Tagger Evaluation Given Hierarchical Tag Sets

I. DAN MELAMED[1] and PHILIP RESNIK[2]
[1]*West Group (E-mail: dan.melamed@westgroup.com);* [2]*University of Maryland (E-mail: resnik@umiacs.umd.edu)*

Abstract. We present methods for evaluating human and automatic taggers that extend current practice in three ways. First, we show how to evaluate taggers that assign multiple tags to each test instance, even if they do not assign probabilities. Second, we show how to accommodate a common property of manually constructed "gold standards" that are typically used for objective evaluation, namely that there is often more than one correct answer. Third, we show how to measure performance when the set of possible tags is tree-structured in an IS-A hierarchy. To illustrate how our methods can be used to measure inter-annotator agreement, we show how to compute the kappa coefficient over hierarchical tag sets.

Key words: evaluation, ambiguity resolution, WSD, inter-annotator agreement

1. Introduction

Objective evaluation has been central in advancing our understanding of the best ways to engineer natural language processing systems. A major challenge of objective evaluation is to design fair and informative evaluation metrics, and algorithms to compute those metrics. When the task involves any kind of tagging (or "labeling"), the most common performance criterion is simply "exact match," i.e. exactly matching the right answer scores a point, and no other answer scores any points. This measure is sometimes adjusted for the expected frequency of matches occuring by chance (Carletta, 1996). Resnik and Yarowsky (1997, 1999), henceforth R&Y, have argued that the exact match criterion is inadequate for evaluating word sense disambiguation (WSD) systems.

R&Y proposed a generalization capable of assigning partial credit, thus enabling more informative comparisons on a finer scale. In this article, we present three further generalizations. First, we show how to evaluate non-probabilistic assignments of multiple tags. Second, we show how to accommodate a common property of manually constructed "gold standards" that are typically used for objective evaluation, namely that there is often more than one correct answer. Third, we show how to measure performance when the set of possible tags is tree-structured in an IS-A hierarchy. To illustrate how our methods can be applied to the comparison of human taggers, we show how to compute the kappa coefficient (Siegel and Castellan, 1988) over hierarchical tag sets.

Table I. Hypothetical output of four WSD systems on a test instance, where the correct sense is (2). The exact match criterion would assign zero credit to all four systems. Source: (Resnik and Yarowsky, 1997)

sense of *interest* (in English)	WSD System			
	1	2	3	4
(1) monetary (e.g. on a loan)	0.47	0.85	0.28	1.00
(2) stake or share ⇐ **correct**	0.42	0.05	0.24	0.00
(3) benefit/advantage/sake	0.06	0.05	0.24	0.00
(4) intellectual curiosity	0.05	0.05	0.24	0.00

Our methods depend on the tree structure of the tag hierarchy, but not on the nature of the nodes in it. For example, although these generalizations were motivated by the SENSEVAL exercise (Kilgarriff and Palmer, this issue), the mathematics applies just as well to any tagging task that might involve hierarchical tag sets, such as part-of-speech tagging or semantic tagging (Chinchor, 1998). With respect to word sense disambiguation in particular, questions of whether part-of-speech and other syntactic distinctions should be part of the sense inventory are orthogonal to the issues addressed here.

2. Previous Work

Work on tagging tasks such as part-of-speech tagging and word sense disambiguation has traditionally been evaluated using the exact match criterion, which simply computes the percentage of test instances for which exactly the correct answer is obtained. R&Y noted that, even if a system fails to uniquely identify the correct tag, it may nonetheless be doing a good job of narrowing down the possibilities. To illustrate the myopia of the exact match criterion, R&Y used the hypothetical example in Table I. Some of the systems in the table are clearly better than others, but all would get zero credit under the exact match criterion.

R&Y proposed the following measure, among others, as a more discriminating alternative:

$$\text{Score}(\mathcal{A}) = \Pr_{\mathcal{A}} (c|w, \text{context}(w)) \tag{1}$$

In words, the score for system $\mathcal{A}$ on test instance w is the probability assigned by the system to the correct sense c given w in its context. In the example in Table I, System 1 would get a score of 0.42 and System 4 would score zero.

3. New Generalizations

The generalizations below start with R&Y's premise that, given a probability distribution over tags and a single known correct tag, the algorithm's score should be the probability that the algorithm assigns to the correct tag.

