Inter-annotator Agreement for Dependency Annotation of Learner Language

Marwa Ragheb
Indiana University
Bloomington, IN USA
mragheb@indiana.edu

Markus Dickinson
Indiana University
Bloomington, IN USA
md7@indiana.edu

Abstract

This paper reports on a study of inter-annotator agreement (IAA) for a dependency annotation scheme designed for learner English. Reliably-annotated learner corpora are a necessary step for the development of POS tagging and parsing of learner language. In our study, three annotators marked several layers of annotation over different levels of learner texts, and they were able to obtain generally high agreement, especially after discussing the disagreements among themselves, without researcher intervention, illustrating the feasibility of the scheme. We pinpoint some of the problems in obtaining full agreement, including annotation scheme vagueness for certain learner innovations, interface design issues, and difficult syntactic constructions. In the process, we also develop ways to calculate agreements for sets of dependencies.

1 Introduction

Learner corpora have been essential for developing error correction systems and intelligent tutoring systems (e.g., Nagata et al., 2011; Rozovskaya and Roth, 2010). So far, error annotation has been the main focus, to the exclusion of corpora and annotation for more basic NLP development, despite the need for parse information for error detection (Tetreault et al., 2010), learner proficiency identification (Hawkins and Buttery, 2010), and acquisition research (Ragheb and Dickinson, 2011). Indeed, there is very little work on POS tagging (Thouësny, 2009; van Rooy and Schäfer, 2002; de Haan, 2000) or parsing (Rehbein et al., 2012; Krivanek and Meurers, 2011; Ott and Ziai, 2010) learner language, and, not coincidentally, there is a lack of annotated data and standards for these tasks. One issue is in knowing how to handle innovative learner forms: some map to a target form before annotating syntax (e.g., Hirschmann et al., 2010), while others propose directly annotating the text (e.g., Ragheb and Dickinson, 2011). We follow this latter strand and further our work towards a syntactically-annotated corpus of learner English by: a) presenting an annotation scheme for dependencies, integrated with other annotation layers, and b) testing the inter-annotator agreement for this scheme. Despite concerns that direct annotation of the linguistic properties of learners may not be feasible (e.g., Rosén and Smedt, 2010), we find that annotators have generally strong agreement, especially after adjudication, and the reasons for disagreement often have as much to do with the complexities of syntax or interface issues as they do with learner innovations.

Probing grammatical annotation can lead to advancements in research on POS tagging and syntactic parsing of learner language, for it shows what can be annotated reliably and what needs additional diagnostics. We specifically report on inter-annotator agreement (IAA) for the annotation scheme described in section 2, focusing on dependency annotation. There are numerous studies investigating inter-annotator agreement between coders for different types of grammatical annotation schemes, focusing on part-of-speech, syntactic, or semantic annotation (e.g., Passonneau et al., 2006; Babarczy et al., 2006; Civit et al., 2003). For learner language, a

Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, pages 169–179,
Atlanta, Georgia, June 13 2013.

number of error annotation projects include measures of interannotator agreement, (see, e.g., Boyd, 2012; Lee et al., 2012; Rozovskaya and Roth, 2010; Tetreault and Chodorow, 2008; Bonaventura et al., 2000), but as far as we are aware, there have been no studies on IAA for grammatical annotation.

We have conducted an IAA study to investigate the quality and robustness of our annotation scheme, as reported in section 3. In section 4, we report quantitative results and a qualitative analysis of this study to tease apart disagreements due to inherent ambiguity or text difficulty from those due to the annotation scheme and/or the guidelines. The study has already reaped benefits by helping us to revise our annotation scheme and guidelines, and the insights gained here should be applicable for future development of other annotation schemes and to parsing studies.

On a final note, our dependency annotation allows for multiple heads for each token in the corpus, violating the so-called *single-head constraint* (Kübler et al., 2009). In the process of evaluating these dependencies (see section 4.1), we also make some minor contributions towards comparing sets of dependencies, moving beyond just F-measure (e.g., Cer et al., 2010) to account for partial agreements.

2 Annotation scheme

We present a sketch of the annotation scheme here, outlining the layers and the general motivation. Our general perspective is to annotate as closely as possible to what the learner wrote, marking grammatical properties even if the meaning of the sentence or clause is unclear within the particular grammatical analysis. For example, in the learner sentence (1), the verb *admit* clearly occurs in the form of an active verb, and is annotated as such, regardless of the (passive) meaning of the sentence (cf. *was admitted*). In this case, basing the annotation on syntactic evidence makes for a more straightforward task. Moreover, adhering to a syntactic analysis helps outline the grammatical properties of a learner's interlanguage and can thus assist in automatic tasks such as native language identification (e.g., Tetreault et al., 2012), and proficiency level determination (Yannakoudakis et al., 2011).

(1) When I admit to Korea University, I decide ...

