RC23543 (W0502-146) February 28, 2005
Computer Science

IBM Research Report

A Performance Metric for Coreference Resolution: Constrained Entity-Alignment F-Measure (CEAF)

Xiaoqiang Luo
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

A Performance Metric for Coreference Resolution: Constrained Entity-Alignment F-Measure (CEAF)

Xiaoqiang Luo
1101 Kitchawan Road, Room 23-121
Yorktown Heights, NY 10598, U.S.A.
xiaoluo@us.ibm.com

Abstract

The paper proposes a Constrained Entity-Alignment F-Measure (CEAF) for coreference resolution based on the best one-one map between reference and system entities. An efficient algorithm is presented to compute the proposed metric. Problems associated with the MUC link-based F-measure (and its variation B-cube F-measure) are fixed in the proposed metrics. Compared with ACE-value, the official metric in the ACE task, the proposed metrics are easier to interpret.

1 Introduction

A working definition of coreference resolution is partitioning noun phrases into equivalence classes, each of which refers to a physical object. We adopt the terminologies used in the Automatic Content Extraction (ACE) task (NIST, 2003) and call each individual noun phrase a *mention* and equivalence class an *entity*. In the following example,

> (1): "The American Medical Association voted yesterday to install the heir apparent as its president-elect, rejecting a strong, upstart challenge by a district doctor who argued that the nation's largest physicians' group needs stronger ethics and new leadership."

mentions are underlined (and the set of mentions in the same entity are marked with the same color). That is, "American Medical Association", "heir apparent," and "its" are examples of mentions. Mentions referring to the same object form an entity. For example, "American Medical Association", "its" and "group" refer to the same organization (object) and they belong to the same entity. Similarly, "heir apparent" and "president-elect" refer to the same person and they form another entity. It is worth pointing out that the entity definition here is different from what used in the Message Understanding Conference (MUC) task (MUC, 1995; MUC, 1998) – ACE entity is called coreference chain or equivalence class in MUC, and ACE mention is called entity in MUC.

An important problem in coreference resolution is how to evaluate a system's performance. A good performance metric should has the following two properties:

- Discriminativity: This refers to the ability to differentiate a good system from a bad one. While this criterion sounds trivial, not all performance metrics possess this property.
- Interpretability: A good metric should be easy to interpret. In other words, a good metric should make it easy to make a statement such as "a system is about 80% correct" and that is, there should be an intuitive "feeling" of how good a system is, and when the metric suggests that a certain percentage coreference results are correct.

A widely-used metric is the link-based F-measure (Vilain et al., 1995) adopted in the MUC tasks. It is computed by first counting the number of common links between the reference (or "truth") and the system output (or "response"); the link precision is the number of common links divided by the number of links in the system output, and the link recall is the number of common links divided by the number of links in the reference. There are known problems associated with the link-based F-measure. First, it ignores single-mention entities since no link can be found in these entities; Second, and more importantly, it lacks of power of differentiating system outputs with different qualities. Third, the link-based F-measure

intrinsically favors systems outputting less number of entities, and may result in higher F-measures for worse systems (c.f. Example in Table 3). We will revisit these issues and give examples in Section 3. In other words, it lacks of discriminativity.

To counter these shortcomings, Bagga and Baldwin (1998) proposed a B-cubed metric, which first computes a precision and recall for each individual mention, and then takes the weighted sum of these individual precisions and recalls as the final precision and recall. While the B-cubed metric fixes some of the shortcomings of the link-based F-measure, there are problems in itself: for example, the mention precision/recall is computed by comparing entities containing the mention and therefore an entity can be used more than once. The implication of this drawback will be revisited in Section 3.