3.1. NON-PROBABILISTIC ALGORITHMS

Algorithms that output multiple tags but do not assign probabilities should be treated as assigning uniform probabilities over the tags that they output. For example, an algorithm that considers tags A and B as possible, but eliminates tags C, D and E for a word with 5 tags in the reference inventory should be viewed as assigning probabilities of 0.5 each to A and B, and probability 0 to each of C, D, and E. Under this policy, algorithms that deterministically select a single tag are viewed as assigning 100% of the probability mass to that one tag, like System 4 in Table I. These algorithms would get the same score from Equation 1 as from the exact match criterion.

3.2. MULTIPLE CORRECT TAGS

Given multiple correct tags for a given word token, the algorithm's score should be the sum of *all* probabilities that it assigns to *any* of the correct tags; that is, multiple tags are interpreted disjunctively. This is consistent with instructions provided to the SENSEVAL annotators: "In general, use disjunction ... where you are unsure which tag to apply" (Krishnamurthy and Nicholls, 1998). In symbols, we build on Equation 1:

$$\text{Score}(\mathcal{A}) = \sum_{t=1}^{C} \Pr_{\mathcal{A}} (c_t | w, \text{context}(w)), \tag{2}$$

where t ranges over the C correct tags. Even if it is impossible to know for certain whether annotators intended a multi-tag annotation as disjunctive or conjunctive, the disjunctive interpretation gives algorithms the benefit of the doubt.

3.3. TREE-STRUCTURED TAG SETS

The same scoring criterion can be used for structured tag sets as for unstructured ones: What is the probability that the algorithm assigns to any of the correct tags? The complication for structured tag sets is that it is not obvious how to compare tags that are in a parent-child relationship. The probabilistic evaluation of taggers can be extended to handle tree-structured tag sets, such as HECTOR (Atkins, 1993), if the structure is interpreted as an IS-A hierarchy. For example, if word sense A.2 is a sub-sense of word sense A, then any word token of sense A.2 *also* IS-A token of sense A.
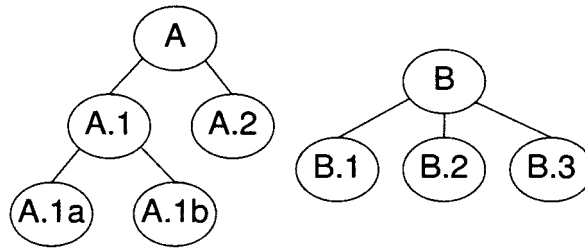
Figure 1. Example tag inventory.

Under this interpretation, the problem can be solved by defining two kinds of probability distributions:

1. Pr(occurrence of parent tag|occurrence of child tag);
2. Pr(occurrence of child tag|occurrence of parent tag).

In a tree-structured IS-A hierarchy Pr(parent|child) = 1, so the first one is easy. The second one is harder, unfortunately; in general, these ("downward") probabilities are unknown. Given a sufficiently large training corpus, the downward probabilities can be estimated empirically. However, in cases of very sparse training data, as in SENSEVAL, such estimates are likely to be unreliable, and may undermine the validity of experiments based on them. In the absence of reliable prior knowledge about tag distributions over various tag-tree branches, we appeal to the maximum entropy principle, which dictates that we assume a uniform distribution of sub-tags for each tag. This assumption is not as bad as it may seem. It will be false in most individual cases, but if we compare tagging algorithms by averaging performance over many different word types, most of the biases should come out in the wash.

Now, how do we use these conditional probabilities for scoring? The key is to treat each non-leaf tag as under-specified. For example, if sense A has just the two subsenses A.1 and A.2, then tagging a word with sense A is equivalent to giving it a probability of one half of being sense A.1 and one half of being sense A.2, given our assumption of uniform downward probabilities. This interpretation applies both to the tags in the output of tagging algorithms and to the manual (correct, reference) annotations.

4. Example

Suppose our sense inventory for a given word is as shown in Figure 1. Under the assumption of uniform downward probabilities, we start by deducing that Pr(A.1|A) = 0.5, Pr(A.1a|A.1) = 0.5 (so Pr(A.1a|A) = 0.25), Pr(B.2|B) = $\frac{1}{3}$, and so on. If any of these conditional probabilities is reversed, its value is always 1. For example, Pr(A|A.1a) = 1. Next, these probabilities are applied in computing Equation 2, as illustrated in Table II.

Table II. Examples of the scoring scheme, for the tag inventory in Figure 1.

