Constructing a Parser Evaluation Scheme

Laura Rimell and Stephen Clark
Oxford University Computing Laboratory
Wolfson Building, Parks Road
Oxford, OX1 3QD, United Kingdom
{laura.rimell,stephen.clark}@comlab.ox.ac.uk

Abstract

In this paper we examine the process of developing a relational parser evaluation scheme, identifying a number of decisions which must be made by the designer of such a scheme. Making the process more modular may help the parsing community converge on a single scheme. Examples from the shared task at the COLING parser evaluation workshop are used to highlight decisions made by various developers, and the impact these decisions have on any resulting scoring mechanism. We show that quite subtle distinctions, such as how many grammatical relations are used to encode a linguistic construction, can have a significant effect on the resulting scores.

1 Introduction

In this paper we examine the various decisions made by designers of parser evaluation schemes based on grammatical relations (Lin, 1995; Carroll et al., 1998). Following Carroll et al. (1998), we use the term *grammatical relations* to refer to syntactic dependencies between heads and dependents. We assume that grammatical relation schemes are currently the best method available for parser evaluation due to their relative independence of any particular parser or linguistic theory. There are several grammatical relation schemes currently available, for example Carroll et al. (1998), King et al. (2003), and de Marneffe et al. (2006). However, there has been little analysis of the decisions made by the designers in creating what turns out to be a complex set of dependencies for naturally occurring sentences. In particular, in this paper we consider how the process can be made more modular to help the parsing community converge on a single scheme.

The first decision to be made by the scheme designer is what types of linguistic constructions should be covered by the scheme. By *construction* we mean syntactic phenomena such as subject of verb, direct object of verb, passive voice, coordination, relative clause, apposition, and so on. In this paper we assume that the constructions of interest have already been identified (and there does appear to be broad agreement on this point across the existing schemes). A construction can be thought of as a unitary linguistic object, although it is often represented by several grammatical relations.

The second decision to be made is which words are involved in a particular construction. This is important because a subset of these words will be arguments of the grammatical relations representing the construction. Again, we assume that there is already broad agreement among the existing schemes regarding this question. One possible point of disagreement is whether to include empty elements in the representation, for example when a passive verb has no overt subject, but we will not address that issue here.

The next question, somewhat orthogonal to the previous one, and a source of disagreement between schemes, is how informative the representation should be. By *informative* we mean the amount of linguistic information represented in the scheme. As well as relations between heads, some schemes include one or more features, each of which expresses information about an individual head. These features can be the locus of richer linguistic information than is represented in the dependencies. A useful example here is tense and mood information for verbs. This is included in the PARC scheme, for example, but not in the Briscoe and Carroll or Stanford schemes; PARC is in general more informative and detailed than competing schemes. Although features are technically different from relations, they form part of an overall evaluation scheme and must be considered by the scheme designer. We will not consider here the question of how informative schemes should be; we only note the importance of this question for the resulting scoring mechanism.

The penultimate question, also a source of disagreement among existing schemes, is which words among all those involved in the construction should be used to represent it in the scheme. This decision may arise when identifying syntactic heads; for example, in the sentence *Brown said house prices will continue to fall*, we assume there is no disagreement about which words are involved in the clausal complement construction ({*said, house, prices, will, continue, to, fall*}), but there may be disagreement about which subset to use to represent the construction in the grammatical relations. Here, either *will* or *continue* could be used to represent the complement of *said*. This decision may also be theory-dependent to some degree, for example whether to use the determiner or the noun as the head of a noun phrase.

The final decision to make is the choice of relations and their arguments. This can also be thought of as the choice of how the set of representative words should be grouped into relations. For example, in a relative clause construction, the scheme designer must decide whether the relation between the relative pronoun and the head noun is important, or the relation between the relative pronoun and the verb, between the head noun and the verb, or some subset of these. The choice of label for each relation will be a natural part of this decision.

An important property of the representation, closely related to the choices made about representative words and how they are grouped into relations, is the number of relations used for a particular construction. We refer to this as the *compactness* property. Compactness essentially boils down to the valency of each relation and the information encoded in the label(s) used for the relation. We show that this property is closely related to the assigning of partial credit — awarding points even when a construction is not recovered completely correctly — and that it can have a significant effect on the resulting scoring mechanism.