In the ACE task, a value-based metric called ACE-value is used. The ACE-value is computed by counting the number of false-alarm, the number of miss, and the number of mistaken entities. Each error is associated with a cost factor that depends on things such entity type and entity class. The total cost is the sum of these error cost, which is then normalized against the cost of a nominal system that does not output any entity. The ACE-value is finally computed by subtracting the normalized cost from 1. A perfect coreference system will get a 100% ACE-value while a system outputs no entities will get a 0 ACE-value. A system outputting many erroneous entities could even get a negative ACE-value. The ACE-value is defined at entity level and can distinguish a good system from a bad one; Thus, it satisfies the discriminativity requirement. The ACE-value is, however, hard to interpret the ACE-value, especially after many weights depending on entity type, entity subtype, entity class etc are used.

It is the goal of this paper to develop an evaluation metric that is able to measure the quality of a coreference system – that is, an intuitively better system would get a higher score than a worse system, and is easy to interpret. To this end, we observe that coreference systems are to recognize *entities* and propose a metric called Constrained Entity-Aligned F-Measure (CEAF). At the core of the metric is the optimal one-one map between subsets of reference and system entities: system entities and reference entities are aligned by maximizing the total entity similarity under the constraint that an entity can only be used at most once. Once the total similarity is defined, it is straightforward to compute recall, precision and F-measure. The constraint imposed in the entity alignment makes it impossible to "cheat" the metric: a system outputting too many entities will be penalized in precision while a system outputting two few entities will be penalized in recall. It also has the property that a perfect system gets F-measure 1 while a system outputting no entity or no common mentions gets F-measure 0. The proposed CEAF has a clear meaning: for mention-based CEAF, it reflects the percentage of mentions that are in the correct entities; For entity-based CEAF, it it reflects the percentage of correctly recognized entities.

The rest of the paper is organized as follows. In Section 2, the Constrained Entity-Alignment F-Measure is presented in detail along with an efficient computing algorithm. We also present two entity-pair similarity measures that can be used in CEAF: one is the absolute number of common mentions between two entities, and the other is a "local" mention F-measure between two entities. The two measures lead to mention-based and entity-based CEAF, respectively. In Section 3, we compare the proposed metrics with the MUC link-based metric and ACE-value on both artificial and real data.

2 Constrained Entity-Alignment F-Measure

Some notations are needed before we present the proposed metric and the algorithm to compute the metric.

Let reference entities in a document d be

$$\mathcal{R}(d) = \{R_i : i = 1, 2, \cdots, |\mathcal{R}(d)|\},$$

and system entities be

$$\mathcal{S}(d) = \{S_i : i = 1, 2, \cdots, |\mathcal{S}(d)|\}.$$

To simply typesetting, we will omit the dependency on d when it is clear from context, and write $\mathcal{R}(d)$ as $\mathcal{R}$ and $\mathcal{S}(d)$ as $\mathcal{S}$.

Let $m = \min\{|\mathcal{R}|, |\mathcal{S}|\}$, and $M = max\{|\mathcal{R}|, |\mathcal{S}|\}$. Let $\mathcal{R}_m \subset \mathcal{R}$ and $\mathcal{S}_m \subset \mathcal{S}$ be any subsets with m entities. That is, $|\mathcal{R}_m| = m$ and $|\mathcal{S}_m| = m$. Let $G(\mathcal{R}_m, \mathcal{S}_m)$ be the set of one-one entity maps from $\mathcal{R}_m$ to $\mathcal{S}_m$, and G_m be the set of all possible one-one maps between the size-m subsets of $\mathcal{R}$ and $\mathcal{S}$. Or

$$G(\mathcal{R}_m, \mathcal{S}_m) = \{g : \mathcal{R}_m \mapsto \mathcal{S}_m\},$$
$$G_m = \cup_{(\mathcal{R}_m, \mathcal{S}_m)} G(\mathcal{R}_m, \mathcal{S}_m).$$

Note that for any $g \in G(\mathcal{R}_m, \mathcal{S}_m)$, and any $R_i \in \mathcal{R}_m$ and $S_i \in \mathcal{S}_m$, we have that $R_i \neq R_j$ implies $g(R_i) \neq g(R_j)$, and $g(R_i) \neq g(R_j)$ implies $R_i \neq R_j$. Clearly, there are $m!$ one-one maps from $\mathcal{R}_m$ to $\mathcal{S}_m$ (or $|G(\mathcal{R}_m, \mathcal{S}_m)| = m!$), and $|G_m| = \binom{M}{m} m!$.

