quotes in the corpora so be sure to watch out for that and disable it if necessary. The first few rows in the file contain the parameters of the search as documentation of the search query that generated the data. The names of the columns appear after the header, followed by the rows of N-gram data. If the search was a bigram, there should be 69 columns of information, 56 otherwise as N-grams other than bigrams will not include contingency table metrics. More metrics are continually added as the team discovers useful formulas so it could be more than the published number. Please consult (Section 8) for an in-depth treatment of the exported statistical data.

Furthermore, it is possible to export the N-grams individually in order to process it separately from ELAN. All the analytics done in (Section 8) is based on this raw data. The data is exported by clicking the "Raw Data" button in the GUI. After supplying the file the data will be exported in the same CSV format as discussed above. The data is formatted in a similar way as discussed above but only 11 columns are present if the N-gram size is bigger than one, 9 otherwise as the annotation/interval timing is not applicable to unigrams. Please consult (Section 9) for an in-depth treatment of the exported raw data.

8. DATA FROM N-GRAM ANALYSIS

The following sections will delve into the resulting data from the analysis and explain how they are generated and further elucidate on how the user can utilize the data.

There are three categories of data that are produced by the analysis: general, metric-descriptive statistics, and contingency table-related statistics. The general data is an overview of the N-gram, providing some useful metrics. The metric-descriptive delves into one metric of the N-gram with a whole host of descriptive statistics like the mean, median, mode, and other statistical measures. Finally, the contingency table-related stats contain the results from various algorithms executed on the table per N-gram. The N-grams being analyzed in this section are generated by collating the raw results so that all occurrences of a given N-gram occur as a contiguous group. This allows us to generate metrics on a N-gram across the entire corpus and further analysis can be done by the user utilizing the raw data in (Section 9).

8.1 General Data

The CSV file contains a header block before listing the rows. A sample header is shown (Figure 10). If desired, it is possible to extract the generic data to use for further calculations.

```
# Export of N-gram Analysis done on Wed Aug 14 18:40:20 CDT 2013
# Selected Domain: ASLLRP
# Selected Tier: main gloss
# N-gram Size: 2
# Search Time: 61.459s
# Files Inspected: 868
# Total Annotations: 11750
# Total N-grams: 10882
# Total Collated N-grams: 7565
```

Figure 10. A sample header block in the CSV file.

The structure of the N-gram is shown in (Figure 11) and from that most of the metrics are calculated. The figure shows a bigram, but the same concept applies to all N-grams with a size greater than one. For a unigram some metrics become meaningless like "Interval Time" so take that into consideration when analyzing the resulting data.

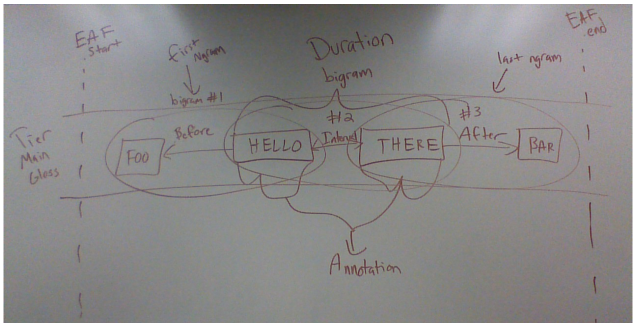


Figure 11. The structure of a bigram.

8.1.1 N-gram

This column is self-explanatory and contains the name of the N-gram. It is constructed from the annotations delimited by a vertical divider "|". The number of annotations is determined by the N-gram size set during the search.

8.1.2 File Occurrences

This column counts the number of files in the search domain that contained the N-gram. The N-gram can appear multiple times in a file and the file will only be counted once.

8.1.3 First N-gram

This column counts the number of times this N-gram appeared in the front of the file. It might not be that useful if the corpora contains files that contain lengthy discourse. However, if a file contains single sentences, then this is an excellent indicator of the "position" of the N-gram relative to the utterance. For example, dividing this value by Occurrences (8.1.5) will give the percent of times the N-gram was at the start of the utterance. For a deeper analysis, refer to N-gram Position (8.2.1.5). Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.1.4 Last N-gram

This column counts the number of times this N-gram appeared in the end of the file. This is analogous to the first N-gram (8.1.3).

8.1.5 Occurrences

This column counts the number of times this N-gram was seen in the corpora.

8.2 Metric-Descriptive

This class of statistics displays a single metric with several descriptive statistics applied to it. For example, the duration of a N-gram is not reported as a general data because it varies within the collation. It is necessary to apply techniques such as mean, median, standard deviation, and others to visualize the spread of the metric. A list of the metrics are given, then the list of descriptive statistics that is applied to each metric.

8.2.1 List of Metrics

Those values contain the observations in the N-gram, and will have the descriptive statistics applied to them. The columns are named by joining the metric with the descriptive statistic by a vertical marker "|" as in "Duration|Mean", "Duration|StdDev" and so on.

8.2.1.1 After Interval

The time in seconds between this N-gram to the next N-gram in the file. If this is the last N-gram in the file, it will be skipped when calculating the collated statistics. To get the count of skipped N-grams, reference Last N-gram (8.1.4). Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.1.2 Before Interval

The time in seconds between this N-gram to the previous N-gram in the file. If this is the first N-gram in the file, it will be skipped when calculating the collated statistics. To get the count of skipped N-grams, reference First N-gram (8.1.3). Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.1.3 Duration

The time in seconds for this N-gram. Contains the annotation time and the interval between annotations. Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.1.4 Latency

The time in seconds in the file for this N-gram to appear. Same concept as the "Starting Time" of the first annotation in the N-gram.

8.2.1.5 N-gram Position

The position of this N-gram in the file. Starts at 1 for the first N-gram. Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.1.6 Total Annotation Time

The time in seconds for the annotations in this N-gram. Does not count the interval between annotation. Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.1.7 Total Interval Time

The time in seconds of intervals between annotations in this N-gram. Does not count the annotation time. Consult (Figure 11) for a diagram illustrating how the annotations and N-grams relate to each other.

