First Results in a Study Evaluating Pre-annotation and Correction Propagation for Machine-Assisted Syriac Morphological Analysis

Paul Felt, Eric Ringger, Kevin Seppi, Kristian Heal[†], Robbie Haertel, Deryle Lonsdale[‡]

Dept. of Computer Science, [†]Neal A. Maxwell Institute, [‡]Dept. of Linguistics Brigham Young University, Provo, Utah 84602 USA

{paul_felt, eric_ringger, kseppi, kristian_heal, robbie_haertel, lonz}@byu.edu

Abstract

Manual annotation of large textual corpora can be cost-prohibitive, especially for rare and under-resourced languages. One potential solution is *pre-annotation*: asking human annotators to correct sentences that have already been annotated, usually by a machine. Another potential solution is *correction propagation*: using annotator corrections to dynamically improve to the remaining pre-annotations within the current sentence. The research presented in this paper employs a controlled user study to discover under what conditions these two machine-assisted annotation techniques are effective in increasing annotator speed and accuracy and thereby reducing the cost for the task of morphologically annotating texts written in classical Syriac. A preliminary analysis of the data indicates that pre-annotations improve annotator accuracy when they are at least 60% accurate, and annotator speed when they are at least 80% accurate. This research constitutes the first systematic evaluation of pre-annotation and correction propagation together in a controlled user study.

Keywords: Annotated Corpora, Annotation, User Study

1. Introduction

The current success and widespread use of data-driven techniques for processing human language make annotated corpora an essential language resource. For instance, many popular natural language processing (NLP) algorithms require significant amounts of high quality annotated training data in order to perform effectively. Also, annotated text can be useful in its own right as a means of exploring the language and the culture that produced it. For example, one might use syntactic annotations to study discourse patterns, or topical annotations to track the movement of important ideas through time and space.

Scholars at the Center for the Preservation of Ancient Religious Texts (CPART) of the Neal A. Maxwell Institute for Religious Scholarship at BYU and at the Oriental Institute at the University of Oxford are jointly working on a project called the Syriac Electronic Corpus, with the goal of creating a comprehensive, labeled corpus of classical Syriac. Classical Syriac ('kthobonoyo') is an under-resourced Semitic language of the Christian Near East and a dialect of Aramaic. It was largely replaced by Arabic as a spoken language by the end of the ninth century, and is now primarily a liturgical language. Many prolific authors wrote in Syriac. The goal of the Syriac Electronic Corpus project is to annotate all of these texts with morphological information to facilitate systematic study of Syriac by historians, linguists, and language learners.

رمېْحكىتْك	مکن	مححل	act
token	stem	citation form	root

Figure 1. The Syriac word token LMaLK'K,uON "to your king" and its related forms.

Morphological analysis of Syriac involves segmenting a word into its constituent morphemes and labeling each according to its grammatical form(s). For our purposes, a word token consists of a prefix, a suffix, and a *stem*, which we define as the remaining text. The dictionary citation form (or baseform) and, where applicable, the root are identified from the stem (Figure 1).

In contrast to English, where searching for a few forms of a word is often sufficient for discovering patterns reflecting the word's usage and meaning, in Semitic languages search and discovery are not so straightforward. If we could search Syriac texts on citation forms or even on roots, we could search for and discover patterns as easily as in English; however, Syriac roots are altered by extensive inflectional and derivational morphological processes such that numerous surface forms correspond to any given root. As a result, searching Syriac text is ineffective since one must either limit one's query to a single inflected surface form or use heuristics to expand the query, buying higher recall at the price of lower precision.

A morphologically annotated digital corpus of a lesser studied language lends itself to search and therefore to careful study in a way that formerly only experts could attempt based on long years of familiarity. Such annotated corpora enable scholars to study and discover the contributions of and trends in historical documents. One outstanding example of such a corpus is the Dead Sea Scrolls Electronic Library, assembled by CPART scholars (Tov, 2007). The Syriac Corpus will be an artifact of similar value to linguists, Syriac students, and scholars of Syriac, the Near East, and Eastern Christianity.