Let $\phi(R, S)$ be a "similarity" metric between two entities R and S. For example, $\phi(R, S)$ could be the number of common mentions shared by R and S, and $\phi(R, R)$ the number of mentions in entity R. For any $g \in G_m$, the total similarity $\Phi(g)$ for a map g is the sum of similarities between the aligned entity pairs:

$\Phi(g) = \sum_{R \in \mathcal{R}_m} \phi(R, g(R))$. Given a document d, and its reference entities $\mathcal{R}$ and system entities $\mathcal{S}$, we can find the best alignment maximizing the total similarity:

$$g^* = \arg\max_{g \in G_m} \Phi(g) = \arg\max_{g \in G_m} \sum_{R \in \mathcal{R}_m} \phi(R, g(R)). \quad (1)$$

Let $\mathcal{R}^*_m$ and $\mathcal{S}^*_m = g^*(\mathcal{R}^*_m)$ denote the reference and system entity subsets where g^* is attained, respectively. Then the maximum total similarity is

$$\Phi(g^*) = \sum_{R \in \mathcal{R}^*_m} \phi(R, g^*(R)). \quad (2)$$

Since we can compute the entity self-similarity $\phi(R_i, R_i)$ and $\phi(S_j, S_j)$ as well, we are now ready to define the precision, recall and F-measure as follows:

$$p = \frac{\Phi(g^*)}{\sum_i \phi(S_i, S_i)} \quad (3)$$

$$r = \frac{\Phi(g^*)}{\sum_i \phi(R_i, R_i)} \quad (4)$$

$$F = \frac{2pr}{p + r}. \quad (5)$$

Since the optimal alignment g^* involves only $m = \min\{|\mathcal{R}|, |\mathcal{S}|\}$ reference and system entities, and entities not aligned do not get credit, F-measure (5) penalizes a coreference system that proposes too many (i.e., lower precision) or too few entities (i.e., lower recall), which is a desired property.

Formulae (3)-(5) assumes that there is only one document in the test corpus. Extension to corpus with multiple test documents is trivial: just accumulate statistics on the per-document basis for both denominators and numerators in (3) and (4), and find the ratio of the two.

So far, we have tacitly kept abstract the similarity measure $\phi(R, S)$ for entity pair R and S. We will defer the discussion of this metric to Section 2.2 after first presening an efficient algorithm for computing g^*.

2.1 Computing Optimal Alignment and F-measure

A naive implementation of (1) would enumerate all the possible one-one maps (or alignments) between size-m (recall that $m = \min\{|\mathcal{R}|, |\mathcal{S}|\}$) subsets of $\mathcal{R}$ and size-m subsets of $\mathcal{S}$, and find the best alignment maximizing the similarity. Since this requires computing the similarities between mM entity pairs and there are $|G_m| = \binom{M}{m} m!$ possible one-one maps, the complexity of this implmentation is $O(Mm + \binom{M}{m} m!)$. This is not satisfactory even for a document with a moderate number of entities: it will have about 3.6 million operations for $M = m = 10$, a document with only 10 reference and 10 system entities.

Since ultimately we are interested in finding the top m links among the Mm links such that these m links represent a one-one map between two size-m entity subsets, it is not necessary enumerate all the possible one-one maps, as shown in Algorithm 1.

Algorithm 1 Efficient algorithm to compute the F-measure (5).