The dividing lines between the various questions we have described are subtle, and in particular the last two questions (which words should represent the construction and which relations to use, and consequently how compactly the relations are represented) have significant overlap with one another. For example, if the auxiliary *are* in the passive construction *prices are affected* is chosen as one of the representative words, then a relation type which relates *are* to either *prices* or *affected* must also be chosen. For the relative clause construction *woman who likes apples and pears*, if the words and relations chosen include a representation along the lines of *relative-clause-subject(likes, woman)* and *subject(likes, who)*, then it is unlikely that the more compact relation *relative-clause(likes, woman, who)* would also be chosen. Despite the overlap, each question can provide a useful perspective for the designer of an evaluation scheme.

Decisions must be made not only about the representations of the individual constructions, but also about the interfaces between constructions. For example, in the sentence *Mary likes apples and pears*, the coordination structure *apples and pears* serves as direct object of *likes*, and it must be determined which word(s) are used to represent the coordination in the direct object relation.

We will illustrate some of the consequences of the decisions described here with detailed examples of three construction types. We focus on passive, coordination, and relative clause constructions, as analysed in the PARC (King et al., 2003), GR (Briscoe and Carroll, 2006), and Stanford (de Marneffe et al., 2006) evaluation schemes, using sentences from the shared task of the COLING 2008 parser evaluation workshop.[1] These three constructions were chosen because we believe they provide particularly good illustrations of the various decisions and their consequences for scoring. Furthermore, they are constructions whose representation differs across at least two of the three grammatical relation schemes under dicsussion, which makes them more interesting as examples. We believe that the principles involved, however,

[1] The shared task includes a number of additional formats besides the three grammatical relation schemes that we consider here, but the representations are sufficiently different that we don't consider a comparison fruitful for the present discussion.

apply to any linguistic construction.

We also wish to point out that at this stage we are not recommending any particular scheme or any answers to the questions we raise, but only suggesting ways to clarify the decision points. Nor do we intend to imply that the ideal representation of any linguistic construction, for any particular purpose, is one of the representations in an existing scheme; we merely use the existing schemes as concrete and familiar illustrations of the issues involved.

2 The Passive Construction

The following is an extract from Sentence 9 of the shared task:

> how many things are made out of eggs

We expect general agreement that this is a passive construction, and that it should be included in the evaluation scheme.[2] We also expect agreement that all the words in this extract are involved in the construction.

Potential disagreements arise when we consider which words should represent the construction. *Things*, as the head of the noun phrase which is the underlying object of the passive, and *made*, as the main verb, seem uncontroversial. We discard *how* and *many* as modifiers of *things*, and the prepositional phrase *out of eggs* as a modifier of *made*; again we consider these decisions to be straightforward. More controversial is whether to include the auxiliary verb *are*. PARC, for example, does not include it in the scheme at all, considering it an inherent part of the passive construction. Even if the auxiliary verb is included in the overall scheme, it is debatable whether this word should be considered part of the passive construction or part of a separate verb-auxiliary construction. Stanford, for example, uses the label auxpass for the relation between *made* and *are*, indicating that it is part of the passive construction.

The next decision to be made is what relations to use. We consider it uncontroversial to include a relation between *things* and *made*, which will be some kind of subject relation. We also want to represent the fact that *made* is in the passive voice, since this is an essential part of the construction and makes it possible to derive the underlying object position of *things*. If the auxiliary *are* is included, then there should be a verb-auxiliary relation between *made* and *are*, and perhaps a subject relation between *are* and *things* (although none of the schemes under consideration use the latter relation). PARC includes a variety of additional information about the selected words in the construction, including person and number information for the nouns, as well as tense and mood for the verbs. Since this is not included in the other two schemes, we ignore it here.