Manual Annotation	Algorithm's Output	Score
B	A	0
A	A	1
A	A.1	1
A	A.1b	1
A.1	A	0.5
A.1 and A.2	A	$0.5 + 0.5 = 1$
A.1a	A	0.25
A.1a and B.2	B	$\Pr(B.2|B) = \frac{1}{3}$
A.1a and B.2	A.1	0.5
A.1a and B.2	A.1 and B.2	$0.5 \times 0.5 + 0.5 \times 1 = 0.75$
A.1a and B.2	A.1 and B	$0.5 \times 0.5 + 0.5 \times 0.333 = 0.4$

5. Inter-Annotator Agreement Given Hierarchical Tag Sets

Gold standard annotations are often validated by measurements of inter-annotator agreement. The computation of any statistic that may be used for this purpose necessarily involves comparing tags to see whether they are the same. Again, the question arises as to how to compare tags that are in a parent-child relationship. We propose the same answer as before: Treat non-leaf tags as underspecified.

To compute agreement statistics under this proposal, every non-leaf tag in each annotation is recursively distributed over its children, using uniform downward probabilities. The resulting annotations involve only the most specific possible tags, which can never be in a parent-child relationship. Agreement statistics can then be computed as usual, taking into account the probabilities distributed to each tag.

One of the most common measures of pairwise inter-annotator agreement is the kappa coefficient (Siegel and Castellan, 1988):

$$K = \frac{\Pr(A) - \Pr(E)}{1 - \Pr(E)} \tag{3}$$

where $\Pr(A)$ is the proportion of times that the annotators agree and $\Pr(E)$ is the probability of agreement by chance. Once the annotations are distributed over the leaves L of the tag inventory, these quantities are easy to compute. Given a set of test instances T,

$$\Pr(A) = \frac{1}{|T|} \sum_{t \in T} \sum_{l \in L} Pr(l|\text{annotation}_1(t)) \cdot Pr(l|\text{annotation}_2(t)) \tag{4}$$

$$\Pr(E) = \sum_{l \in L} \Pr(l)^2 \tag{5}$$

Computing these probabilities over just the leaves of the tag inventory ensures that the importance of non-leaf tags is not inflated by double-counting.

6. Conclusion

We have presented three generalizations of standard evaluation methods for tagging tasks. Our methods are based on the principle of maximum entropy, which minimizes potential evaluation bias. As with the R&Y generalization in Equation 1, and the exact match criterion before it, our methods produce scores that can be justifiably interpreted as probabilities. Therefore, decision processes can combine these scores with other probabilities in a maximally informative way by using the axioms of probability theory.

Our generalizations make few assumptions, but even these few assumptions lead to some limitations on the applicability of our proposal. First, although we are not aware of any algorithms that were designed to behave this way, our methods are not applicable to algorithms that *conjunctively* assign more than one tag per test instance. A potentially more serious limitation is our interpretation of tree-structured tag sets as IS-A hierarchies. There has been considerable debate, for example, about whether this interpretation is valid for such well-known tag sets as HECTOR and WordNet.

This work can be extended in a number of ways. For example, it would not be difficult to generalize our methods from trees to hierarchies with multiple inheritance, such as WordNet (Fellbaum, 1998).

References

Atkins, S. "Tools for computer-aided lexicography: the Hector project". In *Papers in Computational Lexicography: COMPLEX '93*. Budapest, 1993.

Carletta, J. "Assessing agreement on classification tasks: the Kappa statistic". *Computational Linguistics* 22(2), 249–254, 1996.

Chinchor, N. (ed.) "Proceedings of the 7th Message Understanding Conference". Columbia, MD: Science Applications International Corporation (SAIC), 1998. Online publication at http://www.muc.saic.com/proceedings/muc_7_toc.html.

Fellbaum, C. (ed.) *WordNet: An Electronic Lexical Database*; Cambridge, MA: MIT Press, 1998.

Krishnamurthy, R. and D. Nicholls. "Peeling an onion: the lexicographer's experience of manual sense-tagging". In *SENSEVAL Workshop*. Sussex, England, 1998.

Resnik, P. and D. Yarowsky. "A perspective on word sense disambiguation methods and their evaluation". In M. Light (ed.): *ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?* Washington, D.C., 1997.

Resnik, P. and D. Yarowsky. "Distinguishing systems and distinguishing senses: New evaluation methods for word sense disambiguation". *Natural Language Engineering*, 5(2), 1999.

Siegel, S. and N.J. Castellan, Jr. *Nonparametric Statistics for the Behavioral Sciences*. Second edition. McGraw-Hill, 1988.