Input: reference entities:$\mathcal{R}$; system entities: $\mathcal{S}$
Output: optimal alignment g^*; F-measure (5).
1:Initialize: $g^* = \emptyset$; $\Phi(g^*) = 0$.
2:**For** $i = 1$ **to** $|\mathcal{R}|$
3: **For** $j = 1$ **to** $|\mathcal{S}|$
4: **Compute** $\phi(R_i, S_j)$.
5:Δ=**Sort**($\{\phi(R, S) : R \in \mathcal{R}, S \in \mathcal{S}\}$).
6: $I_r = \emptyset$; $I_s = \emptyset$;
7:**While** $|g^*| < m$
8: **Let** $(R_i, S_j) = \arg\max_{(R,S)}\{\phi(R, S) \in \Delta\}$.
9: **If** $i \notin I_r$ **and** $j \notin I_s$; **then**
10: $g^* = g^* \cup \{(i, j)\}$; $\Phi(g^*) = \Phi(g^*) + \phi(R_i, S_j)$.
11: $I_r = I_r \cup \{i\}$; $I_s = I_s \cup \{j\}$.
12: **End-if**
13: $\Delta = \Delta - \{\phi(R_i, S_j)\}$.
14:**End-While**
15:$\Phi(\mathcal{R}) = \sum_{R \in \mathcal{R}} \phi(R, R)$; $\Phi(\mathcal{S}) = \sum_{S \in \mathcal{S}} \phi(S, S)$.
16:$r = \frac{\Phi(g^*)}{\Phi(\mathcal{R})}$; $p = \frac{\Phi(g^*)}{\Phi(\mathcal{S})}$; $F = \frac{2pr}{p+r}$.
17:**return** g^* and F.

The input to the algorithm is reference entities $\mathcal{R}$ and system entities $\mathcal{S}$. The algorithm returns an approximation [1] of the best one-one map g^* and F-measure in equation (5). Loop from line 2 to 4 computes the similarity between all the possible reference and system entity pairs. The complexity of this loop is in the order $O(Mm)$ (in time of computing $\phi(R, S)$). Line 5 sorts the $|\mathcal{R}||\mathcal{S}|$ scores and stores the sorted result in Δ. The complexity of this step is $O(Mm \log(Mm))$. Two index sets, I_r and I_s, initialized on line 6, are used to store the indices of reference and system entities that have been linked so far. Loop from line 7 to 14 selects the top m links that can form a one-one map between m reference entities and m system entities: line 8 picks the best entity pair at that point (constant time since Δ is sorted); The if-block from line 9 to 11 checks if the best entity pair is new: if neither the reference entity R_i nor the system entity S_j has been used before, the pair is selected and stored in g^*, and the total similar-

[1] It is an approximation because of the greedy nature of the loop between line 7-14 in Algorithm 1. Optimal algorithms exists, e.g., integer programming (thanks to Hans), or best bipartite matching algorithm.

ity $\Phi(g^*)$, index sets I_r and I_s are updated accordingly. Line 13 removes the entity pair (R_i, S_j) from Δ so it will be not examined in the next loop. In the worst case, the loop from line 7 to 14 will be executed $|\mathcal{R}||\mathcal{S}|$ times. Line 15 computes the self-similarity for reference and system entities. The overall complexity of the algorithm is $O\Big(|\mathcal{R}||\mathcal{S}|\log(|\mathcal{R}||\mathcal{S}|)\Big)$.

2.2 Entity Similarity Metric

In this section we consider the entity similarity metric $\phi(R, S)$ defined on an entity pair (R, S). It is desirable that $\phi(R, S)$ is large when R and S are "close" and small when R and S are very different. Some straightfoward choices could be

$$\phi_1(R,S) = \begin{cases} 1, & \text{if } R = S \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

$$\phi_2(R,S) = \begin{cases} 1, & \text{if } R \cap S \neq \emptyset \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

(6) insists that two entity are the same if all the mentions are the same, while (7) goes to the other extreme: two entities are the same if they share at lease one common mention.