8.2.2 List of Descriptive Statistics

Those descriptive statistics are applied to the metrics listed above. If a NaN (Not a Number) appears as a value in a cell, it means there are insufficient data to do the calculation (empty list) or a divide by 0 occurred.

8.2.2.1 Max

The maximum value in the list.

8.2.2.2 Min

The minimum value in the list.

8.2.2.3 Mode

The value that occurs the most often in the list. If a multimodal list (several values having the same frequency) occurs, then the highest value is picked.

8.2.2.4 Mean

The arithmetic mean of the list, also known as the average.

8.2.2.5 Quartile1

The 1st quartile in the list. Calculated by finding the median of the lower 50th percentile. Uses the unbiased algorithm by including the median in the percentile and solves ties by averaging the values. Returns NaN if the list contains less than 3 values.

8.2.2.6 Median

The center value of the list, also known as the 2nd quartile or the 50th percentile. Ties are solved by finding the mean of the values.

8.2.2.7 Quartile3

The 3rd quartile in the list. Calculated by finding the median of the upper 50th percentile. Uses the unbiased algorithm by including the median in the percentile and solves ties by averaging the values. Returns NaN if the list contains less than 3 values.

8.2.2.8 Variance

The measure of the spread of the values in the list, also known as the 2nd central moment. Uses the unbiased algorithm where the sum of squared mean deviations is divided by the number of elements in the list minus 1. Returns NaN if the list contains a single element.

8.2.2.9 StdDev

The standard deviation of the list. Uses the unbiased algorithm by taking the square root of the unbiased variance in (8.2.1.8).

8.2.2.10 Skewness

A measure of symmetry in the list, also known as the normalized 3rd central moment. A positive value indicates a longer right tail. A negative value indicates a longer left tail.

8.2.2.11 Kurtosis

A measure of the peakness in the list, also known as the normalized 4th central moment. A positive value indicates a leptokurtic peak with thin tails. A negative value indicates a platykurtic peak with thick tails.

8.3 Contingency Table Statistics

It is important to start this section with a description on how the tables are generated for the bigrams. The analysis in the following sections can only be done on bigrams, the software will not output these results if a different-sized N-gram is selected for the search.

Dunning [19] described the format of a contingency table used to locate collocations in the corpora (Table 1). With given annotations A and B, the table contained the counts of bigrams (A|B) found in the corpus. The notation ~A means an annotation that is not A, and the same applies for ~B.

Table 1. The contingency table proposed by Dunning.

count(A B)	count(~A B)
count(A ~B)	count(~A ~B)

Dunning reasoned that the layout of the table allowed us to propose hypotheses about the relationship between the annotations (words) in the paper:

If the words A and B occur independently, then we would expect p(AB) = p(A)p(B) where p(AB) is the probability of A and B occurring in sequence, p(A) is the probability of A appearing in the first position, and p(B) is the probability of B appearing in the second position. [19, p70]

Using the power of distributive statistics, it is now possible to apply a wide range of formulas on the table to see how bigrams rank and compare to each other. Analyzing a corpus and discovering hidden relationships is essential to advancing the team's coarticulation research. Furthermore, it is now possible to execute parallel corpora analysis by comparing the metrics from one corpus to another. What follows are the list of formulas that was implemented in ELAN. In the CSV file, those metrics are prefixed with "cT|" and sample columns are "cT|Chi-squared", "cT|T-score", and so on.

It is important to note that there is no definitive answer as to which contingency table formula is the best. There are too many competing algorithms and each of them has a specific application. It is up to the user to study the literature and determine which approach best fits their corpus and hypotheses. By providing as many methods as possible it is our hope that researchers will find one of them useful for their project. Otherwise we welcome them to contribute to the ELAN project by adding more algorithms and benefit everyone.

8.3.1 Chi-squared

The well-known Pearson's Chi-squared test of goodness of fit [22]. It is a standard statistic that attempts to model the relationship between the standard deviation of the observed frequencies against the theoretical frequencies.

8.3.2 Dice Coefficient

This approach was proposed by Smadja et al. [23] as a suitable tool to execute parallel corpora comparisons for bigrams.

8.3.3 Fisher Exact Left Sided

This approach was proposed by Pedersen [24] to identify dependent bigrams. Comparing Fisher's tests to Chi-squared and other tests, Pedersen found that Fisher was the best way to rank bigrams and was a useful tool to extract the most relevant relationships. This is the left-sided variant of the test, the one most commonly used.

8.3.4 Fisher Exact Right Sided

The same algorithm as used in Left Sided (8.3.3) but looking at the right side of the tail.

8.3.5 Fisher Exact Two Tailed

The same algorithm as used in Left Sided (8.3.3) but looking at both sides of the tail.

8.3.6 Jaccard Coefficient

This approach was proposed by Chung & Lee [25] as an alternative method to cluster dependent bigrams. The authors concluded that Jaccard was good at emphasizing high frequency terms. This value is computed from the Dice Coefficient (8.3.2) utilizing the transformation formula: dice / (2 – dice).

8.3.7 Log-likelihood

This approach was proposed by Moore [26] to identify rare bigrams. Moore used the observation proposed in Zipf's law [27] to formulate the argument that most corpora will have a distribution of words that is unsuitable for processing with other methods.

8.3.8 Odds Ratio

This metric was utilized in Szmrecsanyi [28] while studying the persistence of associations in corpus. The same formula was used to great effect in Blaheta & Johnson's machine learning system [29]. The calculation of this formula smoothed out zeroes in the denominator by converting them to one before dividing.

8.3.9 Phi Coefficient

This is the mean square contingency coefficient proposed by Pearson. It was useful in identifying concordances in parallel corpora by Church & Gale [30]. A note of caution: this is the Phi^2 coefficient, not the Phi as proposed by Pearson. The reason for this is that Church & Gale used Phi^2 in their paper and if the Phi value is needed, it can be calculated by the user taking the square root of the given Phi^2 value.

8.3.10 Pointwise Mutual Information

This statistic was proposed by Church & Hanks [31] to find associations between words. It was useful in their research to discover associations that was relevant and is based on Palermo & Jenkins' work [32].