The relevant relations from the three schemes under consideration are shown below.[3]

PARC
passive(make, +)
subj(make, thing)

GR
(ncsubj made things obj)
(passive made)
(aux made are)

Stanford
nsubjpass(made, things)
auxpass(made, are)

PARC encodes the grammatical relations less compactly, with one subject relation joining *make* and *thing*, and a separate relation expressing the fact that *make* is in the passive voice. Stanford is more compact, with a single relation nsubjpass that expresses both verb-subject (via the arguments) and passive voice (via the label). GR has an equally compact relation since the obj marker signifies passive when found in the ncsubj relation. GR, however, also includes an additional feature passive, which redundantly encodes the fact that *made* is in passive voice.[4]

Table 1 shows how different kinds of parsing errors are scored in the three schemes. First note the differences in the "everything correct" row, which shows how many points are available for the construction. A parser that is good at identifying passives will earn more points in GR than in PARC and Stanford. Of course, it is always possible to look at accuracy figures by dependency type in order to understand what a parser is good at, as recommended by Briscoe and Carroll (2006), but it is

[2] PARC recognises it as an interrogative as well as a passive construction.

[3] Schemes typically include indices on the words for identification, but we omit these from the examples unless required for disambiguation. Note also that PARC uses the lemma rather than the inflected form for the head words.

[4] Although passive is technically a feature and not a relation, as long as it is included in the evaluation the effect will be of double scoring.

	PARC	GR	Stanf
Everything correct	2	3	2
Misidentify subject	1	2	1
Misidentify verb	0	0	0
Miss passive constr	1	1	0
Miss auxiliary	2	2	1

Table 1: Scores for passive construction.

also desirable to have a single score reflecting the overall accuracy of a parser, which means that the construction's overall contribution to the score is relevant.[5]

Observe also that partial credit is assigned differently in the three schemes. If the parser recognises the subject of *made* but misses the fact that the construction is a passive, for example, it will earn one out of two possible points in PARC, one out of three in GR (if it recognizes the auxiliary), but zero out of two in Stanford. This type of error may seem unlikely, yet examples are readily available. In related work we have evaluated the C&C parser of Clark and Curran (2007) on the BioInfer corpus of biomedical abstracts (Pyysalo et al., 2007), which includes the following sentence:

> Acanthamoeba profilin was cross-linked to actin via a zero-length isopeptide bond using carbodiimide.

The parser correctly identifies *profilin* as the subject of *cross-linked*, yet because it misidentifies *cross-linked* as an adjectival rather than verbal predicate, it misses the passive construction.

Finally, note an asymmetry in the partial credit scoring: a parser that misidentifies the subject (e.g. by selecting the wrong head), but basically gets the construction correct, will receive partial credit in all three schemes; misidentifying the verb, however (again, this would likely occur by selecting the wrong head within the verb phrase) will cause the parser to lose all points for the construction.

3 The Coordination Construction

The coordination construction is particularly interesting with regard to the questions at hand, both because there are many options for representing the construction itself and because the interface with other constructions is non-trivial. Here we consider an extract from Sentence 1 of the shared task:

> electronic, computer and building products

The coordination here is of nominal modifiers, which means that there is a decision to make about how the coordination interfaces with the modified noun. All the conjuncts could interact with the noun, or there could be a single relationship, usually represented as a relationship between the conjunction *and* and the noun.

Again we consider the decisions about whether to represent coordination constructions in an evaluation scheme, and about which words are involved in the construction, to be generally agreed upon. The choice of words to represent the construction in the grammatical relations is quite straightforward: we need all three conjuncts, *electronic*, *computer*, and *building*, and also the conjunction itself since this is contentful. It also seems reasonably uncontroversial to discard the comma (although we know of at least one parser that outputs relations involving the comma, the C&C parser).

The most difficult decision here is whether the conjuncts should be related to one another or to the conjunction (or both). Shown below is how the three schemes represent the coordination, considering also the interface of the coordination and the nominal modification construction.

PARC
adjunct(product, coord)
adjunct_type(coord, nominal)
conj(coord, building)
conj(coord, computer)
conj(coord, electronic)
coord_form(coord, and)
coord_level(coord, AP)

GR
(conj and electronic)
(conj and computer)
(conj and building)
(ncmod _ products and)

Stanford
conj_and(electronic, computer)
conj_and(electronic, building)
amod(products, electronic)
amod(products, computer)
amod(products, building)

[5] We assume that the overall score will be an F-score over all dependencies/features in the relevant test set.