Unfortunately, creating annotated corpora can be extremely time-consuming. The Way International Foundation, a Biblical research, teaching, and fellowship ministry, spent 15 years labeling the Syriac New Testament with morphological annotations (Kiraz, 1994). The Syriac New Testament consists of approximately 100,000 words. Similarly, two Syriac scholars we worked with during the course of this research informally report taking two years to label about one fourth of the Old Testament. By contrast, the Syriac Corpus will encompass over 10,000,000 words. To achieve this goal in a timely manner it will be necessary to increase the speed of annotation.

Pre-annotation, also known as pre-labeling, has the potential to reduce annotation cost by using NLP algorithms to automatically annotate each instance (i.e. sentence) before it is presented to an expert annotator. Expert annotators then need only review and correct the proposed annotations, which can potentially be done much more quickly than annotating from scratch.

Kristjannson et al. (2004) describe an enhancement to pre-annotation for multi-part annotation tasks which they call correction propagation. Correction propagation consists of triggering a pre-annotation update whenever an annotator corrects a pre-annotation. The idea is that the machine annotator can use the correction to improve its guesses regarding other decisions to be made for the item currently being annotated (e.g. sentence). Kristjannson et al. give the example of identifying contact information in free text. In this case, correcting a pre-annotated given name might allow the automatic annotator to correctly identify a corresponding surname and address. To be clear, correction propagation does not involve retraining a model using the new data. Rather, it involves making a multi-part prediction in a hypothesis space that is constrained by a partial annotation.

Both pre-annotation and correction propagation require a model capable of supplying automatic annotations, and correction propagation additionally requires the ability to constrain and update automatic annotations. However, as noted earlier, many NLP algorithms for building such a model require previously annotated training data. For tasks and languages without already existing resources, one must therefore begin the annotation process with low quality pre-annotations and periodically retrain the pre-annotator as more data is labeled. Although pre-annotation and correction propagation attempt to increase annotator efficiency, it is conceivable that inaccurate predictions could reduce annotator speed or accuracy. Because of this, before building annotated corpora in domains with little labeled data, it is desirable to have a sense of how accurate a model must be in order to make pre-annotation and correction propagation helpful instead of harmful. This research constitutes the first systematic evaluation of pre-annotation and correction propagation together in a controlled user study.

2. Related Work

In order to generate pre-annotations and correction propagation updates for Syriac morphological analysis, we use Syromorph, a probabilistic morphological analyzer for Syriac described by McClanahan et al. (2010). Syromorph is an n-best pipeline of classification and transduction tasks. Each task in the pipeline proposes hypotheses based on the data and the results of all previous tasks. Solutions are chosen by running a beam search over all the hypotheses in the pipeline, allowing decisions to be made in a global context without incurring the cost of full joint inference. Syromorph first segments each word into its parts: prefix, stem, and suffix. Syromorph then predicts a baseform, or dictionary citation form, for the stem. Finally, Syromorph predicts the grammatical attributes of the stem and suffix.¹

Pre-annotation has been evaluated on a variety of tasks. Marcus et al. (1993) evaluated pre-annotation using an interface embedded in the GNU Emacs Editor to label the Penn Treebank with English Part-of-speech (POS) tags. They manually timed four annotators and reported that pre-annotation more than doubled annotation speed and also increased accuracy and inter-annotator agreement. Chiou et al. (2001) timed two annotators using an unspecied tool and reported a 70% increase in annotation speed using pre-annotation on a Chinese Treebank annotation task. Baldridge & Osborne (2004) present several choices rather than the single best for a parsing task and report a 74% reduction in cost. Similarly, Ganchev et al. (2007) present a set of candidate pre-annotations to annotators doing named entity recognition. They manually recorded the time of a single annotator and reported a more than 50% increase in speed. Brants & Plaehn (2000) applied pre-annotation to parse tree labeling. In order to make pre-annotation effective for parse tree labeling, they found they had to alter their pre-annotation approach by creating an interactive parse tree where annotators accept or reject suggestions starting at the parse tree's leaves and working their way to the root.

