

Computing Reliability for Co-Reference Annotation

Rebecca J. Passonneau

Columbia University
Computer Science Department
New York, NY 10027
becky@cs.columbia.edu

Abstract

Co-reference annotation is annotation of language corpora to indicate which expressions have been used to co-specify the same discourse entity. When annotations of the same data are collected from two or more coders, the reliability of the data may need to be quantified. Two obstacles have stood in the way of applying reliability metrics: incommensurate units across annotations, and lack of a convenient representation of the coding values. Given N coders and M coding units, reliability is computed from an N -by- M matrix that records the value assigned to unit M_j by coder N_k . The solution I present accommodates a wide range of coding choices for the annotator, while preserving the same units across codings. As a consequence, it permits a straightforward application of reliability measurement. In addition, in coreference annotation, disagreements can be complete or partial. The representation I propose has the advantage of incorporating a distance metric that can scale disagreements accordingly. It also allows the investigator to experiment with alternative distance metrics. Finally, the coreference representation proposed here can be useful for other tasks, such as multivariate distributional analysis. The same reliability methodology has already been applied to another coding task, namely semantic annotation of summaries.

1. Introduction

Co-reference annotation is annotation of language corpora to indicate which expressions have been used to co-specify the same discourse entity. No matter how precise a language user might be, language interpretation is subjective. A given expression can be referentially ambiguous or vague. When annotations of the same data are collected from two or more coders, the reliability of the data should be quantified. Two obstacles have stood in the way of applying a reliability metric to coreference annotation.

1. Annotators might disagree about which expressions are referential
2. Different coders might assign a different number of distinct referential indices

Given N coders and M coding units, reliability is computed from an N -by- M matrix that records the value assigned to unit M_j by coder N_k . Item 1 can be an obstacle to comparing units across annotations; in addition, sometimes annotation projects allow for the annotators to choose which units to code. The solution presented here specifies how to accommodate distinct coding choices while preserving the same units across codings. Item 1 can be an obstacle to comparing coding values across annotations; in fact, determining how to represent the *value* assigned to each unit in coreference annotation is a problem that has not been addressed directly. The method I present for representing coreference annotation permits a straightforward application of a reliability metric. In addition, it allows the investigator to assign a distance metric to values of annotation or coding variables that corresponds to type of scale the values fall within (e.g., nominal vs. ordinal, discrete vs. continuous), and is thus compatible with a reliability metric like Krippendorff's α , which accommodates a range of distance metrics supplied by the investigator. The representation can also be useful for other analytic tasks, such as exploratory data analyses, or multivariate distributional analyses.

2. Problem

Coreference annotation plays a role in the preparation of training corpora for many kinds of applications, from information extraction systems (Kibble and van Deemter, 2000) to dialogue systems (Poesio, 1993). Regardless of the language (e.g., German (Kunz and Hansen-Schirra, 2003)), the modality (e.g., spoken versus written), or the genre (e.g., newswire text versus interview dialogue), different forms can refer to the same entity. While languages like Japanese and English contrast greatly in the degree to which elliptical references can occur, the problem of whether to annotate zero references is a distinct question that should be addressed by the coreference annotation guidelines or tools. Here I assume that such guidelines specify what surface expressions, or elided expressions, will be annotated for coreference. For the sake of simplicity, all the examples in this paper will involve explicit noun phrases, whether proper names, definite or indefinite descriptions, or pronouns.

Figure 1 illustrates a newswire text annotated by three coders. Following the principles outlined in (Passonneau, 1994), I assume that all tokens to be annotated have been identified in advance by the investigator, either through a separate manual annotation process or automatically. I also assume the task of each annotator is to indicate for each token whether it refers to an existing discourse entity, in which case it should be coindexed with all the expressions that have already been indexed for this entity; or introduces a new discourse entity, in which case it receives a new index; or does not refer, in which case it receives no index. token in the text.