(6) does not offer a good granularity of similarity: For example, if $R = \{a, b, c\}$, and one system response is $S_1 = \{a, b\}$, and the other system response $S_2 = \{a\}$, then clearly S_1 is more similar to R than S_2, yet $\phi(R, S_1) = \phi(R, S_2) = 0$. For the same reason, (7) lacks of the desired discriminativity as well.

From the above argument, it is clear that we want to have a metric that can measure the degree to which two entities are similar, not a binary decision. One natural choice is measuring how many common mentions two entities share, and this can be measured by the absolute number or relative number:

$$\phi_3(R,S) = |R \cap S| \quad (8)$$

$$\phi_4(R,S) = \frac{2|R \cap S|}{|R| + |S|}. \quad (9)$$

Metric (8) simply counts the number of common mentions shared by R and S, while (9) is the mention F-measure between R and S, a relative number measuring how similar R and S are. For the abovementioned example,

$$\begin{aligned} \phi_3(R, S_1) &= \phi_3(\{a,b,c\}, \{a,b\}) = 2 \\ \phi_3(R, S_2) &= \phi_3(\{a,b,c\}, \{a\}) = 1 \\ \phi_4(R, S_1) &= \phi_4(\{a,b,c\}, \{a,b\}) = 0.8 \\ \phi_4(R, S_2) &= \phi_4(\{a,b,c\}, \{a\}) = 0.5, \end{aligned}$$

thus both metrics give the desired ranking $\phi_3(R, S_1) > \phi_3(R, S_2),\ \phi_4(R, S_1) > \phi_4(R, S_2)$.

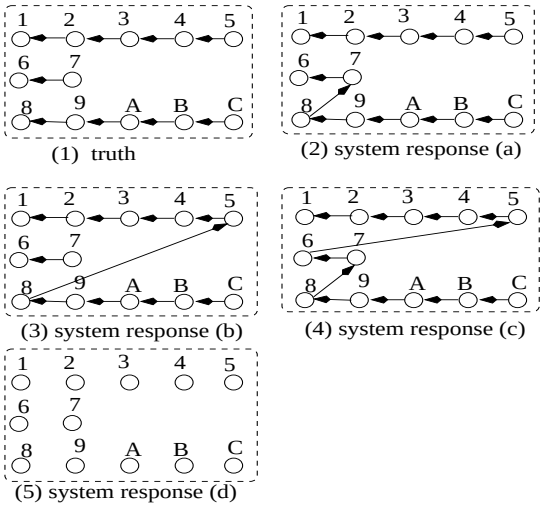

Figure 1: Example entities: (1)truth; (2)system response (a); (3)system response (b); (4)system response (c); (5)system response (d)

If $\phi_3(\cdot,\cdot)$ is adopted in Algorithm 1, $\Phi(g^*)$ is the number of total common mentions corresponding to the best one-one map g^* while the denominators of (3) and (4) are the number of proposed mentions and the number of system mentions, respectively. The F-measure in (5) can be interpreted as the ratio of mentions that are in the "right" entities. Similary, if $\phi_4(\cdot,\cdot)$ is adopted in Algorithm 1, the denominators of (3) and (4) are the number of proposed entities and the number of system entities, respectively, and the F-measure in (5) can be understood as the ratio of correct entities. Therefore, F-measure 5 is called mention-based CEAF and entity-based CEAF when (8) and (9) are used, respectively.

It is worth emphasizing that it is crucial to have the constraint that the similarity between reference entities and system entities is calculated over the best one-one map. We will see examples in Section 3 that misleading results could be produced without the alignment constraint.

3 Comparison with Other Metrics

In this section, we compare the proposed F-measure with the MUC link-based F-measure (and its variation B-cube F-measure) and the more recent ACE-value. The proposed metric has fixed problems associated with the MUC and B-cube F-measure, and has better interpretability than the ACE-value.