8.3.11 Poisson-Stirling Measure

This statistic was proposed by Quasthoff & Wolff [33] as an alternative method to find collocations in corpora. It is based on the mathematical poisson distribution and was a good match for their corpus.

8.3.12 T-score

The applications of the standard t-test to contingency tables was analyzed in depth by Church, et al. [34]. They discovered that the t-score was a good metric to use in their parse tree algorithms.

8.3.13 True Mutual Information

This method was proposed by Lin [35] to identify non-compositional phrases to great effect. It is a modified variant of the PMI formula (8.3.10).

9. RAW DATA FROM N-GRAM ANALYSIS

The following section will discuss the raw data generated from searching the corpus. The raw data lists each individual N-gram and its associated metrics. Utilizing this data might be desirable in situations where the user needs to double-check the analytics done in ELAN (Section 8) or to calculate other algorithms.

9.1 General Data

The raw data CSV file will have a header similar to the statistical data as shown in (Figure 10). After the header, the rows of N-gram data are shown with their metrics. What follows is an explanation of the columns in the data.

9.1.1 N-gram

The name of the N-gram, same as in (8.1.1).

9.1.2 After Interval

The time in seconds between this N-gram to the next N-gram in the file. Similar concept as in (8.2.1.1) but returns NaN if it is the last N-gram in the file.

9.1.3 Before Interval

The time in seconds between this N-gram to the previous N-gram in the file. Similar concept as in (8.2.1.2) but returns NaN if it is the first N-gram in the file.

9.1.4 Duration

The time in seconds for this N-gram. Contains the annotation time and the interval between annotations. Similar concept as in (8.2.1.3).

9.1.5 File

The full path to the EAF file containing this N-gram.

9.1.6 First N-gram

A boolean value representing whether this N-gram is the first one in the file. Returns 1 if true, 0 otherwise.

9.1.7 Last N-gram

A boolean value representing whether this N-gram is the last one in the file. Returns 1 if true, 0 otherwise.

9.1.8 Latency

The time in seconds in the file for this N-gram to appear. Similar concept as in (8.2.1.4).

9.1.9 N-gram Position

The position of this N-gram in the file. Similar concept as in (8.2.1.5).

9.1.10 Total Annotation Time

This column will only be present if the N-gram size is greater than one. The time in seconds for the annotations in this N-gram. Similar concept as in (8.2.1.6).

9.1.11 Total Interval Time

This column will only be present if the N-gram size is greater than one. The time in seconds of the intervals between annotations in this N-gram. Similar concept as in (8.2.1.7).

10. RESULTS

With the extended functionality added to ELAN, it was time to analyze the ASLLRP corpus to see what we could discern. For a first pass, we generated a listing of bigrams/trigrams/quadgrams in the corpus sorted by occurrences (8.1.5) on the main gloss tier. A sample row of data from a bigram is displayed in (Appendix 2). The bigram data is in (Table 2), the trigram data is in (Table 3), and the quadgram data is in (Table 4).

Table 2. ASLLRP top 5 bigrams sorted by occurrences.

HOLD|IX-1p	41
READ|BOOK	41
part:indef|HOLD	35
fs-JOHN|HOLD	33
BUY|HOUSE	32

Table 3. ASLLRP top 5 trigrams sorted by occurrences.

FINISH|READ|BOOK	21
fs-JOHN|FINISH|READ	15
IX-3p:i|OLD|MAN	12
OLD|MAN|ARRIVE	12
READ|BOOK|HOLD	12

Table 4. ASLLRP top 5 quadgrams sorted by occurrences.

fs-JOHN|FINISH|READ|BOOK	12
IX-3p:i|OLD|MAN|ARRIVE	12
RAIN|IX-1p|GO-OUT|MOVIE	10
fs-JOHN|BUY|YESTERDAY|"WHAT"	8
fs-JOHN|SEE|WHO|YESTERDAY	7

It was not that interesting to see the N-grams sorted by occurrences as the ASLLRP corpus contained a lot of similar utterances. A better perspective was enabled by looking at the newfound statistics in the contingency tables. What follows is the listing of bigrams sorted by T-score (Table 5), Log-likelihood (Table 6), and Poisson-Stirling Measure (Table 7). The corpus was skewed due to the small size and other metrics didn't reveal anything interesting or calculated invalid values.

Table 5. ASLLRP top 5 bigrams sorted by T-score.

READ|BOOK	6.2901917496
BUY|HOUSE	5.5409633338
FINISH|READ	5.2800307345
BUY|CAR	5.1479860252
part:indef|HOLD	5.1049584814

Table 6. ASLLRP top 5 bigrams sorted by Log-likelihood.

READ|BOOK	301.3458757921
BUY|HOUSE	216.6114915741
FINISH|READ	196.6206869188
ICL"nailing"|ICL"nailing"	151.4392662362
BUY|CAR	140.1247669654

Table 7. ASLLRP top 5 bigrams sorted by Poisson-Stirling Measure.

READ|BOOK	124.5477961338
BUY|HOUSE	92.4151636457
FINISH|READ	85.1488365045
ICL"nailing"|ICL"nailing"	62.8151343883
BUY|CAR	61.5550805055

It was good to see a high degree of correlation between several metrics in the corpus, but we knew that the corpus needed to be expanded in order to utilize the other metrics. Another thing we did was to analyze the relationships between glosses by taking the T-score sorted list and filtering rows based on search criteria. Some interesting collocations were discovered based on this method and a sample output is presented in (Table 8) and the reversed order in (Table 9).

Table 8. ASLLRP top 5 collocations of READ sorted by T-score.

READ|BOOK	6.2901917496
READ|MAGAZINE	1.9831832384
READ|BOOK+	1.8962966366
READ|YESTERDAY	1.5702314434
READ|LIP	1.4062860649

Table 9. ASLLRP top 5 collocations of READ sorted by T-score.

FINISH|READ	5.2800307345
fs-JOHN|READ	2.4195405281
FUTURE|READ	2.0380232875
TURN|READ	1.7158688712
(2h)MUST|READ	1.4062860649

The contingency tables proved to be a powerful method of looking at the corpus. To further the team's coarticulation efforts different data was needed and it was included in the CSV export. The timing data is contained in the "Total Interval Time" column and with the proper filtering/sorting it revealed interesting relationships between collocated glosses and their timings. A sample output is given in (Table 10) and the reversed order in (Table 11).