Correction propagation has been evaluated on far fewer tasks than pre-annotation. As has already been noted, Kristjannson et al. (2004) applied correction propagation to the task of information extraction, interactively assisting users to fill in database fields. They evaluated the performance of correction propagation in simulation and showed that automatic annotator accuracy significantly increased after even a single correction. They also showed that correction propagation significantly reduced the expected number of user interactions with a hypothetical graphical user interface.

These results are encouraging, but it is unclear which, if any, of the previous pre-annotation or correction propagation results apply to Syriac morphological analysis. For one thing, because of the differences between Syriac morphological analysis and the tasks evaluated by previous work, pre-annotation and correction propagation must be implemented differently. Most importantly, all previous work evaluates the effectiveness of only the highest possible quality pre-annotations and correction propaga-

¹ In accordance with the current needs of the Syriac Corpus project, the original Syromorph (v1.0) has been modified slightly so that it no longer predicts a root form (current version is 2.1). The reason for this change is that the ultimate goal of the project is to link each token to a baseform dictionary entry, and the root form comes for free with this linkage.

tion. However, in many under-resourced language domains little annotated data is available. In such domains, data-driven predictive models necessarily start out with poor accuracies and gradually improve as annotations are accumulated. We are interested in the performance of pre-labeling and correction propagation across all accuracy levels.

3. Methodology

This section describes the conditions under which the data was collected; a preliminary analysis of the data is described in Section 6.

This section will proceed as follows: sub-section 3.1 gives an overview of the user study layout; 3.2 describes the training and evaluation of the automatic annotation models used in the study; 3.3 shows via simulation that correction propagation has the potential to increase effective pre-annotation accuracy; 3.4 explains our method of assigning experimental conditions to participants; 3.5 describes the user study participants; 3.6 describes the framework used to conduct the study and the study's graphical user interface.

3.1 User Study Overview

We designed a web-mediated user study using CCASH,² an open source web application framework for linguistic annotation tasks (Felt, 2010). In the study, annotators took a survey, received a brief training, and then worked through four practice sentences. After each practice sentence, participants received feedback on how their annotations differed from the annotation guidelines they were given. They were required to achieve a high level of accuracy on the final practice sentence before proceeding. Finally, participants annotated 30 sentences under a sequence of randomly assigned experimental conditions, explained in Section 3.4. For each word in the study, CCASH recorded the time each annotator took to spent as well as the number of correct and incorrect decisions they made.

The choice to have all participants annotate the same 30 sentences does not limit our ability to collect large amounts of data and identify statistical trends associated with different annotation conditions. It does limit the applicability of our results to new data; however, that is a problem inherent in any focused study.

A gold standard annotation was constructed by two expert Syriac linguists who completed the study, then discussed and resolved all disagreements in their annotations. It should be noted that annotated Syriac text already exists: The Syriac Peshitta New Testament has been labeled with morphological information (Kiraz, 1994). However, reference copies of this data have been published which could bias the results of our study. Accordingly, the 30 sentences for the study were selected uniformly at random from The Acts of Judas Thomas, an apocryphal text that is similar, but not identical, to the New Testament (Wright,

² http://ccash.sourceforge.net

1871).

When constructing a gold standard, it is important to acknowledge that there are some difficult cases that even experts have difficulty agreeing on (Klebanov, 2009). However, the disagreements between our experts indicated that only around 20 of the 1289 decisions in the user study were difficult. This rate is low enough that it should not greatly affect our results.

3.2 Model Training and Metrics

We trained Syromorph models on various random subsets of the Syriac New Testament data assembled by Kiraz (1994) and augmented with suffix data by McClanahan et al. (2010), consisting of approximately 100,000 labeled tokens. We calculated model accuracy against the 30 Judas Thomas sentences in the study's gold standard. This slight mismatch between model training and test data caused model accuracy to suffer. Thus our most accurate model, trained on all of the New Testament data, achieved an accuracy of only slightly above 90%. In order to obtain models with given target accuracies, we trained Syromorph on random subsets of the training data until a model was found which achieved the desired accuracy $\pm 0.01\%$ measured against the gold standard.

