A Model-Theoretic Coreference Scoring Scheme

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, Lynette Hirschman
The MITRE Corporation

This note describes a scoring scheme for the coreference task in MUC6. It improves on the original approach in (Sundheim *et al.* 1994)[1] by: (1) grounding the scoring scheme in terms of a model; (2) producing more intuitive recall and precision scores; and (3) not requiring explicit computation of the transitive closure of coreference. The principal conceptual difference is that we have moved from a syntactic scoring model based on following coreference links to an approach defined by the model theory of those links.

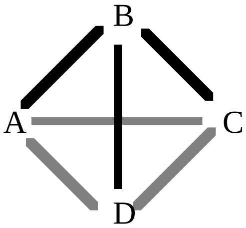
In brief, the scheme operates by comparing the equivalence classes defined by the links in the key and the response, rather than the links themselves (thus, this is only well defined for identity links, at the moment). These classes are of course the models of the IDENT equivalence relation, and this strategy is preferable for a number of reasons, one being that the scores are independent of the particular links used to encode the equivalence relation. The scores themselves are obtained by determining the minimal perturbations to the response that are required to transform its corresponding equivalence classes into those of the key. Specifically, the recall (respectively precision) error terms are found by calculating the least number of links that need to be added to the response (respectively the key) in order to have the classes align. Although at first blush this seems combinatorially explosive, due to references to minimal spanning subsets of the equivalence relation, it turns out it can be accomplished with a very simple counting scheme.

A problematic case.

Consider the first problematic example in (Sundheim *et al.* 1994):

- Key Links: <A-B B-C B-D>
- Response Links: <A-B C-D>

Note that the key links generate an equivalence class, the set {A B C D}. Technically, the links are a spanning tree of the set's implicit equivalence graph, i.e., the *fully-connected* graph whose nodes are the entities A, B, C, and D. The following figure shows the spanning tree in dark lines, and the rest of the graph in gray lines.

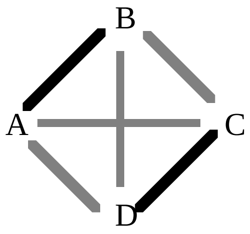


This is just one such spanning tree for the overall equivalence class; there are of course others, including the "non-problematic" case of <A-B B-C C-D>. Either way, a minimal spanning tree of the equivalence relation will always be of size 3, which aligns with the intuitive notion that three links will always be necessary to make four entities coreferential under the criterion of strict identity.

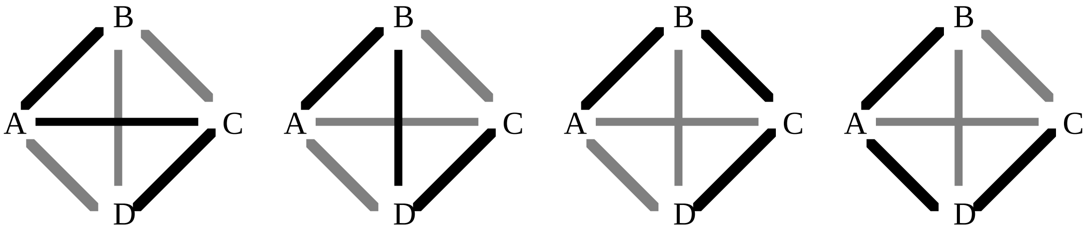
[1] Sundheim, B., Chinchor, N., and Grishman, R. (1994) Coreference task definition., Vers. 2.0 and earlier.

Returning to the task of scoring coreference for this problematic case, we note that a response of <A-B C-D> induces two equivalence classes, thus partitioning the set of key entities into subsets {A B} and {C D}. It is intuitive that the precision score for this response should be 2/2 = 1, since 2 out of 2 of the response links are "correct". That is, both response links are arcs in the equivalence graph generated by the key. For recall, Sundheim *et al.* advance the desirable score of 2/3, which is not obtained by the syntactic scoring measure. This score appeals to the intuitive notion that of the three links necessary to make the key entities fully coreferential, the response only provides two.

Thinking model-theoretically, we note that the response corresponds to a subgraph of the fully-connected equivalence graph.