All three cases are illustrated in Table 1, which shows how three annotators—RA.1, RA.2, RA.3—indexed the selected NPs for coreference. Two NPs receive the same index if they corefer, thus all three coders assign the same index to token A (*Gov. Price Daniel*) and token D (*Daniel*). Note that RA.1 and RA.3 assign 4 indices whereas RA.2 assigns 5, and in addition assigns the NIL value to token J.

Committee approval of [Gov. Price Daniel](A)’s [abandoned property act](B) seemed certain Thursday despite the protests of [Texas bankers](C). [Daniel](D) personally led the fight for [the measure](E). Under committee rules, [it](F) went automatically to a subcommittee for one week. But questions with which [committee members](G) taunted [bankers](H) appearing as [witnesses](I) left little doubt that [they](J) will recommend passage of [it](K).

- The NPs of interest have been bracketed; i.e., not all NPs will be coded.
- Each bracketed NP token receives a unique index.

Figure 1: A Newsire text prepared for coreference annotation

ID	TOKEN	RA.1	RA.2	RA.3
A	Gov. Price Daniel	1	1	1
B	abandoned property act	2	2	2
C	Texas bankers	3	3	3
D	Daniel	1	1	1
E	the measure	2	2	2
F	it	2	2	2
G	committee members	4	4	4
H	bankers	3	5	3
J	witnesses	4	NIL	3
K	they	4	4	3
L	it	2	2	2

Table 1: Co-reference annotation for a newswire text

Figure 2 represents the equivalence classes constituted by the tokens that corefer. Each class represents a single *entity* or index, and links the expressions that refer to that entity, as well as the predications the expressions occur in. Thus each class represents what the annotator takes to have been asserted about the presumed discourse entity. If annotators disagree on the member of an equivalence class, this may reflect a different interpretation of the discourse.

From Table 1, it might seem that the referential indices could be used as variable values, assuming they can be *normalized* to a single set of symbols. For example, RA.1 and RA.3 both use 4 distinct indices, making it possible to use the same 4 coding values for each coding. However, a closer look suggests that using referential indices as coding values is a mistake.

A clear cut case of identical codings can be seen for tokens A and D: all three coders assign tokens A and D to the same equivalence class. By attending to the equivalence class representation instead of the one in Table 1, we can see more clearly how RA.1 and RA.3 disagree, despite their use of the same number of referential indices. Only two of their equivalence classes are identical. In RA.1’s coding, tokens C and H corefer, and do not corefer with any other tokens. In RA.3’s coding, tokens C, H, J and K are assigned to the same equivalence class. If we assume the number “4” is chosen to be the index for C in RA.1, what would it mean

RA.1 (N=4) {A, D} {B, E, F, L} {C, H} {G, J, K}
 RA.2 (N=6) {A, D} {B, E, F, L} {C} {G, K} {} {H} {J}
 RA.3 (N=4) {A, D} {B, E, F, L} {C, H, J, K} {G}

11 Coding Values: {A, D} {B, E, F, L} {C, H, J, K} {C, H, J} {C, H} {C} {G, J, K} {G, K} {G} {H} {}

Figure 2: Equivalence Classes from Three Annotations Yielding 9 Coding Values

	A	B	C	D	E	F	G	H	J	K	L
RA.1	M1	N2	Y2	M1	N2	N2	X1	Y2	X1	X1	N2
RA.2	M1	N2	S6	M1	N2	N2	P3	T7	U8	P3	N2
RA.3	M1	N2	Q4	M1	N2	N2	W9	Q4	Q4	Q4	N2

Table 2: Canonical form for coreference annotation reliability

for “4” to be chosen as the index for C in RA.3, given that coder RA.3 thinks there are four references to this entity compared with two for RA.1? The two coders may have very different conceptions of what has been said about this entity.

At the bottom of Figure 2 is the set of equivalence classes created by the union of all the codings. I propose to use these classes as the *values* assigned to each token in order to directly represent when two annotators have assigned the same referential value to a linguistic expression. This proposal thus results in eleven *values*, instead of the four or five in the individual codings.