In a multi-part annotation task like Syriac morphological analysis, accuracy can be calculated on the sentence level, the word level, or the decision level. These accuracy metrics are highly correlated, but not identical. Furthermore, since decisions can be partitioned into classes according to their sub-task, it is possible to calculate decision-level accuracy either as a macro-average or as a micro-average across decision types. A macro-average is computed by first averaging the decisions for a sub-task, then averaging the resulting averages. A micro-average is computed by averaging the decisions for all sub-tasks at once. Decision-level accuracy using a micro-average is an appropriate accuracy metric since it is computed over the exact set of choices that an annotator must make while annotating. All accuracies mentioned in this paper are decision-level micro-averages calculated against the 30 sentence gold standard set.

3.3 Simulated Correction Propagation

Before conducting a user study to test whether correction propagation reduces annotation effort in a scenario involving real users, we ran simulations to verify that correction propagation has the potential to increase effective pre-annotation accuracy.

In the first series of simulations, referred to in Figure 2 as "Without Correction Propagation," Syromorph models trained on increasing amounts of data were queried for labels a sentence at a time. In the second series of simulations, referred to in the figure as "With Correction Propagation," the same models were queried for labels a decision at a time, constrained by a correct partial labeling of all previous decisions in the sentence. This measures the accuracy of the pre-annotations an infallible annotator would encounter working sequentially through the deci-



Figure 2: Syromorph's accuracy with and without correction propagation.

sions of each sentence, where the model was allowed to update the sentence's pre-annotations after each decision.

Figure 2 shows that correction propagation allows models at all quality levels to improve the accuracy of their decisions by a modest amount. These simulations indicate that correction propagation has the potential to increase pre-annotation accuracy in practice. This increased pre-annotation accuracy could also conceivably increase annotator speed, since a more accurate pre-annotation will usually be easier to correct.

3.4 Experimental Conditions

Pre-annotations were supplied to annotators at the following accuracy levels: *none*, 25%, 35%, 45%, 55%, 65%, 75%, 90%, and 100%. In the *none* case, no pre-annotations were given. In the 100% case, gold standard annotations were given. In all intermediate cases, Syromorph models trained to the indicated accuracy provided pre-annotations. The accuracy levels between 25% and 90% inclusive were chosen to span the range of accuracies achievable by Syromorph trained on the Peshitta New Testament.

Additionally, participants annotated sentences both with and without the assistance of correction propagation. Note that correction propagation requires a model; consequently it cannot be applied to the *none* or 100% cases. In all, there are

 $|\{none, 100\}| + |\{25, 36, 47, 58, 68, 79, 90\} \times \{+CP, -CP\}|$

or 16 parameter combinations to test. We refer to each parameter combination as an *experimental condition*.

It is convenient to assign experimental conditions to participants and sentences using the matrix in Figure 3 where *Prt1* is the first participant to take the study, *St1* is the first sentence in the study, and cell values indicate a pre-annotation quality (25-100) and the optional presence of correction propagation (+C).

This matrix can be duplicated indefinitely to the right and the bottom. That is, Annotator 17 can be assigned to the same column as Annotator 1, and Sentence 17 can be assigned to the same row as Sentence 1. This parameter assignment scheme has some nice properties. It guarantees that each annotator encounter each experimental condition roughly the same number of times. It also ensures that each sentence will be encountered under each condition roughly the same number of times. However, this parameter assignment scheme has an important flaw: annotators encounter sentences of steadily increasing quality. Such an apparent trend may affect the way that annotators interact with the pre-annotations. This problem is resolved without sacrificing the nice properties of the assignment matrix by first permuting the rows of the matrix and afterwards the columns. Annotators thus encountered the study's sentences in a fixed order and under every experimental condition, but without an easily discernible pattern.

	Prt1	Prt2	Prt3	Prt4		Prt16
St1	0	25	25+C	36		100
St2	25	25+C	36	36+C		0
St3	25+C	36	36+C	47		25
			•••			•••
St16	100	0	25	25+C	•••	90+C

Figure 3: Experimental condition assignment scheme

It may be expected that annotators will begin to annotate slowly then move more quickly as they grow used to the task; this could potentially have a confounding effect on our timing data. We dealt with this learning effect in two ways. First, the training and practice at the beginning of the study allowed participants to become accustomed to the task and interface. Second, the parameter assignment scheme ensured that the sentences annotated under a given experimental condition include approximately equal numbers of sentences annotated early and late in the annotation process.