Another part of the motivation for shying away from marking target forms and annotating the syntactic properties of those (cf., e.g., Rehbein et al., 2012) is that, for general essays from learners of many levels, the grammatical evidence can be understood even when the intended meaning is not. Consider (2): in the context of the learner's essay, the sentence probably means that this person guards their personal belongings very well because of prevalent theft in the city they are talking about.

(2) Now I take very hard my personal stuffs.

Annotating the syntax of a target form here could obscure the grammatical properties of the learner's production (e.g., pluralizing a mass noun). Encouraging annotators to focus on the syntactic properties and not intended meanings makes identifying the dependency relations in a sentence like this one easy.

Another aspect of our annotation scheme is that we do not directly annotate errors (except for lexical violations; see section 2.1). Annotators had access to an extensive manual detailing the annotation scheme, which will be made public soon.[1] A brief outline of the guidelines is in section 3.3.

2.1 Initial annotation layers

Using ideas developed for annotating learner language (Ragheb and Dickinson, 2012, 2011; Díaz-Negrillo et al., 2010; Dickinson and Ragheb, 2009), we annotate several layers before targeting dependencies: 1) lemmas (i.e., normalized forms), 2) morphological part-of-speech (POS), 3) distributional POS, and 4) lexical violations.

The idea for **lemma** annotation is to normalize a word to its dictionary form. In (3), for example, the misspelled *excersice* is normalized to the correctly-spelled *exercise* for the lemma annotation. We specify that only "reasonable" orthographic or phonetic changes are allowed; thus, for *prison*, it is lemma-annotated as *prison*, not *person*. In this case, the lemma annotation does not affect the rest of the annotation, as *prison* and *person* are both nouns, but for *no*, the entire analysis changes based on whether we annotate the lemma as *no* or *not*. Marking *no* makes the final tree more difficult, but fits with the principle of staying true to the form the learner has

[1] See: `http://cl.indiana.edu/~salle`

presented. As we will see in section 4.3, determining the lemma can pose challenges for building trees.

(3) After to start , I want to tell that this **excersice** is very important in the life , **no** only as a **prison** .

We annotate two POS layers, one capturing **morphological** evidence and one for **distributional**. For most words, the layers include the same information, but mismatches arise with non-canonical structures. For instance, in (3) the verb *(to) start* has a morphological POS of base form verb (VV0), but it appears in a context where some other verb form would better be licensed, e.g., a gerund. Since we do not want to overstate claims, we allow for underspecified POS tags and annotate the distributional POS simply as verb (VV). The use of two POS layers captures the mismatch between morphology and distribution without referencing a unified POS.

Finally, annotators can mark **lexical violations** when nothing else appears to capture a non-standard form. Specifically, lexical violations are for syntactically ungrammatical forms where the specific word choice seems to cause the ungrammaticality. In (4), for example, *about* should be marked as a lexical violation. Lexical violations were intended as a last resort, but as we will see in section 4.3, there was confusion about when to use lexical violations and when to use other annotations, e.g., POS mismatches.

(4) ... I agree **about** me that my country 's help and cooperation influenced . . .

2.2 Dependencies

While the initial annotation layers are used to build the syntactic annotation, the real focus of the annotation concerns dependencies. Using a set of 45 dependencies,[2] we mark two types of annotations here: 1) dependency relations rooted in the lemma and the morphological POS tag, and 2) subcategorization information, reflecting not necessarily what is in the tree, but what is required. Justification for a morphological, or morphosyntactic, layer of dependencies, along with a layer of subcategorization, is given in Ragheb and Dickinson (2012). Essentially, these two layers allow one to capture issues involving argument structure (e.g., missing argument), without having to make the kind of strong claims a layer of distributional dependencies would require. In (5), for example, *wondered* subcategorizes for a finite complement (COMP), but finds a non-finite complement (XCOMP), as the tree is based on the morphological forms (e.g., *to*).

(5) I wondered what success to be .

An example tree is shown in figure 1, where we can see a number of properties of our trees: a) we annotate many "raised" subjects, such as *I* being the subject (SUBJ) of both *would* and *like*, thereby allowing for multiple heads for a single token; b) we ignore semantic anomalies, such as the fact that *life* is the subject of *be (successful)*; and c) dependencies can be selected for, but not realized, as in the case of *career* subcategorizing for a determiner (DET).

3 Inter-annotator agreement study

3.1 Selection of annotation texts

From a learner corpus of written essays we have collected from students entering Indiana University, we chose a topic (*What Are Your Plans for Life?*) and randomly selected six essays, based on both learner proficiency (beginner, intermediate, advanced) and the native language of the speaker (L1).[3] From each essay, we selected the first paragraph and put the six paragraphs into two texts; each text contained, in order, one beginner, one intermediate, and one advanced paragraph. Text 1 contained 19 sentences (333 tokens), and Text 2 contained 22 sentences (271 tokens). Annotators were asked to annotate only these excerpts, but had access to the entire essays, if they wanted to view them.