Table 10. ASLLRP top 5 collocations of CAR sorted by Total Interval Time|Mean, descending.

CAR|FINISH	1.1s
CAR|i:(1h)GIFT:k	0.333s
CAR|STEAL	0.30825s
CAR|BOOK+	0.267s
CAR|IX-loc:j	0.266s

Table 11. ASLLRP top 5 collocations of CAR sorted by Total Interval Time|Mean, ascending.

CAR|HOLD	0s
CAR|NEVER	0.066s
CAR|BUY	0.089s
CAR|#NO	0.1s
CAR|(1h)GIFT:k	0.1s

The scope of this project is now finished, and the improved analytics capabilities of ELAN will help the team study the corpus in depth and develop coarticulation models based on corpus data. Many interesting combinations of the metrics can be imagined and sorted to their content to discover new relationships.

11. FUTURE WORK

This project was completed over the summer, and time constraints prevented the implementation of everything that was discussed in our brainstorming sessions and research. Some ideas are listed in this section, and hopefully the opportunity will present itself for somebody to tackle them and contribute massively to ELAN and related corpora research.

11.1 Parallel Corpus Analysis

With the application of the contingency tables, the ability is there for ELAN to analyze multiple search domains at the same time and present concordances between the domains. This is an extension of the metrics and would be a great fit for some of them as they are tailored towards that goal.

11.2 Tier Correlation Analysis

With the infrastructure in place to generate N-grams, it should be trivial to change the focus of the search engine from a single tier to multiple tiers. Finding correlations between tiers and generating N-grams of the matches would be immensely helpful to the sign language linguistics community. It can answer questions such as “Which gloss is commonly used with a head nod?” or “When the eyes are gazing left, which annotations are active?” It is our belief that kind of data would be the next step in corpus analytics.

11.3 Speed Optimizations

Programming the Java code was completed in a relatively short time and could benefit from dedicated math libraries to reduce the calculation time. Furthermore, taking advantage of multiple processors could give another boost to reducing the time to calculate all the statistics.

11.4 Skiplists and Improved Parsing

In our analytics of the ASLLRP corpus, we noticed that HOLD was one of the most common annotations in the gloss tier. It would be beneficial to have a GUI dialog to input annotations to skip in order to improve the accuracy of the analysis. Furthermore, adding algorithms to the N-gram builder could help in identifying common boundaries and segmenting them to improve the N-gram position indicators, revealing the true position of specific annotations in the corpus.

12. ACKNOWLEDGMENTS

This project was made possible by the generous grants from the Distributed Research Experiences for Undergraduates (DREU) and National Alliance for Access to Computing Careers (AccessComputing) organizations.

AccessComputing: http://www.washington.edu/accesscomputing

DREU: http://parasol.tamu.edu/dreu

Research Blog: http://0ne.us/DREU

Furthermore, I would like to thank my advisor, Dr. James Nickerson from Gallaudet University for encouraging me to join this program. My mentor, Dr. Rosalee Wolfe at DePaul University was a wonderful person to work with and I learned a tremendous amount in only a short time. I wish I could continue this summer research internship as it was the perfect fusion of my math, programming, and ASL knowledge.

13. REFERENCES

[1] Stokoe, William C. (1978) Sign Language Structure: The First Linguistic Analysis of AMERICAN SIGN LANGUAGE. Linstok Press.

[2] ASL Team. (N.D.) Retrieved August 3, 2013 from The DePaul University American Sign Language Project: http://asl.cs.depaul.edu/

[3] Neidle, C., & Vogler, C. (2012) A New Web Interface to Facilitate Access to Corpora: Development of the ASLLRP Data Access Interface. *Proceedings of the 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, LREC 2012, Istanbul, Turkey.*

[4] Neidle, Carol. (2007) SignStream Annotation: Addendum to Conventions used for the American Sign Language Linguistic Research Project. *American Sign Language Linguistic Research Project, Report 13.* Boston University, Boston, MA.

[5] Vogler, Christian. (N.D.) SignStream-XMLParser. Retrieved August 3, 2013 from ASLLRP: http://www.bu.edu/asllrp/ncslgr-for-download/signstream-parser.zip

[6] Wolfe, R. (2012) SignStream to ELAN converter. Unpublished software. DePaul University, Chicago, IL.

[7] ELAN. (N.D.) Retrived August 3, 2013 from The Language Archive, Max Planck Institute for Psycholinguistics: http://tla.mpi.nl/tools/tla-tools/elan/

[8] Valli C., & Lucas C. (1998) *Linguistics of American Sign Language*. Gallaudet University Press, Washington, DC.

[9] Fallora, J. (2012) SignStream histogram generator. Unpublished software. DePaul University, Chicago, IL.

[10] Hanke, T. (2002) iLex - A tool for Sign Language Lexicography and Corpus Analysis. *Proceedings of the International Conference on Language Resources and Evaluation 2002.*

[11] Kipp, M. (2001) Anvil - A Generic Annotation tool for Multimodal Dialogue. *Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech), p1367-1370*

[12] Bird, S., Day, D., Garofolo, J., Henderson, J., Laprun, C., & Liberman, M. (2000) ATLAS: A flexible and extensive architecture for linguistic annotation. *ArXiv preprint cs/0007022.*

[13] Johnston, T. (2012) Lexical frequency in sign languages. *Journal of deaf studies and deaf education, 17(2), p163-193.*

[14] Elliott, R., Glauert, J.R.W., Jennings, V., & Kennaway, J.R. (2004) An overview of the SiGML notation and SiGMLSigning software system. *In Sign Language Processing Satellite Workshop of the Fourth International Conference on Language Resources and Evaluation, LREC.*

[15] Hanke, T. (2004) HamNoSys – Representing sign language data in language resources and language processing contexts. *In Sign Language Processing Satellite Workshop of the Fourth International Conference on Language Resources and Evaluation, LREC.*

[16] Zhao, L., Kipper, K., Schuler, W., Vogler, C., Badler, N., & Palmer, M. (2000) A machine translation system from English to American Sign Language. *In Envisioning Machine Translation in the Information Future p54-67.* Springer Berlin Heidelberg.