3.5 User Study Participants

Nine Syriac experts, invited by colleagues associated with CPART and the Oriental Institute at the University of Oxford, successfully completed the study. Their answers to the survey at the beginning of the study indicated that all participants consider themselves reasonably proficient in Syriac and comfortable using of computers.



Figure 4: The graphical user interface for Syriac morphological analysis used in the study.

3.6 Graphical User Interface

The graphical user interface used to conduct Syriac morphological analysis, implemented in CCASH, is an important part of this study since it affects annotation speed and also the applicability of this study to other tasks. Some time was spent refining the interface with Syriac experts to make sure it is reasonably efficient.

Annotators work through a sentence at a time. The sentence being annotated, along with some text preceding and following, is shown on the left side of the screen (see Figure 4A). Annotators navigate from one word to another in the sentence either by using clicking on the desired word, or by holding down control on the keyboard and navigating with the arrow keys. Within each word, annotators begin by segmenting prefixes and suffixes using either mouse clicks or a keyboard shortcut in Figure 4B. Then a grammatical category is chosen in Figure 4C (in the example, NOUN), after which a set of stem and suffix tags appear in Figure 4D that are applicable for the chosen segmentation and grammatical category. Annotators set tag values either by clicking on them with a mouse and selecting a value from the resulting drop-down list, or else by typing them using a keyboard. For annotators who choose to type, the text is autocompleted for them based on the values that are applicable to that field. Finally, annotators may input Syriac text either by using their mouse to click keys on a virtual keyboard, or by using their keyboard directly in Figure 4E.

Once an annotator changes a field value, that field's background changes color. When correction propagation is active, each time the annotator changes a field, the model is queried for a new prediction constrained by all of the decisions that the annotator has made so far in the sentence. In the scope of the word currently being annotated, if the new pre-annotation differs from the old preannotation, the new value is displayed as a hyperlink to the right of its target field as shown in Figure 4F. For all other words in the sentence, pre-annotation values are simply updated in place.

As annotators proceed, CCASH records detailed information about each word including accuracy, the time each element spent in focus, mouse clicks, and the number of keystrokes. To ensure that timing information is accurate, participants are instructed to press the pause button on the bottom left of Figure 4 whenever they take a break. Whenever the task is paused, the screen is also obscured.

4. Preliminary Analysis

Annotations produced under the same experimental conditions are treated as samples and used to test the various hypotheses of the experiment. In this section, we describe the data and its analysis in more detail.

4.1 The Data

Although participants labeled a sentence at a time, it is problematic to do time analysis on the sentence level because the length of each sentence clearly affects its cost, making annotation time difficult to compare across sentences. Controlling sentence length could alleviate this problem, but introduces a new problem since the length-controlled sentences are not representative of the data as a whole. We avoid these difficulties by doing analysis on the word level.

To estimate word annotation times, we record the time that each word was in focus in the GUI. This time is not a perfect stand-in for the time an annotator spent actually working on each word, since it is possible for an annotator to consider a word that is not actually selected. Also, the first word of each sentence will naturally tend to be selected longer than other words in the sentence as an annotator orients herself by reading the sentence and context. However, given sufficient data, these times should be an acceptable approximation for the true time spent annotating each word.



Figure 5: Box plots representing the data collected so far at each level of pre-annotation. Data generated using correction propagation are not included here.

We compute a word's annotation accuracy by calculating the accuracy of the decisions applicable to the word, as explained in Section 3.2.

The study's 9 participants each annotated 30 sentences, or 152 words, resulting in 1,368 word-level data points both for annotation time and accuracy. Since there are 16 experimental conditions, each condition has roughly 85 data points. Figure 5 uses standard box plots to summarize the data collected under each pre-annotation condition. Corresponding plots for correction propagation are not shown due to space constraints. Notice that for each condition there is considerable variance in both the accuracy of words annotated (5a) and the time required to annotate each word (5b).