The recall score of 2/3 aligns with the fact that one equivalence arc is required to "complete" the response graph, yielding one of the following four spanning trees.



Note that the problem with the syntactic (link-wise) scorer is that there are combinatorially many such spanning trees for a given equivalence class, while keys only list one.

Computing model-theoretic recall

How then can we turn this notion of minimal missing links into a computationally effective scoring procedure that works in the general case? Roughly stated, the scoring mechanism for recall must form the equivalence sets generated by the key, and then determine for each such key set how many subsets the response partitions the key set into. The score then follows by simple arithmetic.

Getting a bit more formal (but not much), let us define recall using these notions. First, let S be an equivalence set generated by the key, and let $R_1 \ldots R_m$ be equivalent classes generated by the response. Then we define the following functions over S:

- p(S) is a partition of S relative to the response. Each subset of S in the partition is formed by intersecting S and those response sets R_i that overlap S. Note that the equivalence classes defined by the response may include implicit singleton sets—these correspond to elements that are mentioned in the key but not in the response. For example, say the key generates the equivalence class S = {A B C D}, and the response is simply <A-B>. The relative partition p(S) is then {A B} {C} and {D}.

- c(S) is the minimal number of "correct" links necessary to generate the equivalence class S. It is clear that c(S) is one less than the cardinality of S, *i.e.,*

  $$c(S) = (|S| - 1)$$

- m(S) is the number of “missing” links in the response relative to the key set S. As noted above, this is the number of links necessary to fully reunite any components of the p(S) partition. We note that this is simply one fewer than the number of elements in the partition, that is,

$$m(S) = (|p(S)| – 1)$$

Looking in isolation at a single equivalence class in the key, the recall *error* for that class is just the number of missing links divided by the minimal number of correct links, i.e.,

$$\frac{m(S)}{c(S)}$$

Recall in turn is

$$\frac{c(S) – m(S)}{c(S)}$$

$$= \frac{(|S| – 1) – (|p(S)| -1)}{|S| – 1}$$

$$= \frac{|S| – |p(S)|}{|S| – 1}$$

To see how this works in practice, consider the second problematic example noted by Sundheim *et al.*

- Key Links: <A-B B-C>
- Response Links: <A-C>

The key generates a single equivalence class S: {A B C}. The size of the class is

$$|S| = 3,$$

and the minimum number of links necessary to establish the class is

$$c(S) = (|S| – 1) =2.$$

The response partitions this class into a partition p(S) of size 2, containing {A C} and {B}, where the latter element is implicitly defined. Working through the arithmetic, we have:

$$R = \frac{|S| – |p(S)|}{|S| – 1}$$

$$= \frac{3 – 2}{3 – 1}$$

$$= 1/2$$

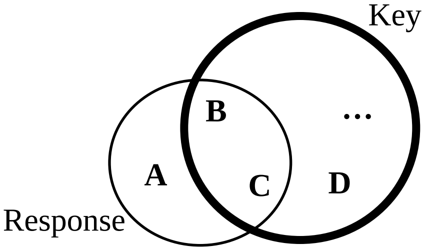
This score of 1/2 is the intuitively “correct” one that the syntactic measure fails to calculate.

Finally, we note that extending this measure from a single key equivalence class to an entire test set T simply requires summing over the key equivalence classes. That is,

$$R_T = \frac{\sum(|S_i| - |p(S_i)|)}{\sum(|S_i| - 1)}$$

Scoring precision

The recall scoring procedure operates by merging the subsets of a key equivalence class that are defined by equivalence classes in the response. It is of course the case that the response classes may not be proper subsets of the key. When the response overlaps the key in such a way as to produce a non-trivial set difference, as in the following figure, the response contains precision errors.



How may we use our model-theoretic notions to provide a scoring mechanism for precision? In the case of recall, we conceptually needed to add links to the response, building up the response's equivalence classes so as to end up with the key. In the case of precision, we need to do the converse: add links to equivalence classes in the key so as to yield equivalence classes in the response. We are switching the "figure" and the "ground"; that is we are switching our notion of where the base sets come from (the response rather than the key), and of what defines the partitions on those base sets (the key rather than the response).