[17] Ong, S.C., & Ranganath, S. (2005) Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(6), p873-891.*

[18] Vogler, C., & Metaxas, D. (1999) Parallel hidden markov models for american sign language recognition. *In Proceedings of the Seventh IEEE International Conference on Computer Vision, p116-122.*

[19] Dunning, T. (1993) Accurate methods for the statistics of surprise and coincidence. *Computational linguistics, 19(1) p61-74.*

[20] Pecina, P. (2005) An extensive empirical study of collocation extraction methods. *In Proceedings of the ACL Student Research Workshop June 2005, p13-18.* Association for Computational Linguistics.

[21] Bannerjee, S., & Pedersen, T. (2003) The design, implementation, and use of the ngram statistics package. *In Computational Linguistics and Intelligent Text Processing p370-81.* Springer Berlin Heidelberg.

[22] Plackett, R.L. (1983) Karl Pearson and the Chi-Squared Test. *International Statistical Review 51(1), p59-72.*

[23] Smadja, F., McKeown, K. R., & Hatzivassiloglou, V. (1996) Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics 22(1), p1-38.*

[24] Pedersen, T. (1996) Fishing for exactness. *ArXiv preprint cmp-lg/9608010.*

[25] Chung, Y.M., & Lee, J. Y. (2001) A corpus-based approach to comparative evaluation of statistical term association measures. *Journal of the American Society for Information Science and Technology, 52(4), p283-296.*

[26] Moore, R. C. (2004) On Log-Likelihood-Ratios and the Significance of Rare Events. *In Proceedings of Empirical Methods in Natural Language Processing July 2004, p333-340.*

[27] Zipf, G.K. (1949) Human behavior and the principle of least effort. Addison-Wesley Press. Oxford, England.

[28] Szmrecsanyi, B. (2005) Language users as creatures of habit: A corpus-based analysis of persistence in spoken English. *Corpus Linguistics and Linguistic Theory, 1(1), p113-150.*

[29] Blaheta, D., & Johnson, M. (2001). Unsupervised learning of multi-word verbs. *In Proceedings of the ACL/EACL 2001 Workshop on the Computational Extraction, Analysis and Exploitation of Collocations, p54-60.*

[30] Church, K. W., & Gale, W. A. (1991). Concordances for parallel text. *In Proceedings of the Seventh Annual Conference of the UW Centre for the New OED and Text Research, p40-62.*

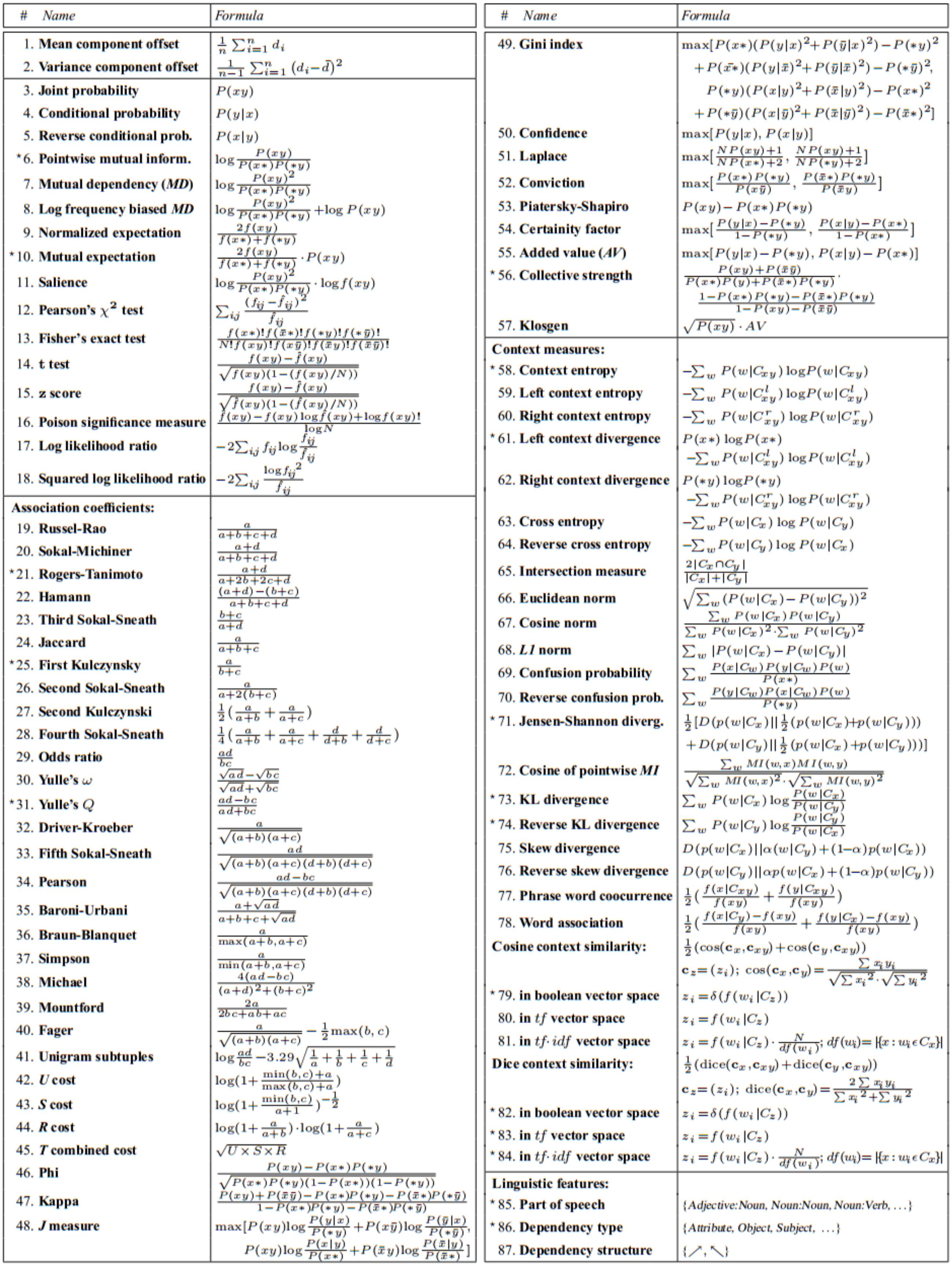
[31] Church, K. W., & Hanks, P. (1990) Word association norms, mutual information, and lexicography. *Computational Linguistics, 16(1), p22-29.*

[32] Palermo, D., & Jenkins, J. (1964) Word Association Norms. University of Minnesota Press, Minneapolis, MN.