4.2 Hypothesis Tests

Our goal is to use data gathered in the study to determine when pre-annotation and correction propagation improve accuracy and increase speed. A simple way of doing this is by comparing the means of various groups of data and testing whether they are significantly different using null hypotheses. We pose three pairs of null hypotheses.

The first pair of null hypotheses is that annotator speed and accuracy are not significantly different for words annotated with and without pre-annotations. Testing these hypotheses at each of the eight pre-annotation accuracy levels indicates when the pre-annotation ought to be used.

The second pair of null hypotheses is that annotator speed and accuracy are not significantly different for words annotated without assistance and those annotated with the *combination* of pre-annotation and correction propagation. Testing this hypothesis at each pre-annotation accuracy level indicates when combined pre-annotation and correction propagation ought to be used.

The third pair of null hypotheses attempts to tease apart the effects of correction propagation and pre- annotation: assuming pre-annotations are being used, annotator speed and accuracy are not significantly different for words annotated with and without correction propagation. Testing this hypothesis at each pre-annotation accuracy level indicates when correction propagation ought to be used above and beyond pre-annotation.

Each null hypothesis is tested using both a standard two-sided Student's t-test as well as a permutation test (Menke, 2004). The Student's t-test is used since it is widely understood and used. A two-sided t-test is appropriate since there is the possibility that accuracy and annotation time will either increase or decrease. The permutation test is used since it does not rely on assumptions about any underlying distribution. Note that with 48 null hypotheses being tested, we expect a few spurious rejections. This can be seen by recalling that if we draw two sets of data from the same process, we expect a standard *t*-test with a p-value threshold of 0.05 to incorrectly reject the null hypothesis one time in twenty. However, if pre-annotation and correction propagation do indeed improve annotator time or accuracy, there should be clear trends in the rejections.

4.3 Results

Table 1 shows the difference between the mean annotator accuracies (a) and times (b) of words annotated under the control condition and of words annotated under the test condition at various levels of pre-annotation quality. Increases in accuracy are good and decreases in time are good. Removing outliers has little effect on the outcomes, so we leave them in for all analyses.

In the first row of Table 1a, which compares the accuracy of words annotated without pre-annotations to those annotated with pre-annotations, there is a clear block of significant results. It appears that pre-annotations generated by models of quality 60% or higher increase average

Control	Test	Pre-annotation model quality (as measured against gold standard)							
Condition	Condition	25%	36%	47%	58%	68%	79%	90%	100%
none	PA	+ 2.6	+ 0.3	+ 2.5	+ <u>5.4</u>	+ 4.8	+ 4.6	+ <u>5.8</u>	+ <u>7.8</u>
none	PA + CP	+ 3.1	+ 2.8	+ 1.9	+ 1.9	+ 3.8	+ 4.7	+ <u>5.4</u>	NA
PA	PA + CP	+ 0.5	+ 2.5	- 0.6	- 3.5	- 1.0	+ 0.1	- 0.4	NA

(a) Change in Mean Word Accuracy

(b) Change in Mean Word Time (sec)

Control	Test	Pre- ann	Pre- annotation model quality (as measured against gold standard)								
Condition	Condition	25%	36%	47%	58%	68%	79%	90%	100%		
none	PA	+ 5.4	- 9.9	+ 11.1	+ 15.4	- 7.1	- <u>20.0</u>	- 10.4	- <u>27.6</u>		
none	PA + CP	- 7.0	+ 5.0	+ 3.1	- 8.0	- 11.2	- 3.9	- 2.9	NA		
PA	PA + CP	- 12.5	+ 14.9	- 8.0	- 23.4	- 4.1	+ 16.1	- 7.5	NA		

Table 1: The difference in the mean accuracy (a) and time (b) of words annotated under two experimental conditions: pre-annotation (PA) and correction propagation (CP). Statistical significance at or below the 0.05 level is indicated by underlining for the two-sided *t*-test and bolding for the permutation test.

annotator accuracy by 5-7%, and that increase is usually greater than can be explained by the natural variance of the data. This is an encouraging result for those contemplating using pre-annotation on similar tasks. Although 60% appears relatively high in the range of model accuracies that we have presented, it is actually quite low for a reasonable predictive model. That is, 60% accurate models can be attained with relatively little data for most tasks (in our case roughly 50 annotated sentences), resulting in a low barrier to entry for those wishing to employ pre-annotation on similar tasks.