While the total number of tokens is only 604, the depth of the annotation is quite significant, in that there are at least seven decisions to be made for every token: lemma, lexical violation, morphological POS, distributional POS, subcategorization, attachment, and dependency label, in addition to possible extra dependencies for a given word, i.e., a few thousand decisions. It is hard to quantify the effort, as some layers are automatically pre-annotated (see section 3.5) and some are used sparingly (lexical violations), but we estimate around 2000 new or changed annotations from each annotator.

[2] We use a label set adapted from Sagae et al. (2010).

[3] Korean, Spanish, Chinese, Arabic, Japanese, Hungarian.

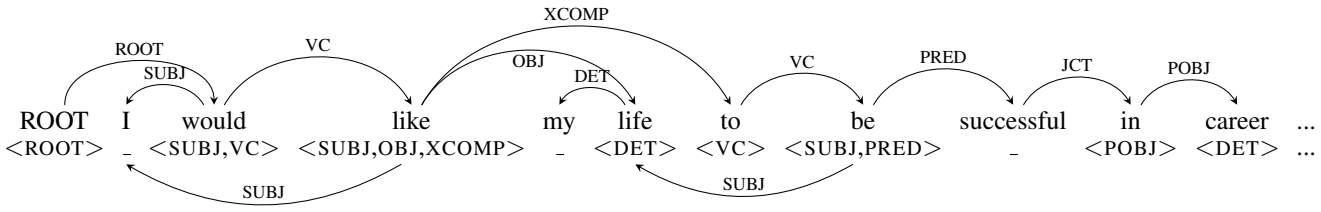

Figure 1: Morphosyntactic dependency tree with subcategorization information

3.2 Annotators

This study involved three annotators, who were undergraduate students at Indiana. They were native speakers of English and majors in Linguistics (2 juniors, 1 senior). Two had had a syntax course before the semester, and one was taking it concurrently. We trained them over the course of an academic semester (fall 2012), by means of weekly meetings to discuss relevant readings, familiarize them with the scheme, and give feedback about their annotation. The IAA study took place Nov. 9–Dec. 15.

Annotators were taking course credit for participating in this project. This being the case, they were encouraged to learn from the experience, and part of their training was to make notes of challenging cases and their decision-making process. This has provided significant depth in qualitatively analyzing the IAA outcomes (section 4.3).

3.3 Guidelines

At the start of the study, the annotators were given a set of guidelines (around 100 pages) to reference as they made decisions. These guidelines outline the general principles of the scheme (e.g., give the learner the benefit of the doubt), an overview of the annotation layers, and annotation examples for each layer. The guidelines refer to the label sets used for POS (Sampson, 1995) and dependencies (Sagae et al., 2010), but emphasize the properties of our scheme. Although the guidelines discuss general syntactic treatment (e.g., "attach high" in the case of attachment ambiguities), a considerable focus is on handling learner innovations, across different layers. While we cannot list every example of how learners innovate, we include instructions and examples that should generalize to other non-native constructions (e.g., when to underspecify a label). Examples of how to treat difficult syntactic constructions are also illustrated (e.g., coordination).

	Text 1				Text 2			
	Time	Avg.	Min.	Max.	Time	Avg.	Min.	Max.
A	224	11.8	3	25	151	6.9	2	21
B	280	14.7	4	30	170*	8.5	3	20
C	480	25.3	8	60	385	17.5	10	45

Table 1: Annotation time, in minutes, for phase 1 (*times for two sentences were not reported and are omitted)

3.4 Annotation task

Via oral and written instructions, the annotators were asked to independently annotate the two texts and take notes on difficult issues, in addition to marking how long they spent on each sentence. Times are reported in table 1 for the first phase, as described next. Longer sentences take more time (cf. Text 1 vs. Text 2), and annotator times vary, but, given the times of nearly 30–60 minutes per sentence at the start of the semester, these times seemed reasonable for the depth of annotation required.

The annotation task proceeded in phases. **Phase 1:** Text 1 was annotated over the course of one week, and Text 2 over the next week. **Phase 2:** After an hour-long meeting with annotators covering general annotation points that seemed to be problematic (e.g., lemma definitions), they were given another week to individually go over their annotations and make modifications. At the meeting, nothing about the scheme or guidelines was added, and no specific examples from the data being annotated were used (only ones from earlier in the semester). **Phase 3:** Each annotator received a document pointing out pairwise disagreements between annotators, in a simple textual format like (6). Each annota-

tor was asked to use this document and make any changes where they thought that their analysis was not the best one, given the other two. This process took approximately a week. **Phase 4:** The annotators met (for three hours) and discussed remaining differences, to see whether they could reach a consensus. Each annotator fixed their own file based on the results of this discussion. At each point, we took a snapshot of the data, but at no point did we provide feedback to the annotators on their decisions.