More precisely, given an equivalence class S' defined by the response, we must determine the minimal number of links to be added to the key, so as to ensure that each of the members of the response set is in the same key set. Once again, we proceed by generating a relative partition, in this case the partition of the response equivalence class S' relative to key equivalence classes $K_1 \ldots K_n$. Elements of the response that are not found in the key once again generate implicit subsets, one per element. The number of missing elements is once again 1 less than the size of the partition.

For the example above, we see that the response generates an equivalence class of size 3, namely the set S' = {A B C}. The key partitions this class into subsets {B C} and {A}, where the latter is implicit. The partition is of size 2, and so the minimal number of links that need to be added to reunite the partition is just 1.

Switching the figure and ground in the recall formula, the scoring arithmetic for precision works itself out as follows, where S' is now an equivalence class from the response, and p'(S') is the partition of S' vis-à-vis the key(s). We then have:

$$c'(S') = (|S'| - 1) \qquad \textit{size of spanning tree for S'}$$

$$m'(S') = (|p'(S')| - 1) \qquad \textit{number of missing links}$$

$$P = \frac{c'(S') - m'(S')}{c'(S')} \qquad \textit{Precision}$$

$$= \frac{(|S'| - 1) - (|p'(S')| -1)}{|S'| - 1}$$

Key	Response	R	P
A-B B-C C-D	A-B C-D	2/3	2/2
A-B C-D	A-B B-C C-D	2/2	2/3
A-B B-C B-D	A-B B-C C-D	3/3	3/3
A-B B-C B-D	A-B C-D	2/3	2/2
A-B B-C	A-C	1/2	1/1

Table 1: Recall and precision scores for coreference examples

$$= \frac{|S'| - |p'(S')|}{|S'| - 1}$$

Thus, like the scheme proposed in Sundheim *et al.*, we have an aesthetically pleasing inverse relationship between precision and recall. For the example above, these formulæ yield a precision of 1/2, which is intuitively appropriate, since of the two minimum links needed to generate the response class {A B C}, the key only provides one, B-C. To extend from a single response to a complete test set T, we once again sum over the test set, this time iterating over response equivalence classes.

$$P_T = \frac{\sum \left(|S'_i| - |p'(S'_i)|\right)}{\sum \left(|S'_i| - 1\right)}$$

Table 1 shows the precision and recall scores, using the model-theoretic measures , for all of the examples given in Sundheim *et al.* Note that these results agree with the original scoring proposal for the first three cases, but agree with intuition for the last two.

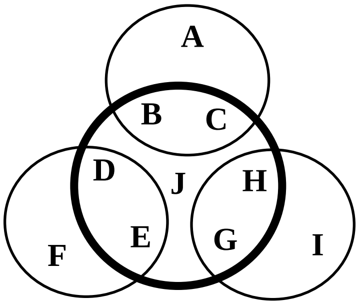
Examples with more complexity

The examples so far have been purposefully simplified in that we have only considered cases that defined one key class and one response class. Let us now consider some more-complex examples where keys and responses don’t so neatly overlap.

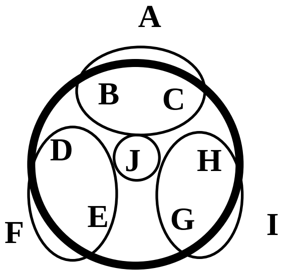
To begin with, imagine that the key and response are as follows.

- Key: <B-C C-D D-E E-G G-H H-J>
- Response: <A-B B-C D-E E-F G-H-H-I>

The key establishes a single equivalence class, while the response defines three: {A B C}, {D E F}, {G H I}. In addition, the key contains the element J, which is missing entirely from the response. These are shown in the following figure, where the thick lines denote the key class, and thin ones denote response classes.



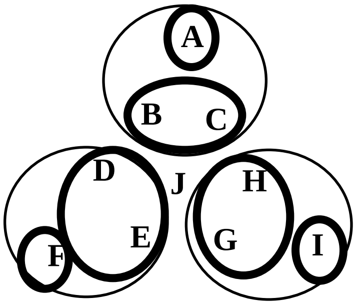
Eyeballing the problem, we note that as there are seven elements in the key, six links must minimally be provided to achieve 100% recall. The response only provides three correct ones, so we would expect recall to come out at 50%. Precision should be 50% as well, as half of the links indicated in the response are not in the key. Working through the math, we note that p(S), the partition of the key with respect to the response yields four subsets, shown in the following figure with thin lines.