[33] Quasthoff, U., & Wolff, C. (2002) The poisson collocation measure and its applications. *In Proceedings of Workshop on Computational Approaches to Collocations. Wien, Austria.*

[34] Church, K., Gale, W., Hanks, P., & Kindle, D. (1991) 6. using statistics in lexical analysis. *Lexical acquisition: exploiting on-line resources to build a lexicon.* Psychology Press.

[35] Lin, D. (1999) Automatic identification of non-compositional phrases. *In Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, p317-324.*



# Name	Formula
1. **Mean component offset**	$\frac{1}{n}\sum_{i=1}^{n} d_i$
2. **Variance component offset**	$\frac{1}{n-1}\sum_{i=1}^{n} (d_i-\bar{d})^2$
3. **Joint probability**	$P(xy)$
4. **Conditional probability**	$P(y\mid x)$
5. **Reverse conditional prob.**	$P(x\mid y)$
⋆6. **Pointwise mutual inform.**	$\log\frac{P(xy)}{P(x*)P(*y)}$
7. **Mutual dependency (*MD*)**	$\log\frac{P(xy)^2}{P(x*)P(*y)}$
8. **Log frequency biased *MD***	$\log\frac{P(xy)^2}{P(x*)P(*y)}+\log P(xy)$
9. **Normalized expectation**	$\frac{2f(xy)}{f(x*)+f(*y)}$
⋆10. **Mutual expectation**	$\frac{2f(xy)}{f(x*)+f(*y)}\cdot P(xy)$
11. **Salience**	$\log\frac{P(xy)^2}{P(x*)P(*y)}\cdot\log f(xy)$
12. **Pearson's χ^2 test**	$\sum_{ij}\frac{(f_{ij}-\hat{f}_{ij})^2}{\hat{f}_{ij}}$
13. **Fisher's exact test**	$\frac{f(x*)!f(\bar{x}*)!f(*y)!f(*\bar{y})!}{N!f(xy)!f(x\bar{y})!f(\bar{x}y)!f(\bar{x}\bar{y})!}$
14. **t test**	$\frac{f(xy)-\hat{f}(xy)}{\sqrt{f(xy)(1-(f(xy)/N))}}$
15. **z score**	$\frac{f(xy)-\hat{f}(xy)}{\sqrt{\hat{f}(xy)(1-(\hat{f}(xy)/N))}}$
16. **Poison significance measure**	$\frac{\hat{f}(xy)-f(xy)\log\hat{f}(xy)+\log f(xy)!}{\log N}$
17. **Log likelihood ratio**	$-2\sum_{ij} f_{ij}\log\frac{f_{ij}}{\hat{f}_{ij}}$
18. **Squared log likelihood ratio**	$-2\sum_{ij}\frac{\log f_{ij}{}^2}{\hat{f}_{ij}}$
Association coefficients:	
19. **Russel-Rao**	$\frac{a}{a+b+c+d}$
20. **Sokal-Michiner**	$\frac{a+d}{a+b+c+d}$
⋆21. **Rogers-Tanimoto**	$\frac{a+d}{a+2b+2c+d}$
22. **Hamann**	$\frac{(a+d)-(b+c)}{a+b+c+d}$
23. **Third Sokal-Sneath**	$\frac{b+c}{a+d}$
24. **Jaccard**	$\frac{a}{a+b+c}$
⋆25. **First Kulczynsky**	$\frac{a}{b+c}$
26. **Second Sokal-Sneath**	$\frac{a}{a+2(b+c)}$
27. **Second Kulczynski**	$\frac{1}{2}(\frac{a}{a+b}+\frac{a}{a+c})$
28. **Fourth Sokal-Sneath**	$\frac{1}{4}(\frac{a}{a+b}+\frac{a}{a+c}+\frac{d}{d+b}+\frac{d}{d+c})$
29. **Odds ratio**	$\frac{ad}{bc}$
30. **Yulle's ω**	$\frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}$
⋆31. **Yulle's Q**	$\frac{ad-bc}{ad+bc}$
32. **Driver-Kroeber**	$\frac{a}{\sqrt{(a+b)(a+c)}}$
33. **Fifth Sokal-Sneath**	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$
34. **Pearson**	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$
35. **Baroni-Urbani**	$\frac{a+\sqrt{ad}}{a+b+c+\sqrt{ad}}$
36. **Braun-Blanquet**	$\frac{a}{\max(a+b,a+c)}$
37. **Simpson**	$\frac{a}{\min(a+b,a+c)}$
38. **Michael**	$\frac{4(ad-bc)}{(a+d)^2+(b+c)^2}$
39. **Mountford**	$\frac{2a}{2bc+ab+ac}$
40. **Fager**	$\frac{a}{\sqrt{(a+b)(a+c)}}-\frac{1}{2}\max(b,c)$
41. **Unigram subtuples**	$\log\frac{ad}{bc}-3.29\sqrt{\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}}$
42. ***U* cost**	$\log(1+\frac{\min(b,c)+a}{\max(b,c)+a})$
43. ***S* cost**	$\log(1+\frac{\min(b,c)}{a+1})^{-\frac{1}{2}}$
44. ***R* cost**	$\log(1+\frac{a}{a+b})\cdot\log(1+\frac{a}{a+c})$
45. ***T* combined cost**	$\sqrt{U\times S\times R}$
46. **Phi**	$\frac{P(xy)-P(x*)P(*y)}{\sqrt{P(x*)P(*y)(1-P(x*))(1-P(*y))}}$
47. **Kappa**	$\frac{P(xy)+P(\bar{x}\bar{y})-P(x*)P(*y)-P(\bar{x}*)P(*\bar{y})}{1-P(x*)P(*y)-P(\bar{x}*)P(*\bar{y})}$
48. ***J* measure**	$\max[P(xy)\log\frac{P(y\mid x)}{P(*y)}+P(x\bar{y})\log\frac{P(\bar{y}\mid x)}{P(*\bar{y})},$ $P(xy)\log\frac{P(x\mid y)}{P(x*)}+P(\bar{x}y)\log\frac{P(\bar{x}\mid y)}{P(\bar{x}*)}]$