(6) Sentence 2, word 1: relation ... JCT NJCT

3.5 Annotation interface

The annotation is done via the Brat rapid annotation tool (Stenetorp et al., 2012).[4] This online interface, shown in figure 2, allows an annotator to drag an arrow between words to create a dependency. Annotators were given automatically-derived POS tags from TnT (Brants, 2000), trained on the SUSANNE corpus (Sampson, 1995), but created the dependencies from scratch.[5] Subcategorizations, lemmas, and lexical violations are annotated within one of the POS layers; lemmas are noted by the blue shading, and the presence of other layers is noted by asterisks, an interface point discussed in section 4.2.3. Annotators liked the tool, but complained of its slowness.

4 Evaluation

4.1 Methods of comparison

For lemma and POS annotation, we can calculate basic agreement statistics, as there is one annotation for each token. But our primary focus is on subcategorization and dependency annotation, where there can be multiple elements (or none) for a given token.

For subcategorization, we treat elements as members of a set, as annotators were told that order was unimportant (e.g., <SUBJ,OBJ> = <OBJ,SUBJ>); we discuss metrics for this in section 4.1.1. For dependencies, we adapt standard parse evaluation (see Kübler et al., 2009, ch. 6). In brief, **unlabeled attachment agreement (UAA)** measures the number of attachments annotators agree upon for each token, disregarding the label, whereas **labeled attachment agreement (LAA)** requires both the attachment and labeling to be the same to count as an agreement. **Label only agreement (LOA)** ignores the head a token attaches to and only compares labels.

All three metrics (UAA, LAA, LOA) require calculations for *sets* of dependencies, described in sections 4.1.1 and 4.1.2. In figure 3, for instance, one annotator (accidentally) drew a JCT arrow in the wrong direction, resulting in two heads for *is*. For *is*, the annotator's set of dependencies is {(0,ROOT), (1,JCT)}, compared to another's of {(0,ROOT)}. We thus treat dependencies as sets of (head, label) pairs.

4.1.1 Metrics

For sets, we use two different calculations. First is **MASI** (Measuring Agreement on Set-valued Items, Passonneau et al., 2006), which assigns each comparison between sets a value between 0 and 1, assigning partial credit for partial set matches and allowing one to treat agreement on a per-token basis. We use a simplified form of MASI as follows: 1 = identical sets, $\frac{2}{3}$ = one set is a subset of the other, $\frac{1}{3}$ = the intersection of the sets is non-null, and so are the set differences, & 0 = disjoint sets.[6]

The second method is a global comparison method (**GCM**), which counts all the elements in each annotator's sets in the whole file and counts up the total number of agreements. In the following subcategorization example over three tokens, there are two agreements, compared to four total elements used by A1 ($\text{GCM}_{A1} = \frac{2}{4}$) and compared to three elements used by A2 ($\text{GCM}_{A2} = \frac{2}{3}$). These metrics are essentially precision and recall, depending upon which annotator is seen as the "gold" (Kübler et al., 2009, ch. 6). For MASI scores, we have 0, 1, and $\frac{1}{3}$, respectively, giving $1\frac{1}{3}/3$, or 0.44.

- A1: {SUBJ}, A2: {}
- A1: {SUBJ}, A2: {SUBJ}
- A1: {SUBJ,PRED}, A2: {SUBJ,OBJ}

Since every word is annotated, the methods assign similar numbers for dependencies. Subcategorization gives different results, due to empty sets. If annotator 1 and annotator 2 both mark an empty set,

[4] http://brat.nlplab.org

[5] Annotators need to provide the dependency annotations since we lacked an appropriate L2 parser. It is a goal of this project to provide annotated data for parser development.

[6] Since our sets tend to be small (rarely bigger than two), we do not expect much change with a full MASI calculation.

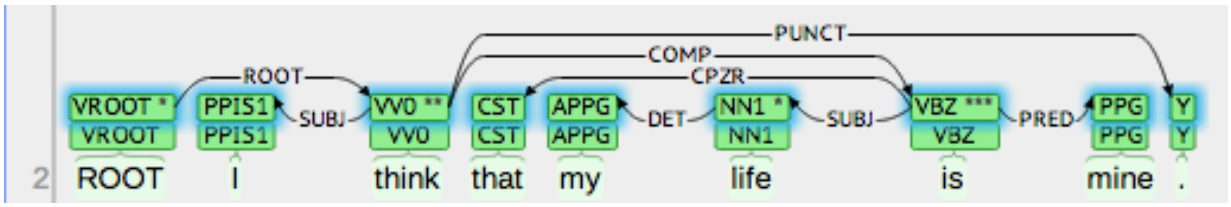

Figure 2: Example of the annotation interface

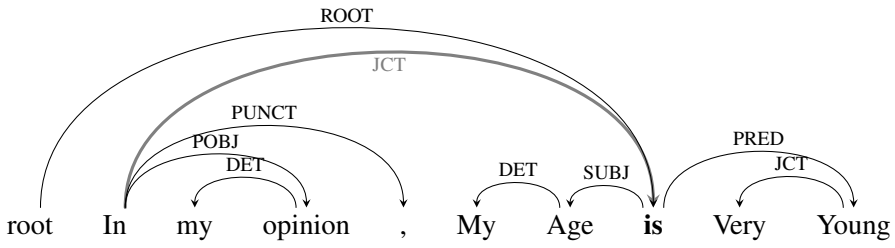

Figure 3: A mistaken arrow (JCT) leading to two dependencies for *is* ((0,ROOT),(1,JCT))

we count full agreement for MASI, i.e., a score of 1; for GCM, nothing gets added to the totals.