Evaluating for recall:

$$
\begin{aligned}
R &= \frac{|S| - |p(S)|}{|S| - 1} \\
&= \frac{7-4}{7-1} \\
&= 3/6\text{, or }50\%
\end{aligned}
$$

For precision, we must consider the three equivalence classes defined in the response. Partitioning these three classes with respect to the key yields two subsets in each of the classes, for a total of six subsets. These are shown in thick lines in the following figure.



To evaluate for precision, we must use the corpus-wide formula:

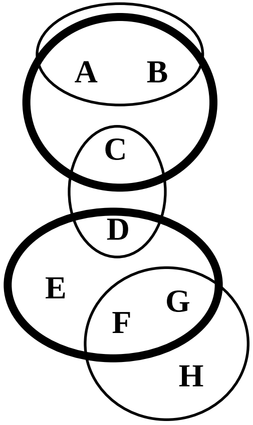
$$P_T = \frac{\sum\left(|S'_i| - |p'(S'_i)|\right)}{\sum\left(|S'_i| - 1\right)}$$

$$= \frac{(3-2)+(3-2)+(3-2)}{(3-1)+(3-1)+(3-1)}$$

$$= 3/6, \text{ or } 50\%$$

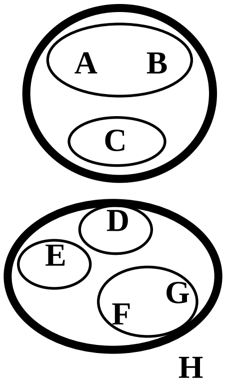
Before closing, let us consider an even more complex example with multiple sets in both the key and the response.

- Key: <A-B B-C D-E E-F F-G>
- Response: <A-B C-D F-G G-H>



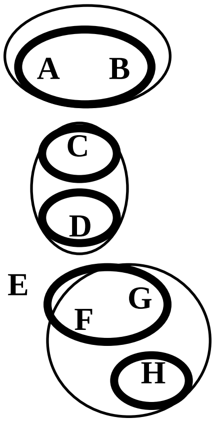
Thick = key
Thin = response

Figure 1: Key and response



Thick = key
Thin = partition wrt/response

Figure 2: Recall



Thick = partition wrt/key
Thin = Response

Figure 3: Precision

Figure 1 shows the equivalence classes corresponding to the key and response. Figure 2 shows how the response partitions the two key sets. Based on this, we can compute recall using the corpus-wide formula for recall. Since there are two key sets, there will be two terms each in the numerator and the denominator.

$$R_T = \frac{\sum\left(|S_i| - |p(S_i)|\right)}{\sum\left(|S_i| - 1\right)}$$

$$= \frac{(3-2)+(4-3)}{(3-1)+(4-1)}$$

$$= 2/5 \text{ or } 40\%$$

This is consistent with the observation that the response only provides two out of the five links that are minimally required to fully designate all the coreference relations. Turning to precision, Figure 3 shows the partitions induced on the three response sets by the key.

$$P_T = \frac{\sum\left(|S'_i| - |p'(S'_i)|\right)}{\sum\left(|S'_i| - 1\right)}$$

$$= \frac{(2-1)+(2-2)+(3-2)}{(2-1)+(2-1)+(3-1)}$$

$$= 2/4, \text{ or } 50\%$$

It is delightful that the formula yields exactly what intuition would dictate in this case.

Computational considerations, briefly explored

Our scoring expressions are easy to compute. The key step is the formation of equivalence classes, which can be accomplished by many algorithms. Tarjan's classic UNION-FIND, to cite the obvious example, operates in time effectively linear with the number of entities under consideration. This is substantially less expensive than enumerating the transitive closure of keys and answers, as is required by the original syntactic scoring procedure. Not often does attention to model theory yield efficient algorithms, but in this particular case the effort was well worth the while.