3.1 Comparison with the MUC F-measure and B-cube Metric

We use the example in Figure 1 to compare the MUC link-based F-measure, B-cube, and the proposed mention- and entity-based CEAF. In Figure 1, men-

tions are represented in circles and mentions in an entity are connected by arrows. Intuitively, the system response (a) is better than the system response (b) since the latter mixes two big entities, $\{1,2,3,4,5\}$ and $\{8,9,A,B,C\}$, while the former mixes a small entity $\{6,7\}$ with one big entity $\{8,9,A,B,C\}$. System response (b) is clearly better than system response (c) since the latter puts all the mentions into a single entity while (b) has correctly separated the entity $\{6,7\}$ from the rest. The system response (d) is the worst: the system does not link any mentions and outputs 12 single-mention entities.

Table 1 summarizes various F-measures for system response (a) to (d): the first column contains the indices of the system responses found in Figure 1; the second and third columns are the MUC F-measure and B-cubic F-measure respectively; the last two columns are the proposed CEAF F-measures, using the entity similarity metric $\phi_3(\cdot,\cdot)$ and $\phi_4(\cdot,\cdot)$, respectively.

System response	MUC	B-cube	CEAF	
			$\phi_3(\cdot,\cdot)$	$\phi_4(\cdot,\cdot)$
(a)	0.947	0.865	0.833	0.733
(b)	0.947	0.737	0.583	0.667
(c)	0.900	0.545	0.417	0.294
(d)	–	0.400	0.250	0.178

Table 1: Comparison of coreference evaluation metrics

As shown in Table 1, the MUC link-based F-measure fails to distinguish the system response (a) and the system response (b) as the two are assigned the same F-measure. The system response (c) represents a trivial output: all mentions are put in the same entity. Yet the MUC metric will lead to a 100% recall (9 out of 9 reference links are correct) and a 81.2% precision (9 out of 11 system links are correct), which gives rise to a 90% F-measure. It is striking that a "bad" system response gets such a high F-measure. Another problem with the MUC link-based metric is that it is not able to handle single-mention entities, as there is no link for a single mention entity. That is why the entry for system repsonse (d) in Table 1 is empty.

B-cube F-measure ranks the four system responses in Table 1 as desired. This is because B-cube metric (Bagga and Baldwin, 1998) is computed based on mentions (as opposed to links in the MUC F-measure). But B-cube uses the same entity "intersecting" procedure found in computing the MUC F-measure (Vilain et al., 1995), and it sometimes can give counter-intuitive results. To see this, let us take a look at recall and precision for system response (c) and (d) for B-cube metric. Notice that all the reference entities are found after intersecting with the system responsce (c): $\{\{1,2,3,4,5\},\{6,7\},\{8,9,A,B,C\}\}$. Therefore, B-cube recall is 100% (the corresponding precision is $\frac{1}{12} * (10 * \frac{5}{12} + 2 * \frac{2}{12}) = 0.375$). This is counter-intuitive because the set of reference entities is not a subset of the proposed entities, thus the system response should not have gotten a 100% recall. The same problem exists for the system response (d): it gets a 100% B-cube precision (the corresponding B-cube recall is $\frac{1}{12}(5 * \frac{1}{5} + 2 * \frac{1}{2} + 5 * \frac{1}{5}) = 0.25$), but clearly not all the entities in the system response (d) are correct! These numebrs are summarized in Table 2, where columns with R and P represent recall and precision, respectively.

System response	B-cube		CEAF			
	R	P	ϕ_3-R	ϕ_3-P	ϕ_4-R	ϕ_4-P
(c)	1.0	0.375	0.417	0.417	0.196	0.588
(d)	0.25	1.0	0.250	0.250	0.444	0.111

Table 2: Example of counter-intuitive B-cube recall or precision: system repsonse (c) gets 100% recall (column R) while system repsonse (d) gets 100% precision (column P). The problem is fixed in both CEAF metrics.

The counter-intuitive results associated with the MUC and B-cube F-measures are rooted in the procedure of "intersecting" the reference and system entities, which allows an entity to be used more than once! We will come back to this after discussing the CEAF numbers.