Table 2 shows the range of scores assigned for correct and partially correct parses across the three schemes. A parser that analyses the entire construction correctly will earn anywhere from four points in GR, to seven points in PARC. Therefore, a parser that does very well (or poorly) at coordination will earn (or lose) points disproportionately in the different schemes.

	Parc	GR	Stanf
Everything correct	7	4	5
Misidentify conjunction	6	0	3
Misidentify one conjunct	6[a]	3	3[b]
Misidentify two conjuncts	5[a]	2	1

[a] The parser might also be incorrect about the coord_level relation if the conjuncts are misidentified.
[b] The score would be 2 if it is the first conjunct that is misidentified.

Table 2: Scores for coordination, including interface with nominal modification.

A parser that recognises the conjuncts correctly but misidentifies the conjunction would lose only one point in PARC, where the conjunction is separated out into a single coord_form relation, but would lose all four available points in GR, because the word *and* itself takes part in all four GR dependencies. Only two points are lost in Stanford (and it is worth noting that there is also an "uncollapsed" variant of the Stanford scheme in which the coordination type is not rolled into the dependency label, in which case only one point would be lost).

Note also an oddity in Stanford which means that if the first conjunct is missed, all the dependencies are compromised, because the first conjunct enters into relations with all the others. The more conjuncts there are in the construction, the more points are lost for a single parsing error, which can easily result from an error in head selection.

Another issue is how the conjuncts are represented relative to the nominal modifier construction. In PARC and GR, the conjunct *and* stands in for all the conjuncts in the modifier relation. This means that if a conjunct is missed, no extra points are lost on the modifier relation; whereas in Stanford, points are lost doubly – on the relations involving both conjunction and modification.

4 The Relative Clause Construction

For the relative clause construction, as for coordination, the choice of words used to represent the construction is straightforward, but the choice of relations is less so. Consider the following relative clause construction from Sentence 2 of the shared task:

not all those who wrote

All three schemes under consideration use the set {*those, who, wrote*} to describe this construction.[6]

PARC
pron_form(pro$_3$, those)
adjunct(pro$_3$, write)
adjunct_type(write, relative)
pron_form(pro$_4$, who)
pron_type(pro$_4$, relative)
pron_rel(write, pro$_4$)
topic_rel(write, pro$_4$)

GR
(cmod who those wrote)
(ncsubj wrote those _)

Stanford
nsubj(wrote, those)
rel(wrote, who)
rcmod(those, wrote)

Note that PARC represents the pronouns *who* and *those*, as it does all pronouns, at a more abstract level than GR or Stanford, creating a representation that is less compact than the others. GR and Stanford differ in terms of compactness as well: GR's cmod relation contains all three words; in fact, the ncsubj relationship might be considered redundant from the point of view of an evaluation scheme, since an error in ncsubj entails an error in cmod. Stanford's representation is less compact, containing only binary relations, although there is also a redundancy between nsubj and rcmod since the two relations are mirror images of each other.

For the sake of comparison, we include here two additional hypothetical schemes which have different characteristics from those of the three target schemes. In Hypothetical Scheme 1 (HS1), there are three relations: one between the head noun and the relative clause verb, one between the

[6] PARC also encodes the fact that pro$_3$ is a demonstrative pronoun, but we don't consider this part of the relative clause construction.

	PARC	GR	Stanf	HS1	HS2
Everything correct	7	2	3	3	1
Misidentify head noun	6	0	1	1	0
Misidentify verb	3	0	0	2	0
Miss relative clause construction	3	0	0	1	0

Table 3: Scores for relative clauses.

relative pronoun and the relative clause verb, and a third which relates the relative pronoun to the head noun. This third relation is not included in any of the other schemes. Hypothetical Scheme 2 (HS2) involves only one relation, which includes the same words as GR's cmod relation; the representation as a whole is quite compact since only one dependency is involved and it includes all three words.

Hypothetical Scheme 1
relative-subject(wrote, those)
subject(wrote, who)
relative-pronoun(those, who)

Hypothetical Scheme 2
relative-clause(wrote, those, who)

Table 3 shows the range of scores that can be attained in the different schemes. The total possible score varies from one for HS2, to three for Stanford and HS1, and up to seven for PARC.

