Type Level Clustering Evaluation: New Measures and a POS Induction Case Study

Roi Reichart[1]* **Omri Abend**[2]*† **Ari Rappoport**[2]

[1]ICNC [2]Institute of Computer Science
Hebrew University of Jerusalem
{roiri|omria01|arir}@cs.huji.ac.il

Abstract

Clustering is a central technique in NLP. Consequently, clustering evaluation is of great importance. Many clustering algorithms are evaluated by their success in tagging corpus tokens. In this paper we discuss *type level* evaluation, which reflects class membership only and is independent of the token statistics of a particular reference corpus. Type level evaluation casts light on the merits of algorithms, and for some applications is a more natural measure of the algorithm's quality.

We propose new type level evaluation measures that, contrary to existing measures, are applicable when items are polysemous, the common case in NLP. We demonstrate the benefits of our measures using a detailed case study, POS induction. We experiment with seven leading algorithms, obtaining useful insights and showing that token and type level measures can weakly or even negatively correlate, which underscores the fact that these two approaches reveal different aspects of clustering quality.

1 Introduction

Clustering is a central machine learning technique. In NLP, clustering has been used for virtually every semi- and unsupervised task, including POS tagging (Clark, 2003), labeled parse tree induction (Reichart and Rappoport, 2008), verb-type classification (Schulte im Walde, 2006), lexical acquisition (Davidov and Rappoport, 2006; Davidov and Rappoport, 2008), multilingual document clustering (Montavlo et al., 2006), coreference resolution (Nicolae and Nicolae, 2006) and named entity recognition (Elsner et al., 2009). Consequently, the methodology of clustering evaluation is of great importance. In this paper we focus on external clustering evaluation, i.e., evaluation against manually annotated gold standards, which exist for almost all such NLP tasks. External evaluation is the dominant form of clustering evaluation in NLP, although other methods have been proposed (see e.g. (Frank et al., 2009)).

In this paper we discuss *type level* evaluation, which evaluates the set membership structure created by the clustering, independently of the token statistics of the gold standard corpus. Many clustering algorithms are evaluated by their success in tagging corpus tokens (Clark, 2003; Nicolae and Nicolae, 2006; Goldwater and Griffiths, 2007; Gao and Johnson, 2008; Elsner et al., 2009). However, in many cases a type level evaluation is the natural one. This is the case, for example, when a POS induction algorithm is used to compute a tag dictionary (the set of tags that each word can take), or when a lexical acquisition algorithm is used for constructing a lexicon containing the set of frames that a verb can participate in, or when a sense induction algorithm computes the set of possible senses of each word. In addition, even when the goal is corpus tagging, a type level evaluation is highly valuable, since it may cast light on the relative or absolute merits of different algorithms (as we show in this paper).

Clustering evaluation has been extensively investigated (Section 3). However, the discussion centers around the monosemous case, where each item belongs to exactly one cluster, although polysemy is the common case in NLP.

The contribution of the present paper is as follows. First, we discuss the issue of type level evaluation and explain why even in the monosemous case a token level evaluation presents a skewed

* Both authors equally contributed to this paper.

† Omri Abend is grateful to the Azrieli Foundation for the award of an Azrieli Fellowship.

picture (Section 2). Second, we show for the common polysemous case why adapting existing information-theoretic measures to type level evaluation is not natural (Section 3). Third, we propose new mapping-based measures and algorithms to compute them (Section 4). Finally, we perform a detailed case study with part-of-speech (POS) induction (Section 5). We compare seven leading algorithms, showing that token and type level measures can weakly or even negatively correlate. This shows that type level evaluation indeed reveals aspects of a clustering solution that are not revealed by the common tagging-based evaluation.

Clustering is a vast research area. As far as we know, this is the first NLP paper to propose type level measures for the polysemous case.

2 Type Level Clustering Evaluation

This section motivates why both type and token level external evaluations should be done, even in the monosemous case.

Clustering algorithms compute a set of *induced clusters* (a *clustering*). Some algorithms directly compute a clustering, while some others produce a tagging of corpus tokens from which a clustering can be easily derived. A clustering is *monosemous* if each item is allowed to belong to a single cluster only, and *polysemous* otherwise. An *external* evaluation is one which is based on a comparison of an algorithm's result to a gold standard. In this paper we focus solely on external evaluation, which is the most common evaluation approach in NLP.

Token and type level evaluations reflect different aspects of a clustering. External token level evaluation assesses clustering quality according to the clustering's accuracy on a given manually annotated corpus. This is certainly a useful evaluation measure, e.g. when the purpose of the clustering algorithm is to annotate a corpus to serve as input to another application.

External type level evaluation views the computed clustering as a set membership structure and evalutes it independently of the token statistics in the gold standard corpus. There are two main cases in which this is useful. First, a type level evaluation can be the natural one in light of the problem itself. For example, if the purpose of the clustering algorithm is to automatically build a lexicon (e.g., VerbNet (Kipper et al., 2000)), then the lexicon structure itself should be evaluated. Second, it may be valuable to decouple corpus statistics from the induced clustering when the latter is to be used for annotating corpora that exhibit different statistics. In other words, if we evaluate an algorithm that will be invoked on a diverse set of corpora having different token statistics, a type level evaluation might provide a better picture (or at least a complementary one) on the quality of the clustering algorithm.