The proposed representation also makes it easier to compare disagreements. Where two codings disagree, the penalty assigned to the disagreement will depend on how different the equivalence classes are. Here, RA.1 and RA.2 assign different values to tokens C and H: {C, H} versus {C, H, J, K}. Note that the former is a subset of the latter. Intuitively, this difference in values should be penalized less than if the two sets were related by intersection rather than a subset relation, which in turn should be penalized less than if they were disjoint.

3. Proposed Solution

3.1. Representation of the Coded Data

Although I propose to use the equivalence classes that tokens are assigned to as the coding values, this representation can become unwieldy if the number of tokens and number of coders grows large. In this section, I introduce a level of indirection that makes the representation more compact.

I assign a unique index to each equivalence class, analogous to the primary key in a relational database. For clarity of presentation, I use a combination of letters of the alphabet and integers to differentiate the variable values from the tokens IDs (letters) and from the original referential indices (integers). Table 2 shows the same data in this new representation that uses indices to point to the equivalence classes.

Table 2 is in a form that Krippendorff (Krippendorff, 1980) refers to as the canonical form of a reliability table:

the columns labels are the units being coded, which in this case are NP tokens; the row labels are the annotators; the cell contents indicate the value that a specific annotator assigned to a specific unit. This representation, in contrast to the one in Table 1, shows much very clearly the distribution of agreement and disagreement across annotators. The columns where each cell has the same value correspond to the tokens where all coders assigned the same values: the columns for tokens A, B, D, E, F, and K. Similarly, patterns of disagreement are directly discernible. Because no column that does not show perfect agreement has less than 3 symbols, we see easily that there are no cases where two coders agree and the third disagrees. Further, a symbol that occurs in only one row, e.g., *Q4*, indicates an equivalence class assigned by only one of the annotators (e.g., RA.3: {C, H, J, K}). What is missing from this representation is how to quantify the difference in values in a way that accords with the informal observation made above about the case where one coding subsumed the other.

If we treat the cell values in Table 2 as "nominal" variables, meaning the difference between *N2* and *P3* is the same as the difference between *P3* and *R5*, then the inter-annotator reliability for the data in Table 2, using Krippendorff's α , is .45.¹ The distance metric for nominal data is binary: all values are either alike (delta=1) or not (delta=0). This is not the best way to compute reliability for coreference annotation because it fails to capture the intuition that some equivalence classes are more alike than others.

3.2. Krippendorff's α

I briefly illustrate here the formula for Krippendorff's α , mainly to illustrate where the distance metric figures in. Given a table of the form in Table 2 with m coders and r coding units, the agreement coefficient is given by summing all the disagreements within and across columns, as computed with the following α formula:

$$\alpha = 1 - \frac{rm - 1}{m - 1} \frac{\sum_i \sum_b \sum_{c>b} n_{b_i} n_{c_i} \delta_{bc}}{\sum_b \sum_c n_b n_c \delta_{bc}}$$

The numerator is a summation over the product of counts of values b and values c , for all pairs of values, times the δ , thus in nominal scales the product will be zero when $b = c$. The denominator is a summation over the distribution of agreements and disagreements within columns (for j to m columns). A full discussion is in Krippendorff (Krippendorff, 1980).

3.3. Distance Metrics for Sets

Now I will illustrate three cases of disagreement using examples where a pair of coders assigned distinct *referential values* to the same NP tokens. In the first example, the values are fairly similar: one is a subset of the other. Coder RA.1 assigned to unit C a value represented in Table 2 as Y2 whereas RA.3's coding is represented as Q4. These correspond respectively to the sets {C, H} and {C, H, J, K}. In

¹Conceptually, α resembles Cohen's Kappa (Cohen, 1960): it is 1 less the ratio of observed disagreements to expected disagreements, but is more general to handle multiple coders and multiple coding scales.

my coding scheme, every token is necessarily a member of the set that is its *referential value*, so to pose the question of how different two values are, I first remove the current token from the values assigned by the annotators; note that if this step is not taken, all values across annotators for the same unit will necessarily overlap. Here, in the case of the RA.1 and RA.3 codings of C, the set differences yield {H, J, K} and {H}. These two sets thus represent each annotator's decision about the co-specifying expressions for C: RA.3's coding of C subsumes RA.1's.