From Table 1, we see that both mention-based (column under $\phi_3(\cdot,\cdot)$) CEAF and entity-based ($\phi_4(\cdot,\cdot)$) CEAF are able to rank the four systems properly: system (a) to (d) are increasingly worse. To see how the CEAF numbers are computed, let us take the system response (a) as an example: first, the best one-one entity map is determined. In this case, the best map is: the reference entity $\{1,2,3,4,5\}$ is aligned to the system entity $\{1,2,3,4,5\}$, the reference entity $\{8,9,A,B,C\}$ is aligned to the system $\{6,7,8,9,A,B,C\}$ and the reference entity $\{6,7\}$ is unaligned. The number of common mentions is therefore 10 which results in a mention-based ($\phi_3(\cdot,\cdot)$) recall $\frac{5}{6}$ and precision $\frac{5}{6}$. Since $\phi_4(\{1,2,3,4,5\},\{1,2,3,4,5\}) = 1$, and $\phi_4(\{8,9,A,B,C\},\{6,7,8,9,A,B,C\}) = \frac{10}{12}$, $\Phi(g^*) = 1 + \frac{10}{12}$ (c.f. equation (4) and (3)), and the entity-based F-measure (c.f. equation (9)) is therefore

$$\frac{2 * (1 + \frac{10}{12})}{3 + 2} = \frac{11}{15} = 0.733.$$

CEAF for other system responses are computed similarly.

CEAF recall and precision breakdown for system (c) and (d) are listed in column 4 through 7 of Table 1. As can be seen, neither mention-based nor entity-

based CEAF has the abovementioned problem associated with the B-cube metric, and the recall and precision numbers are more or less compatible with our intuition: for instance, for system (c), based on ϕ_3-CEAF number, we can say that about 41.7% mentions are in the right entity, and based on the ϕ_4-CEAF recall and precision, we can state that about 19.6% of "true" entities are recovered (recall) and about 58.8% of the proposed entities are correct.

A comparison of B-cube and CEAF's computation reveals that the crucial difference is that B-cube allows an entity to be used multiple times while CEAF insists that an entity can not be aligned with more than one entity. Take the system repsonse (c) as an example, intersecting three reference entity in turn with the reference entities produces the same set of reference entities, which leads to a 100% recall. In the intersection step, the system entity is effectively used three times. In contrast, the system entity is aligned to only one reference entity when computing CEAF.

3.2 Comparison On Real Data

We have seen the different behaviors of the MUC F-measure, B-cube F-measure and CEAF on the artificial data. We now compare the MUC F-measure, CEAF, and ACE-value metrics on real data. The B-cube is not implemented as of this writing, and it will not be compared. Comparsion between the MUC F-measure and CEAF is done on the MUC6 coreference test set, while comparison between the CEAF and ACE-value is done on the IBM's 2004 ACE devtest set. The setup reflects the fact that the official MUC scorer and ACE scorer run on their own data format and are not easily portable to the other data set. All the experiments in this section are done on true mentions.

The coreference system is the one used in (Luo et al., 2004). Results in Table 3 are produced by a system trained on the MUC6 training data and tested on the 30 official MUC6 test documents. The test set contains 460 reference entities. The coreference system uses a penalty parameter to balance miss and false alarm errors: the smaller the parameter, the less number of entities will be generated. We vary the parameter from -0.6 to -10, listed in the first column of Table 3, and compare the system performance measured by the MUC F-measure and the proposed mention-based CEAF.

As can be seen, the mention-based CEAF has a clear maximum when the number of proposed entities is close to the truth: at the penlaty value -1.2, the system produces 483 entities, very close to 460, and the ϕ_3-CEAF achieves the maximum 0.768. In contrast, the MUC F-measure increases almost monotonically as the system proposes fewer and fewer entities. In fact, the best system according to the MUC F-measure is the one proposing only 113 entities. This demonstrates a fundamental flaw of the MUC F-measure: the metric intrinsically favors a system producing less number of entities and therefore lacks of discriminativity.