We could, of course, report various coefficients commonly used in IAA studies, such as kappa or alpha (see Artstein and Poesio, 2008), but, given the large number of classes and lack of predominant classes, chance agreement seems very small.

4.1.2 Dependency-specific issues

As a minor point: for dependencies, we calculate agreements for matches in only attachment or labeling. Consider (7), where there is one match only in attachment ((24,OBJ)-(24,JCT)), counting towards UAA, and one only in labeling ((24,SUBJ)-(22,SUBJ)) for LOA. Importantly, we have to ensure that (24,SUBJ) and (24,JCT) are not linked.

(7) A1: {(24,SUBJ), (24,OBJ)}
A2: {(22,SUBJ), (24,JCT)}

In general, we prioritize identical attachment over labeling, if a dependency could match in either. We wrote a short script to align attachment/label matches between two sets, but omit details here, due to space. We generally do not have large sets of dependencies to compare, but these technical decisions should allow for any situation in the future.

4.2 Results

4.2.1 Bird's-eye view

Table 2 presents an overview of pairwise agreements between annotators for all 604 tokens. Of the four phases of annotation, we report two: the files they annotated (and revised) independently (phase 2) and the final files after discussion of problematic cases (phase 4). Annotators reported feeling rushed during phase 1, so phase 2 numbers likely better indicate the ability to independently annotate, and phase 4 can help to investigate the reasons for lingering disagreements. The numbers for subcategorization and dependency (UAA, LAA) agreements are the MASI agreement rates.

A few observations are evident from these figures. First, for both POS_m (morphology) and POS_d (distribution), the high agreement rates reflect the fact that annotators made very few changes to the automatic pre-annotation, partly because such layers were not heavily emphasized. Lemmas were also pre-annotated, as identical to the surface form, but more changes were made here (decapitalization, affix-stripping, etc.). Comparing phases 2 and 4 shows an improvement in agreement, although agreement seems like it could be higher, given the simplicity of lemma information. We discuss lemmas, and associated lexical violations, more in sec-

Annotators	lemma		POS_m		POS_d		Subcat.		UAA		LAA	
	P2	P4	P2	P4	P2	P4	P2	P4	P2	P4	P2	P4
A, B	93.4	96.9	99.0	98.7	99.2	98.7	85.5	94.0	86.6	97.0	80.0	95.2
B, C	94.4	97.7	99.0	99.5	98.7	99.3	86.1	95.7	86.7	97.1	80.3	96.0
C, A	92.4	96.9	99.7	99.7	98.5	99.3	86.1	96.6	86.9	97.7	82.4	96.7

Table 2: Overview of agreement rates before & after discussion (phases 2 & 4)

tion 4.3.

Dependency-related annotations had no pre-annotation. While the starting value of agreement rates for these last three layers is not as high as for lemma and POS annotation, agreement rates around 80–85% still seem moderately high. More important is how much the agreement rates improved after discussion, achieving approximately 95% agreement. This was without any direct intervention from the researchers regarding how to annotate disagreements. We examine dependencies in section 4.2.2 and subcategorization in 4.2.3, breaking results down by text to see differences in difficulty.

4.2.2 Dependencies

We report MASI agreement rates for dependencies in tables 3 and 4 for Text 1 and Text 2, respectively.[7] Comparing the starting agreement values (e.g., 73.6% vs. 87.8% LAA for annotators A and B), it is clear that text difficulty had an enormous impact on annotator agreement. The clear difference in tokens per sentence (17.5 in Text 1 vs. 12.3 in Text 2; see section 3.1) contributed to the differences. The reported difficulty from annotators referred to more non-native properties present in the text, and, to a smaller extent, the presence of more complex syntactic structures. Though we take up some of these issues up again in section 4.3, an in-depth analysis of how text difficulty affects the annotation task is beyond the scope of this paper, and we leave it for future investigation.

Looking at the agreement rates for Text 1 in table 3, we can see that the initial rates of agreement for UAA and LOA are moderately high, indicating that annotator training and guideline descriptions were working moderately well. However, they are only 73% for LAA. Note, though, that this may be more related to issues of fatigue and hurry than of understanding of the guidelines: the numbers improve considerably by phase 4. The labeled attachment rates, for example, increase between 17 and 21 percent, to reach values around 95%.

[7]We only report MASI scores for dependencies, since the GCM scores are nearly the same. For example, for raters A & B, the GCM value for phase 4 is 96.15% with respect to either annotator vs. 96.10% for MASI.