In the second example, we find a different set relation. Coder RA.1 assigned a value to J encoded as X1, whereas RA.4's coding is represented as Q4. Removing J itself from the equivalence classes that are the *actual* values yields {G, K} and {C, H, K}. Neither set subsumes the other, but the set intersection is non-null: {K}. In this case, the referential values of the two annotations overlap, so they are not in as much disagreement the third case, where there are disjoint difference sets. The example that illustrates the third case involves token K: RA.2 and RA.3 assigned the values P3 and Q4, whose difference sets are {G} and {C, H, J}, which are disjoint.

Intuitively, identity, subsumption, intersection and disjunction are ordered from most agreement to least agreement. To capture this intuition, I assign the δ values 0 for identity, .33 for subsumption, .66 for intersection, and 1 for disjunction. Applying this distance metric to the data in Table 2 yields a much higher α of 74.

Let us reconsider the distribution we see in Table 2 as we attempt to understand the two different values for α we get, depending on the distance metric. If we look only at the columns with perfect agreement, we see quickly that 6 out of 11 columns exhibit this pattern, or roughly half the table. On a metric that treats all disagreements equally, than about half the data involves disagreement, which corresponds to the case where we treat the values as nominal data, and we get $\alpha = .45$, or close to half. However, if we treat the non-identical values within a column for a token as more or less different, depending on whether we find subsumption, intersection or disjunction relations, then we can capture the intuition that the disagreements are not all alike: some should be weighted more heavily than others.

4. Comparison with Other Coreference Scoring Schemes

Because of the obstacles to applying a reliability metric to coreference data, other investigators who have looked at how to compare coreference annotations have used recall and precision, notably (Vilain et al., 1995). As proposed here, they use equivalence classes to represent a coreference encoding, but their approach is otherwise entirely different. In (Passonneau, 1997), I reported on a comparison of their approach with an earlier version of the method proposed here. The main difference between (Vilain et al., 1995) and (Passonneau, 1997) was that I addressed the problem of incommensurate units in order to apply Cohen's κ .

To compare the two metrics, I used coreference data I had already collected. As described in work (Passonneau

and Litman, 1997), we had created a gold-standard coreference coding of a set of transcribed spoken monologues known as the Pear stories (Chafe, 1980). To evaluate the difficulty of coreference annotation, I computed interannotator reliability for a new annotator encoding the entire set of twenty narrative monologues against our gold standard.² Before coding the monologues, the annotator had had multiple training sessions.

To be able to apply κ , I first insured that the gold standard and the new annotation would have the same coding units, i.e., the same NPs to encode. The annotator was free to put an NP into a singleton set (no co-specifying phrases), or to assign the empty set (non-referential), both possibilities being inadmissible in (Vilain et al., 1995). Then I showed that given commensurate units, the precision and recall method proposed in (Vilain et al., 1995) for handling the equivalence classes resulting from a coreference annotation could yield a contingency table for computing Cohen's Kappa. A comparison of the two metrics on the same data showed clear differences, which suggests that precision and recall should not be relied on as a substitute for a reliability metric. A reliability metric takes into account the frequency with which annotators apply coding variables, thus builds in probability. κ ranged from .65 to .93, depending on the narrative. Impressionistically, this correlated with differences in the speaker's ability to tell the story clearly. In contrast, precision ranged from .92 to .99, and recall from .88 to .97. The narrative that had a κ of .65 had recall and precision of .90 and .93. Thus their metric does not discriminate between narratives that were easier to code and narratives that were more difficult.