Penalty	#sys-ent	MUC-F	ϕ_3-CEAF
-0.6	561	.851	0.750
-0.8	538	.854	0.756
-0.9	529	.853	0.753
-1	515	.853	0.753
-1.1	506	.856	0.764
-1.2	483	.857	**0.768**
-1.4	448	.863	0.761
-1.5	425	.862	0.749
-1.6	411	.864	0.740
-1.7	403	.865	0.741
-10	113	**.902**	0.445

Table 3: MUC F-measure and mention-based CEAF on the official MUC6 test set. The first column contains the penalty value in decreasing order. The second column contains the number of system-proposed entities. The column under MUC-F is the MUC F-measure while ϕ_3-CEAF is the mention-based CEAF.

Penalty	#sys-ent	ACE-value(%)	ϕ_3-CEAF
0.6	1221	88.5	0.726
0.4	1172	89.1	0.749
0.2	1145	89.4	0.755
0	1105	89.7	0.766
-0.2	1050	**89.7**	0.775
-0.4	1015	89.7	0.780
-0.6	990	89.5	0.782
-0.8	930	88.6	**0.794**
-1	891	86.9	0.780
-1.2	865	86.7	0.778
-1.4	834	85.6	0.769
-1.6	790	83.8	0.761

Table 4: ACE-value and mention-based CEAF on IBM's 2004 devtest set. The first column contains the penalty value in decreasing order. The second column contains the number of system-proposed entities. ACE-values are in percentage.

Now let us turn to ACE-value. Results in Table 4 are produced by a system trained on the ACE 2002 and 2004 training data and tested on IBM's 2004 devtest set. The devtest set contains 853 reference entities. Both ACE-value and the mention-based CEAF penalizes systems over-producing or under-producing entities: ACE-value is maximum when the penalty value is -0.2 and CEAF is maximum when the penalty value

is -0.8. However, the optimal CEAF system produces 930 entities while the optimal ACE-value system produces 1050 entities. Judging from the number of entities, the optimal CEAF system is closer to the "truth" than the counterpart of ACE-value. This is not very surprising since since ACE-value is a weighted metric while CEAF treats each mention and entity equally. As such, the two metrics have very weak correlation.

While we can make statement such as the system with penalty -0.8 puts about 79.4% mentions in right entities, it is hard to interpret the ACE-value numbers.

4 Conclusions

A coreference performance metric – CEAF – is proposed in this paper. The CEAF metric is computed based on the best one-one map between reference entities and system entities. An efficient algorithm is presented to compute the metric. Depending how the entity-pair similarity is calculated, the metric can be mention-based or entity-based. It has been shown that the proposed CEAF metric has fixed problems associated with MUC link-based F-measure and B-cube F-measure. The proposed metric also has better interpretability than ACE-value.

Acknowledgments

This work was partially supported by the Defense Advanced Research Projects Agency and monitored by SPAWAR under contract No. N66001-99-2-8916. The views and findings contained in this material are those of the authors and do not necessarily reflect the position of policy of the Government and no official endorsement should be inferred.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the Linguistic Coreference Workshop at The First International Conference on Language Resources and Evaluation (LREC'98)*, pages 563–566.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proc. of ACL*.

MUC-6. 1995. *Proceedings of the Sixth Message Understanding Conference(MUC-6)*, San Francisco, CA. Morgan Kaufmann.

MUC-7. 1998. *Proceedings of the Seventh Message Understanding Conference(MUC-7)*.

NIST. 2003. The ACE evaluation plan. www.nist.gov/speech/tests/ace/index.htm.

M. Vilain, J. Burger, J. Aberdeen, D. Connolly, , and L. Hirschman. 1995. A model-theoretic coreference scoring scheme. In *In Proc. of MUC6*, pages 45–52.