Ann.	UAA		LAA		LOA	
	P2	P4	P2	P4	P2	P4
A, B	81.8	96.1	73.6	93.4	80.3	95.5
B, C	80.9	96.2	73.4	94.4	79.3	97.1
A,C	83.6	97.6	79.7	96.7	81.8	97.9

Table 3: MASI percentages for dependencies, Text 1

Ann.	UAA		LAA		LOA	
	P2	P4	P2	P4	P2	P4
A, B	92.6	98.1	87.8	97.4	89.3	97.8
B, C	93.8	98.3	88.7	97.9	90.2	98.6
A, C	90.9	97.9	85.7	96.8	87.6	97.9

Table 4: MASI percentages for dependencies, Text 2

For Text 2 in table 4, we notice again the higher phase 2 rates and the similar improvement in phase 4, with LAA around 97%. Encouragingly, despite the initially lower agreements for Text 1, annotators were able to achieve nearly the same level of agreement as for the "easier" text. This illustrates that annotators can learn the scheme, even for difficult sentences, though there may be a tradeoff between speed and accuracy.

4.2.3 Subcategorization

For subcategorization, we present both MASI and GCM percentage rates, as they give different emphases. Results are again broken down by text, in tables 5 and 6. As with dependencies, we see solid improvement from phase 2 to phase 4, and we see

generally higher agreement for Text 2.

Ann.	MASI		GCM_1		GCM_2	
	P2	P4	P2	P4	P2	P4
A,B	84.3	92.4	81.9	90.8	72.8	88.1
B,C	83.6	93.8	74.4	91.6	73.6	90.2
A,C	84.9	96.1	83.0	96.4	73.1	92.2

Table 5: Agreement rates for subcategorization, Text 1

Ann.	MASI		GCM_1		GCM_2	
	P2	P4	P2	P4	P2	P4
A,B	87.1	95.9	88.9	96.0	77.2	94.1
B,C	89.3	98.0	88.3	98.0	82.0	96.8
A,C	87.6	97.2	91.2	97.3	73.7	94.2

Table 6: Agreement rates for subcategorization, Text 2

The GCM numbers are much lower because of the way empty subcategorization values are handled—being counted towards agreement for MASI and not for GCM (see section 4.1.1). A further issue, though, is that one annotator often simply left out subcategorization annotation for a token. In table 6, for example, annotators A and C have vastly different GCM values for phase 2 (91.2% vs. 73.7%), due to annotator C annotating many more subcategorization labels. This is discussed more in section 4.3.2.

4.3 Qualitative differences

We highlight some of the important issues that stand out when we take a closer look at the nature of the disagreements in the final phase.

4.3.1 Text-related issues

As pointed out earlier regarding the differences between Text 1 and Text 2 (section 4.2.2), some disagreements are likely due to the nature of the text itself, both because of its non-native properties and because of the syntactic complexity. Starting with unique learner innovations leading to non-uniform treatment, several cases stemmed from not agreeing on the lemma, when a word looks non-English or does not fit the context. An example is *cares* in (8): although the guidelines should lead the annotators to choose *care* as the lemma, staying true to the learner form, one annotator chose to accommodate the context and changed the lemma to *case*. This relying too heavily on intended meaning and not enough on syntactic evidence—as the scheme is designed for—was a consistent problem.

(8) My majors are bankruptcy , corporate reorganizations . . . and arquisisiton **cares** .

For (8), the trees do not change because the different lemmas are of the same syntactic category, but more problematic are cases where the trees differ based on different readings. In the learner sentence (9), the non-agreement between *this* and *cause* led to a disagreement of *this* being a COORD of *and* vs. *this* being an APPOS (appositive) of *factors*. The annotator reported that the choice for this latter analysis came from treating *this* as *these*, again contrary to guidelines but consistent with one meaning.

(9) Sometimes animals are subjected to changed environmental factors during their developmental process and **this** cause FA .

Another great source of disagreement stems from the syntactic complexity of some of the structures, even if native-like, though this can be intertwined with non-native properties, as in (10). Although annotators eventually agreed on the annotation here, there was initial disagreement on the coordination structure of this sentence, questioning whether *to be* coordinates with *pursuing* or only with *to earn*, or whether *pursuing* coordinates only with *to earn* (the analysis they finally chose).

(10) My most important goals are **pursuing** the profession **to be** a top marketing manager and then **to earn** a lot of money to buy a beautiful house and a good car .

4.3.2 Task-related issues

Annotator disagreements stemmed not only from the text, but from other factors as well, such as aspects of the scheme that needed more clarification, some interface issues, and the fact that the guidelines though extensive, are still not comprehensive.

A few parts of the annotation scheme were confusing to annotators and likely need refinement. For example, if the form of a word was incorrect, we saw a lot of lexical violation annotation, even if it

was only an issue of grammatical marking and POS (e.g., *did*/VVD instead of *done*/VVN), as opposed to a truly different word choice. We are currently tightening the annotation scheme and adding clarifications about lexical violations in our guidelines.