49. **Gini index**	$\max[P(x*)(P(y\mid x)^2+P(\bar{y}\mid x)^2)-P(*y)^2$ $+P(\bar{x}*)(P(y\mid\bar{x})^2+P(\bar{y}\mid\bar{x})^2)-P(*\bar{y})^2,$ $P(*y)(P(x\mid y)^2+P(\bar{x}\mid y)^2)-P(x*)^2$ $+P(*\bar{y})(P(x\mid\bar{y})^2+P(\bar{x}\mid\bar{y})^2)-P(\bar{x}*)^2]$
50. **Confidence**	$\max[P(y\mid x),P(x\mid y)]$
51. **Laplace**	$\max[\frac{NP(xy)+1}{NP(x*)+2},\frac{NP(xy)+1}{NP(*y)+2}]$
52. **Conviction**	$\max[\frac{P(x*)P(*y)}{P(x\bar{y})},\frac{P(\bar{x}*)P(*y)}{P(\bar{x}y)}]$
53. **Piatersky-Shapiro**	$P(xy)-P(x*)P(*y)$
54. **Certainity factor**	$\max[\frac{P(y\mid x)-P(*y)}{1-P(*y)},\frac{P(x\mid y)-P(x*)}{1-P(x*)}]$
55. **Added value (*AV*)**	$\max[P(y\mid x)-P(*y),P(x\mid y)-P(x*)]$
⋆56. **Collective strength**	$\frac{P(xy)+P(\bar{x}\bar{y})}{P(x*)P(y)+P(\bar{x}*)P(*y)}\cdot$ $\frac{1-P(x*)P(*y)-P(\bar{x}*)P(*y)}{1-P(xy)-P(\bar{x}\bar{y})}$
57. **Klosgen**	$\sqrt{P(xy)}\cdot AV$
Context measures:	
⋆58. **Context entropy**	$-\sum_w P(w\mid C_{xy})\log P(w\mid C_{xy})$
59. **Left context entropy**	$-\sum_w P(w\mid C^l_{xy})\log P(w\mid C^l_{xy})$
60. **Right context entropy**	$-\sum_w P(w\mid C^r_{xy})\log P(w\mid C^r_{xy})$
⋆61. **Left context divergence**	$P(x*)\log P(x*)$ $-\sum_w P(w\mid C^l_{xy})\log P(w\mid C^l_{xy})$
62. **Right context divergence**	$P(*y)\log P(*y)$ $-\sum_w P(w\mid C^r_{xy})\log P(w\mid C^r_{xy})$
63. **Cross entropy**	$-\sum_w P(w\mid C_x)\log P(w\mid C_y)$
64. **Reverse cross entropy**	$-\sum_w P(w\mid C_y)\log P(w\mid C_x)$
65. **Intersection measure**	$\frac{2\lvert C_x\cap C_y\rvert}{\lvert C_x\rvert+\lvert C_y\rvert}$
66. **Euclidean norm**	$\sqrt{\sum_w(P(w\mid C_x)-P(w\mid C_y))^2}$
67. **Cosine norm**	$\frac{\sum_w P(w\mid C_x)P(w\mid C_y)}{\sum_w P(w\mid C_x)^2\cdot\sum_w P(w\mid C_y)^2}$
68. ***L1* norm**	$\sum_w\lvert P(w\mid C_x)-P(w\mid C_y)\rvert$
69. **Confusion probability**	$\sum_w\frac{P(x\mid C_w)P(y\mid C_w)P(w)}{P(x*)}$
70. **Reverse confusion prob.**	$\sum_w\frac{P(y\mid C_w)P(x\mid C_w)P(w)}{P(*y)}$
⋆71. **Jensen-Shannon diverg.**	$\frac{1}{2}[D(p(w\mid C_x)\parallel\frac{1}{2}(p(w\mid C_x)+p(w\mid C_y)))$ $+D(p(w\mid C_y)\parallel\frac{1}{2}(p(w\mid C_x)+p(w\mid C_y)))]$
72. **Cosine of pointwise *MI***	$\frac{\sum_w MI(w,x)MI(w,y)}{\sqrt{\sum_w MI(w,x)^2}\cdot\sqrt{\sum_w MI(w,y)^2}}$
⋆73. **KL divergence**	$\sum_w P(w\mid C_x)\log\frac{P(w\mid C_x)}{P(w\mid C_y)}$
⋆74. **Reverse KL divergence**	$\sum_w P(w\mid C_y)\log\frac{P(w\mid C_y)}{P(w\mid C_x)}$
75. **Skew divergence**	$D(p(w\mid C_x)\parallel\alpha(w\mid C_y)+(1-\alpha)p(w\mid C_x))$
76. **Reverse skew divergence**	$D(p(w\mid C_y)\parallel\alpha p(w\mid C_x)+(1-\alpha)p(w\mid C_y))$
77. **Phrase word coocurrence**	$\frac{1}{2}(\frac{f(x\mid C_{xy})}{f(xy)}+\frac{f(y\mid C_{xy})}{f(xy)})$
78. **Word association**	$\frac{1}{2}(\frac{f(x\mid C_y)-f(xy)}{f(xy)}+\frac{f(y\mid C_x)-f(xy)}{f(xy)})$
Cosine context similarity:	$\frac{1}{2}(\cos(\mathbf{c}_x,\mathbf{c}_{xy})+\cos(\mathbf{c}_y,\mathbf{c}_{xy}))$ $\mathbf{c}_z=(z_i);\ \cos(\mathbf{c}_x,\mathbf{c}_y)=\frac{\sum x_i y_i}{\sqrt{\sum x_i{}^2}\cdot\sqrt{\sum y_i{}^2}}$
⋆79. **in boolean vector space**	$z_i=\delta(f(w_i\mid C_z))$
80. **in *tf* vector space**	$z_i=f(w_i\mid C_z)$
81. **in *tf·idf* vector space**	$z_i=f(w_i\mid C_z)\cdot\frac{N}{df(w_i)};\ df(w_i)=\lvert\{x:w_i\epsilon C_x\}\rvert$
Dice context similarity:	$\frac{1}{2}(\mathrm{dice}(\mathbf{c}_x,\mathbf{c}_{xy})+\mathrm{dice}(\mathbf{c}_y,\mathbf{c}_{xy}))$ $\mathbf{c}_z=(z_i);\ \mathrm{dice}(\mathbf{c}_x,\mathbf{c}_y)=\frac{2\sum x_i y_i}{\sum x_i{}^2+\sum y_i{}^2}$
⋆82. **in boolean vector space**	$z_i=\delta(f(w_i\mid C_z))$
⋆83. **in *tf* vector space**	$z_i=f(w_i\mid C_z)$
⋆84. **in *tf·idf* vector space**	$z_i=f(w_i\mid C_z)\cdot\frac{N}{df(w_i)};\ df(w_i)=\lvert\{x:w_i\epsilon C_x\}\rvert$
Linguistic features:	
⋆85. **Part of speech**	{*Adjective:Noun, Noun:Noun, Noun:Verb,* …}
⋆86. **Dependency type**	{*Attribute, Object, Subject,* …}
87. **Dependency structure**	{↗, ↖}