Another treatment that builds on (Vilain et al., 1995) appears in (Baldwin et al., 1998).

5. Conclusion

Previous work on annotating coreference has not computed inter-annotator reliability using an agreement metric. The obstacles to doing so included:

1. the lack of a method for comparing the *referential value* assigned to a unit by different coders;
2. the potential that the units that were coded would not be the same units.

I have presented an annotation method that solves both problems, and that allows for direct application of an agreement metric such as Krippendorff's α . By proposing a method for using equivalence classes as the coding values for each referential token, or coding unit, I have made it possible also to propose a meaningful distance metric that assigns numeric values to distinct relations among sets, in particular, set identity, subsumption, intersection and disjunction. Where a gold standard for coreference annotation in a corpus is lacking, this annotation method in combination with evaluation of interannotator reliability can facilitate the creation of a gold standard. Where a gold standard

exists, this method can be used to quantify agreement of human or automated annotations to the gold standard.

The results presented here are not restricted to annotations of coreference relations. This is a method for comparing annotations where the annotators create sets from the units being coded, and are free to create any number of sets. In (Passonneau and Nenkova, 2003), we faced an identical problem in a semantic annotation method we designed for annotating content units in summaries. To evaluate the interannotator of our Summary Content Units (SCUs), I applied the identical approach presented here, as noted in (Nenkova and Passonneau, (To appear) 2004).

6. References

- Baldwin, Breck, Tom Morton, Amit Bagga, Jason Baldridge, Raman Chandraseker, Alexis Dimitriadis, Kieran Snyder, and Magdalena Wolska, 1998. Description of the upenn camp system as used for coreference. In *Proceedings of the 7th Message Understanding Conference*, volume MUC-7.
- Chafe, Wallace L., 1980. *The Pear Stories: Cognitive, Cultural and Linguistic Aspects of Narrative Production*. Norwood, NJ: Ablex Publishing Corporation.
- Cohen, Jacob, 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.
- Kibble, Rodger and Kees van Deemter, 2000. Coreference annotation: Whither? In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*.
- Krippendorff, Klaus, 1980. *Content Analysis: An Introduction to Its Methodology*. Beverly Hills, CA: Sage Publications.
- Kunz, Kerstin and Silva Hansen-Schirra, 2003. Coreference annotation of the tiger treebank. In *Poster Session Presentation at Second Workshop on Treebanks and Linguistic Theories (TLT 2003)*.
- Nenkova, Ani and Rebecca Passonneau, (To appear) 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology conference and North American chapter of the Association for Computational Linguistics annual meeting*.
- Passonneau, Rebecca and Ani Nenkova, 2003. Evaluating content selection in human- or machine-generated summaries: The pyramid method. Technical report, Columbia University.
- Passonneau, Rebecca J., 1994. Protocol for coding discourse referential noun phrases and their antecedents. Technical report, Columbia University.
- Passonneau, Rebecca J., 1997. Applying reliability metrics to co-reference annotation. Technical Report CUCS-017-97, Columbia University.
- Passonneau, Rebecca J. and Diane Litman, 1997. Discourse segmentation by human and automated means. *Computational Linguistics*, 23.1:103–139. Special Issue on Empirical Studies in Discourse Interpretation and Generation.
- Poesio, M., 1993. A situation-theoretic formalization of definite description interpretation in plan elaboration di-

²One use of a reliability metric is to determine whether an annotator can match the performance of a gold standard, and thus be enlisted to annotate new gold standard data.

alogues. In P. Aczel, D. Israel, Y. Katagiri, and S. Peters (eds.), *Situations Theory and its Applications, vol.3*, chapter 12. Stanford: CSLI, pages 339–374.

Vilain, Marc, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman, 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th Message Understanding Conference*. San Francisco: Morgan Kaufmann.