As another example, verb raising was often not marked (cf. figure 1), in spite of the scheme and guidelines requiring it. In their comments, annotators mentioned that it seemed "redundant" to them and that it caused arcs to cross, which they found "unappealing." One annotator commented that they did not have enough syntactic background to see why marking multiple subjects was necessary. We are thus considering a simpler treatment. Another option in the future is to hire annotators with more background in syntax.

The interface may be partly to blame for some disagreements, including subcategorizations which annotators often left unmarked (section 4.2.3) or only partly marked (e.g., leaving off a SUBJect for a verb which has been raised). There are a few reasons for this. First, marking subcategorization likely needed more emphasis in the training period, seeing as how it relates to complicated linguistic notions like distinguishing arguments and adjuncts. Secondly, the interface is an issue, as the subcategorization field is not directly visible, compared to the arcs drawn for dependencies; in figure 2, for instance, subcategorization can only be seen in the asterisks, which need to be clicked on to be seen and changed. Relatedly, because it is not always necessary, subcategorization may seem more optional and thus forgettable.

By the nature of being an in-progress project, the guidelines were necessarily not comprehensive. As one example, the TRANS(ition) label was only generally defined, leading to disagreements. As another, a slash could indicate coordination (*actor/actress*), and annotators differed on its POS labeling, as either CC (coordinating conjunction), or a PUNCT (punctuation). The different POS labels then led to vastly different dependency graphs. In spite of a lengthy section on how to handle coordination in the guidelines, it seems that an additional case needs to be added to the guidelines to cover when punctuation is used as a conjunction.

5 Conclusion and outlook

Developing reliable annotation schemes for learner language is an important step towards better POS tagging and parsing of learner corpora. We have described an inter-annotator agreement study that has helped shed light on several issues, such as the reliability of our annotation scheme, and has helped identify room for improvement. This study shows that it is possible to apply a multi-layered dependency annotation scheme to learner text with considerably good agreement rates between three trained annotators. In the future, we will of course be applying the (revised) annotation scheme to larger data sets, but we hope other grammatical annotation schemes can learn from our experience. In the shorter term, we are constructing a gold standard of the text files used here, to test annotation accuracy and whether any (or all) annotators had consistent difficulties. Another next step is to gather a larger pool of data and focus more on analyzing the effects of L1 and learner proficiency level on annotation. Finally, given that syntactic representations can assist in automating tasks such as developmental profiling of learners (e.g., Vyatkina, 2013), grammatical error detection (Tetreault et al., 2010), identification of native language (e.g., Tetreault et al., 2012), and proficiency level determination (Dickinson et al., 2012)—all of which impact NLP-based educational tools—one can explore the effect of specific syntactic decisions on such tasks, as a way to provide feedback on the annotation scheme.

Acknowledgments

We would like to thank the three annotators for their help with this experiment. We also thank the IU CL discussion group, as well as the three anonymous reviewers, for their feedback and comments.

References

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

Anna Babarczy, John Carroll, and Geoffrey Sampson. 2006. Definitional, personal, and mechanical constraints on part of speech annotation performance. *Natural Language Engineering*, 12:77–90.

Patrizia Bonaventura, Peter Howarth, and Wolfgang Menzel. 2000. Phonetic annotation of a non-native speech corpus. In *Proceedings International Workshop on Integrating Speech Technology in the (Language) Learning and Assistive Interface, InStil*, pages 10–17.

Adriane Amelia Boyd. 2012. *Detecting and Diagnosing Grammatical Errors for Beginning Learners of German: From Learner Corpus Annotation to Constraint Satisfaction Problems*. Ph.D. thesis, Ohio State University.

Thorsten Brants. 2000. TnT – a statistical part-of-speech tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP 2000)*, pages 224–231. Seattle, WA.

Daniel Cer, Marie-Catherine de Marneffe, Daniel Jurafsky, and Christopher D. Manning. 2010. Parsing to Stanford dependencies: Trade-offs between speed and accuracy. In *Proceedings of LREC-10*. Malta.

M. Civit, A. Ageno, B. Navarro, N. Bufí, and M. A. Martí. 2003. Qualitative and quantitative analysis of annotators' agreement in the development of Cast3LB. In *Proceedings of 2nd Workshop on Treebanks and Linguistics Theories (TLT-2003)*, pages 33–45.

Pieter de Haan. 2000. Tagging non-native English with the TOSCA-ICLE tagger. In Christian Mair and Markus Hundt, editors, *Corpus Linguistics and Linguistic Theory*, pages 69–79. Rodopi, Amsterdam.

Ana Díaz-Negrillo, Detmar Meurers, Salvador Valera, and Holger Wunsch. 2010. Towards interlanguage POS annotation for effective learner corpora in SLA and FLT. *Language Forum*, 36(1–2):139–154. Special Issue on New Trends in Language Teaching.

Markus Dickinson, Sandra Kübler, and Anthony Meyer. 2012. Predicting learner levels for online exercises of Hebrew. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 95–104. Association for Computational Linguistics, Montréal, Canada.