Appendix 1. Pecina's Collocation Extraction Methods.

N-gram	READ|BOOK	**After Interval|Kurtosis**	0.0499770475	**After Interval|Max**	0.267
After Interval|Mean	0.09490625	**After Interval|Median**	0.1	**After Interval|Min**	0
After Interval|Mode	0	**After Interval|Quartile1**	0	**After Interval|Quartile3**	0.167
After Interval|Skewness	0.0360263132	**After Interval|StdDev**	0.0880972188	**After Interval|Variance**	0.00776112
Before Interval|Kurtosis	0.088082922	**Before Interval|Max**	0.267	**Before Interval|Mean**	0.1343170732
Before Interval|Median	0.134	**Before Interval|Min**	0.033	**Before Interval|Mode**	0.1
Before Interval|Quartile1	0.1	**Before Interval|Quartile3**	0.167	**Before Interval|Skewness**	0.1208337897
Before Interval|StdDev	0.0487085408	**Before Interval|Variance**	0.002372522	**Duration|Kurtosis**	0.10522378
Duration|Max	0.834	**Duration|Mean**	0.425097561	**Duration|Median**	0.4
Duration|Min	0.167	**Duration|Mode**	0.4	**Duration|Quartile1**	0.3495
Duration|Quartile3	0.5	**Duration|Skewness**	0.1368870124	**Duration|StdDev**	0.1344374957
Duration|Variance	0.0180734402	**File Occurrences**	41	**First N-gram**	0
Last N-gram	9	**Latency|Kurtosis**	0.2543078537	**Latency|Max**	6.5
Latency|Mean	1.6534146341	**Latency|Median**	1.233	**Latency|Min**	0.6
Latency|Mode	1.233	**Latency|Quartile1**	1	**Latency|Quartile3**	1.933
Latency|Skewness	0.375469452	**Latency|StdDev**	1.0962834254	**Latency|Variance**	1.2018373488
N-gram Position|Kurtosis	0.2699299013	**N-gram Position|Max**	17	**N-gram Position|Mean**	4.3170731707
N-gram Position|Median	3	**N-gram Position|Min**	2	**N-gram Position|Mode**	3
N-gram Position|Quartile1	2.5	**N-gram Position|Quartile3**	5	**N-gram Position|Skewness**	0.3826203245
N-gram Position|StdDev	2.8145961024	**N-gram Position|Variance**	7.9219512195	**Occurrences**	41
Total Annotation Time|Kurtosis	0.1494467014	**Total Annotation Time|Max**	0.667	**Total Annotation Time|Mean**	0.2584390244
Total Annotation Time|Median	0.234	**Total Annotation Time|Min**	0.099	**Total Annotation Time|Mode**	0.267
Total Annotation Time|Quartile1	0.1995	**Total Annotation Time|Quartile3**	0.268	**Total Annotation Time|Skewness**	0.218791281
Total Annotation Time|StdDev	0.1107497288	**Total Annotation Time|Variance**	0.0122655024	**Total Interval Time|Kurtosis**	0.1434024263
Total Interval Time|Max	0.4	**Total Interval Time|Mean**	0.1666585366	**Total Interval Time|Median**	0.166
Total Interval Time|Min	0.066	**Total Interval Time|Mode**	0.2	**Total Interval Time|Quartile1**	0.133
Total Interval Time|Quartile3	0.2	**Total Interval Time|Skewness**	0.1736940625	**Total Interval Time|StdDev**	0.0627883786
Total Interval Time|Variance	0.0039423805	**cT|Chi-squared**	2283.0803835282	**cT|Dice Coefficient**	0.4315789474
cT|Fisher Exact Left Sided	1	**cT|Fisher Exact Right Sided**	4.80782203369758E-067	**cT|Fisher Exact Two Tailed**	4.80782203369758E-067
cT|Jaccard Coefficient	0.2751677852	**cT|Log-likelihood**	301.3458757921	**cT|Odds Ratio**	250.0301136364
cT|Phi Coefficient	0.2098033802	**cT|Pointwise Mutual Information**	5.8252435247	**cT|Poisson-Stirling Measure**	124.5477961338
cT|T-score	6.2901917496	**cT|True Mutual Information**	378.4420409622		

Appendix 2. Sample row from the analytics of bigrams in the ASLLRP corpus.