Observe that any of the three types of error in Table 3 will immediately lose all points in both GR and HS2. Since all the schemes use the same set of words, this is due solely to the choice of relations and the compactness of the representations. Neither GR nor HS2 allow for partial credit, even when the parser assigns an essentially correct relative clause structure. This is a scenario which could easily occur due to a head selection error. For example, consider the following phrase from the shared task GENIA (Kim et al., 2003) data set , Sentence 8:

> . . . the RelA (p65) subunit of NF-kappa B , which activates transcription of the c-rel gene . . .

The C&C parser correctly identifies the relative clause structure, including the pronoun *which* and the verb *activates*, but incorrectly identifies the head noun as *B* instead of *subunit*.

Even between GR and HS2, which share the characteristic of not allowing for partial credit, there is a difference in scoring. Because GR starts with two dependencies, there is a loss of two points, rather than just one, for any error, which means errors in relative clauses are weighted more heavily in GR than in HS2.

Stanford also has a problematic redundancy, since the nsubj and rcmod relations are mirror images of each other. It therefore duplicates the GR characteristic of penalising the parser by at least two points if either the head noun or the relative clause verb is misidentified (in fact three points for the verb).

Observe also the asymmetry between misidentifying the head noun (one out of seven points lost in PARC, two out of three lost in Stanford and HS1) compared to misidentifying the verb (three points lost in PARC, all three lost in Stanford, but only one point lost in HS1). This reflects a difference between the schemes in whether the relative pronoun enters into a relation with the subject, the verb, or both.

5 Conclusion

In this paper we have shown how the design process for a relational parser evaluation scheme can be broken up into a number of decisions, and how these decisions can significantly affect the scoring mechanism for the scheme. Although we have focused in detail on three construction types, we believe the decisions involved are relevant to any linguistic construction, although some decisions will be more difficult than others for certain constructions. A direct object construction, for example, will normally be represented by a single relation between a verbal head and a nominal head, and indeed this is so in all three schemes considered here. This does not mean that the representation is trivial, however. The choice of which heads will represent the construction is important. In addition, Stanford distinguishes objects of prepositions from objects of verbs, while PARC and GR collapse the two into a single relation. Although part of speech information can be used to distinguish the two, a parser which produces PARC- or GR-style output in this regard will lose points in Stanford without some additional processing.

We have made no judgements about which decisions are best in the evaluation scheme design process. There are no easy answers to the questions raised here, and it may be that different solutions will suit different evaluation situations. We leave these questions for the parsing community to decide. This process may be aided by an empirical study of how the decisions affect the scores given to various parsers. For example, it might be useful to know whether one parser could be made to score significantly higher than another simply by changing the way coordination is represented. We leave this for future work.

References

Briscoe, Ted and John Carroll. 2006. Evaluating the accuracy of an unlexicalized statistical parser on the PARC DepBank. In *Proceedings of the ACL-Coling '06 Main Conf. Poster Session*, pages 41–48, Sydney, Austrailia.

Carroll, John, Ted Briscoe, and Antonio Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st LREC Conference*, pages 447–454, Granada, Spain.

Clark, Stephen and James R. Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552.

de Marneffe, Marie-Catherine, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th LREC Conference*, pages 449–454, Genoa, Italy.

Kim, Jin-Dong, Tomoko Ohta, Yuka Teteisi, and Jun'ichi Tsujii. 2003. GENIA corpus – a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19:i180–i182.

King, Tracy H., Richard Crouch, Stefan Riezler, Mary Dalrymple, and Ronald M. Kaplan. 2003. The PARC 700 Dependency Bank. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora*, Budapest, Hungary.

Lin, Dekang. 1995. A dependency-based method for evaluating broad-coverage parsers. In *Proceedings of IJCAI-95*, pages 1420–1425, Montreal, Canada.

Pyysalo, Sampo, Filip Ginter, Juho Heimonen, Jari Björne, Jorma Boberg, Jouni Järvinen, and Tapio Salakoski. 2007. BioInfer: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8:50.