Markus Dickinson and Marwa Ragheb. 2009. Dependency annotation for learner corpora. In *Proceedings of the Eighth Workshop on Treebanks and Linguistic Theories (TLT-8)*, pages 59–70. Milan, Italy.

John A. Hawkins and Paula Buttery. 2010. Criterial features in learner corpora: Theory and illustrations. *English Profile Journal*, 1(1):1–23.

Hagen Hirschmann, Anke Lüdeling, Ines Rehbein, Marc Reznicek, and Amir Zeldes. 2010. Syntactic overuse and underuse: A study of a parsed learner corpus and its target hypothesis. Talk given at the Ninth Workshop on Treebanks and Linguistic Theory.

Julia Krivanek and Detmar Meurers. 2011. Comparing rule-based and data-driven dependency parsing of learner language. In *Proceedings of the Int. Conference on Dependency Linguistics (Depling 2011)*. Barcelona.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. *Dependency Parsing*. Morgan & Claypool Publishers.

Sun-Hee Lee, Markus Dickinson, and Ross Israel. 2012. Developing learner corpus annotation for Korean particle errors. In *Proceedings of the Sixth Linguistic Annotation Workshop*, LAW VI '12, pages 129–133. Association for Computational Linguistics, Stroudsburg, PA, USA.

Ryo Nagata, Edward Whittaker, and Vera Sheinman. 2011. Creating a manually error-tagged and shallow-parsed learner corpus. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1210–1219. Portland, OR.

Niels Ott and Ramon Ziai. 2010. Evaluating dependency parsing performance on German learner language. In *Proceedings of TLT-9*, volume 9, pages 175–186.

Rebecca Passonneau, Nizar Habash, and Owen Rambow. 2006. Inter-annotator agreement on a multilingual semantic annotation task. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 1951–1956.

Marwa Ragheb and Markus Dickinson. 2011. Avoiding the comparative fallacy in the annotation of learner corpora. In *Selected Proceedings of the 2010 Second Language Research Forum: Re-*

considering SLA Research, Dimensions, and Directions, pages 114–124. Cascadilla Proceedings Project, Somerville, MA.

Marwa Ragheb and Markus Dickinson. 2012. Defining syntax for learner language annotation. In *Proceedings of the 24th International Conference on Computational Linguistics (Coling 2012), Poster Session*. Mumbai, India.

Ines Rehbein, Hagen Hirschmann, Anke Lüdeling, and Marc Reznicek. 2012. Better tags give better trees - or do they? *Linguistic Issues in Language Technology (LiLT)*, 7(10).

Victoria Rosén and Koenraad De Smedt. 2010. Syntactic annotation of learner corpora. In Hilde Johansen, Anne Golden, Jon Erik Hagen, and Ann-Kristin Helland, editors, *Systematisk, variert, men ikke tilfeldig. Antologi om norsk som andrespråk i anledning Kari Tenfjords 60-årsdag [Systematic, varied, but not arbitrary. Anthology about Norwegian as a second language on the occasion of Kari Tenfjord's 60th birthday]*, pages 120–132. Novus forlag, Oslo.

Alla Rozovskaya and Dan Roth. 2010. Annotating ESL errors: Challenges and rewards. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 28–36. Los Angeles, California.

Kenji Sagae, Eric Davis, Alon Lavie, and Brian MacWhinney an Shuly Wintner. 2010. Morphosyntactic annotation of childes transcripts. *Journal of Child Language*, 37(3):705–729.

Geoffrey Sampson. 1995. *English for the Computer: The SUSANNE Corpus and Analytic Scheme*. Clarendon Press, Oxford.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Avignon, France.

Joel Tetreault, Daniel Blanchard, Aoife Cahill, and Martin Chodorow. 2012. Native tongues, lost and found: Resources and empirical evaluations in native language identification. In *Proceedings of COLING 2012*, pages 2585–2602. Mumbai, India.

Joel Tetreault and Martin Chodorow. 2008. Native judgments of non-native usage: experiments in preposition error detection. In *Proceedings of the Workshop on Human Judgements in Computational Linguistics*, HumanJudge '08, pages 24–32. Association for Computational Linguistics, Stroudsburg, PA, USA.

Joel Tetreault, Jennifer Foster, and Martin Chodorow. 2010. Using parse features for preposition selection and error detection. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 353–358. Uppsala, Sweden.

Sylvie Thouësny. 2009. Increasing the reliability of a part-of-speech tagging tool for use with learner language. Presentation given at the Automatic Analysis of Learner Language (AALL'09) workshop on automatic analysis of learner language: from a better understanding of annotation needs to the development and standardization of annotation schemes.

Bertus van Rooy and Lande Schäfer. 2002. The effect of learner errors on POS tag errors during automatic POS tagging. *Southern African Linguistics and Applied Language Studies*, 20:325–335.

Nina Vyatkina. 2013. Specific syntactic complexity: Developmental profiling of individuals based on an annotated learner corpus. *The Modern Language Journal*, 97(S1):1–20.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189. Portland, OR.