The second row in Table 1a shows a similar positive trend for the combination of pre-annotation and correction propagation, but with weaker significance. It is unclear whether this trend is explained entirely by the presence of pre-annotation, or whether correction propagation is playing a role in helping or hurting accuracy. The third row of Table 1a shows mixed signs with no statistical significance, preventing us from drawing any strong conclusions about the effect of correction propagation above and beyond that of pre-annotation.

The first row in Table 1b shows the difference in the mean time required to label words with and without pre-annotations. Pre-annotations generated by models of quality 80% or better decrease average word annotation time by around 10-20 seconds, and that decrease is usually greater than can be explained by the variance in the data, although this trend is still noisy in our current data. Since most words take between 10 and 70 seconds to annotate (see Figure 5b), 10-20 seconds is an appreciable improvement. Pending additional evidence to strengthen the outcome, it is reasonably clear that moderately good pre-annotation reduce the time required for annotation.

One natural way to attempt to anticipate the effect of

Control	Test	Pre-annotation model quality (as measured against gold standard)						
Condition	Condition	25 & 36	36 & 47	47 & 58	58 & 68	68 & 79	79 & 90	90 & 100
none	PA	+ 1.5	+ 1.4	+ 4.1	+ <u>5.1</u>	+ <u>4.7</u>	+ <u>5.2</u>	+ <u>6.8</u>
none	PA + CP	+ 3.0	+ 2.3	+ 1.9	+ 2.9	+ 4.2	+ <u>5.1</u>	+ <u>5.4</u>
PA	PA + CP	+ 1.4	+ 0.9	- 2.3	- 2.3	- 0.5	- 0.2	- 1.5

(a) Change in Mean Word Accuracy

(b) Change	in Mean	Word Time	(sec)
------------	---------	-----------	-------

Control Condition	Test	Pre- anno	old standard)					
Condition	Condition	25 & 36	36 & 47	47 & 58	58 & 68	68 & 79	79 & 90 90 & 10	90 & 100
none	PA	- 1.8	+ 0.8	+ 13.6	+ 4.9	- 12.9	- 14.9	- <u>19.2</u>
none	PA + CP	- 1.1	+ 4.1	- 2.5	- 9.6	- 7.9	- 3.4	- 2.9
PA	PA + CP	+ 0.7	+ 3.3	- 16.1	- 14.5	+ 5.0	+ 11.6	+ 16.3

 Table 2: Identical to Table 1 after grouping instances more coarsely in order to account for current data scarcity.

additional data is to group data points from similar annotation conditions. In Table 2 we do this and test our null hypotheses again. It is worth noting that the results in Table 2 are less applicable to most real world annotation situations than Table 1, since they involve comparing the times and accuracies of words annotated with no pre-annotations (the *none* case) with the times and accuracies of words annotated with a mixture of two different models. However, since the models being mixed are those of similar quality, these results should give us an idea of what our data will look like if present trends continue.

The trends that we noted in Table 1 are slightly clearer in Table 2: both pre-annotation and the combination of pre-annotation and correction propagation reduce annotation time and increase annotation accuracy using low-to-medium quality pre-annotation models. Again, the individual contribution of correction propagation is unclear, although there is some indication in the third row of Table 2b that it may negatively impact annotation speed. It seems safe to say that whether it hurts or helps, the effects of correction propagation on annotator speed and accuracy are dwarfed by the effects of pre-annotation.

Because machine learners improve as additional annotations become available, annotators in large projects will often have access to high quality machine assistance, making the effects of high quality machine of particular interest. Accordingly we asked each participant in the study to annotate two additional randomly selected sentences using what we anticipated would be the most effective experimental condition: 90+C. This yielded an additional 122 word level data points. Adding this new data to Table 1 left the mean accuracy difference between *none* and 90+C unchanged, but changed the mean time difference from -2.9 to -19.0 seconds, and that difference was highly statistically significant. It is likely that additional data would similarly strengthen our other results.

5. Conclusions and Future Work

We have presented a systematic evaluation of pre-annotation and correction propagation together in a controlled user study, providing a detailed data point for those wishing to apply these techniques to similar domains. Preliminary analysis indicates that for our experimental setup, even low quality pre-annotations are effective in increasing average annotator accuracy (i.e. agreement with a gold standard) by 5-7%. Our results also indicate that pre-annotations of moderate quality reduce average annotation time by 10-20 seconds per word. Correction propagation's contribution to annotator speed and accuracy is unclear.

This preliminary analysis will inform continuing work on the creation of the Syriac Electronic Corpus, described in Section 1. As a part of this, we plan to conduct additional analyses of the study's timing data to identify ways of improving the efficiency of user interactions in our GUI. Additionally, we plan to use the timing data collected during the course of the study to model the cost of Syriac morphological annotation so that cost-conscious active learning may be used to reduce the cost of learning high quality pre-annotation models (Haertel et al., 2008). Although active learning shows theoretical promise, there is still a large need for evidence that it can reduce cost in a practical setting.

6. References

- Baldridge, J., & Osborne, M. (2004). Active Learning and the Total Cost of Annotation. In *Proceedings of EMNLP*, 2004, pp. 9-16.
- Brants, T. & Plaehn, O. (2000). Interactive Corpus Annotation. In *Proceedings of LREC 2000*. Athens, Greece.
- Chiou, F., Chiang, D., & Palmer, M. (2001). Facilitating Treebank Annotation with a Statistical Parser. In *Proceedings of the Human Language Technology (HLT) Conference*. San Diego: ACL.
- Felt, P., Merkling, O., Carmen, M., Ringger, E., Lemmon, W., Seppi, K. and Haertel, R. (2010). CCASH: A Web Application Framework for Efficient, Distributed Language Resource Development. In *Proceedings of LREC 2010*. Malta.
- Ganchev, K., Pereira, F., & Mandel, M. (2007). Semi-automated Named Entity Annotation. In *Proceedings of the Linguistic Annotation Workshop*. Prague: ACL, pp. 53-56.
- Haertel, R., Ringger, E., Seppi, K., Carroll, J., and McClanahan, P. (2008). Assessing the Costs of Sampling Methods in Active Learning for Annotation. In *Proceedings of the 46th Annual Meeting of the ACL-HLT*. Columbus, OH: ACL, pp. 65-68.
- Kiraz, G. (1994). Automatic concordance generation of Syriac texts. R. Lavenant, editor. Rome, Italy: VI Symposium Syriacum 1992, Orientalia Christiana Analecta, pp. 461-475.
- Klebanov, B., and Beigman, E. (2009). From Annotator Agreement to Noise Models. *Computational Linguistics* 35(4), pp. 495-503.
- Kristjannson, T., Culotta, A., Viola, P., and McCallum, A. M. (2004). Interactive Information Extraction with Constrained Conditional Random Fields. In Nineteenth Conference of the AAAI. San Jose, CA.
- Marcus, M., Santorini, B., & Marcinkiewicz, M. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics* 19(2), pp. 313-330.
- McClanahan, P., Busby, G., Haertel, R., Heal, K., Lonsdale, D. Seppi, K. and Ringger, E. (2010). A Probabilistic Morphological Analyzer for Syriac. In Proceedings of EMNLP 2010, pp. 810-820.
- Menke, J. and Martinez, T. R. (2004). Using Permutations Instead of Student's t Distribution for P-values in Paired-difference Algorithm Comparisons. In Proceedings of 2004 IEEE International Joint Conference on Neural Networks, pp. 1331-1335.
- Tov, E. editor. The Dead Sea Scrolls Electronic Library. Brill, 2007.
- Wright, William. Apocryphal Acts of the Apostles. 2 vols. London: Williams and Norgate, 1871, vol. 1, pp. 171-333;