To motivate type level evaluation, consider POS induction, which exemplifies both cases above. Clearly, a word form may belong to several parts of speech (e.g., 'contrast' is both a noun and a verb, 'fast' is both an adjective and an adverb, 'that' can be a determiner, conjunction and adverb, etc.). As an evaluation of a POS induction algorithm, it is natural to evaluate the lexicon it generates, even if the main goal is to annotate a corpus. The lexicon lists the possible POS tags for each word, and thus its evaluation is a polysemous type level one.

Even if we ignore polysemy, type level evaluation is useful for a POS induction algorithm used to tag a corpus. There are POS classes whose members are very frequent, e.g., determiners and prepositions. Here, a very small number of word types usually accounts for a large portion of corpus tokens. For example, in the WSJ Penn Treebank (Marcus et al., 1993), there are 43,740 word types and over 1M word tokens. Of the types, 88 are tagged as prepositions. These types account for only 0.2% of the types, but for as many as 11.9% of the tokens. An algorithm which is accurate only on prepositions would do much better in a token level evaluation than in a type level one.

This phenomenon is not restricted to prepositions or English. In the WSJ corpus, determiners account for 0.05% of the types but for 9.8% of the tokens. In the German NEGRA corpus (Brants, 1997), the article class (both definite and indefinite) accounts for 0.04% of the word types and for 12.5% of the word tokens, and the coordinating conjunctions class accounts for 0.05% of the word types but for 3% of the tokens.

The type and token behavior differences result from the Zipfian distribution of word tokens to word types (Mitzenmacher, 2004). Since the word frequency distribution is Zipfian, any clustering algorithm that is accurate only on a small number of frequent words (not necessarily members of a particular class) would perform well in a token level evaluation but not in a type one. For example, the most frequent 100 word types (regardless of POS class) in WSJ (NEGRA) account for 43.9% (41.3%) of the tokens in the corpus. These words appear in 32 out of the 34 non-punctuation POS classes in WSJ and in 38 out of the 51 classes in NEGRA.

Other natural language entities also demonstrate Zipfian distribution of tokens to types. For example, the distribution of syntactic categories in parse tree constituents is Zipfian, as shown in (Reichart and Rappoport, 2008) for English, German and Chinese corpora. Thus, the distinction between token and type level evaluation is important also for grammar induction algorithms.

It may be argued that a token level evaluation is sufficient since it already reflects type information. In this paper we demonstrate that this is not the case, by showing that they correlate weakly or even negatively in an important NLP task.

3 Existing Clustering Evaluation Measures

Clustering evaluation is challenging. Many measures have been proposed in the past decades (Pfitzner et al., 2008). In this section, we briefly survey the three main types: mapping based, counting pairs, and information theoretic measures, and motivate our decision to focus on the first in this paper.

Mapping based measures are based on a post-processing step in which each induced cluster is mapped to a gold class (or vice versa). The standard mappings are greedy many-to-one (M-1) and greedy one-to-one (1-1). Several measures which rely on these mappings were proposed. The most common and perhaps the simplest one is accuracy, which computes the fraction of items correctly clustered under the mapping. Other measures include: L (Larsen, 1999), D (Van Dongen, 2000), misclassification index (MI) (Zeng et al., 2002), H (Meila and Heckerman, 2001), clustering F-measure (Fung et al., 2003) and micro-averaged precision and recall (Dhillon et al., 2003). In Section 4 we show why existing mapping-based measures cannot be applied to the polysemous type case and present new mapping-based measures for this case.

Counting pairs measures are based on a combinatorial approach which examines the number of data element pairs that are clustered similarly in the reference and proposed clustering. Among these are Rand Index (Rand, 1971), Adjusted Rand Index (Hubert and Arabie, 1985), Γ statistic (Hubert and Schultz, 1976), Jaccard (Milligan et al., 1983), Fowlkes-Mallows (Fowlkes and Mallows, 1983) and Mirkin (Mirkin, 1996). Schulte im Walde (2006) used such a measure for type level evaluation of monosemous verb type clustering.

Meila (2007) described a few problems with such measures. A serious one is that their values are unbounded, making it hard to interpret their results. This can be solved by adjusting their values to lie in $[0, 1]$, but even adjusted measures suffer from severe distributional problems, limiting their usability in practice. We thus do not address counting pairs measures in this paper.

Information-theoretic (IT) measures. IT measures assume that the items in the dataset are taken from a known distribution (usually the uniform distribution), and thus the gold and induced clusters can be treated as random variables. These measures utilize a co-occurrence matrix I between the gold and induced clusters. We denote the induced clustering by K and the gold clustering by C. I_{ij} contains the number of items in the intersection of the i-th gold class and the j-th induced cluster. When assuming the uniform distribution, the probability of an event (a gold class c or an induced cluster k) is its relative size, so $p(c) = \sum_{k=1}^{|K|} \frac{I_{ck}}{N}$ and $p(k) = \sum_{c=1}^{|C|} \frac{I_{ck}}{N}$ (N is the total number of clustered items).

Under this assumption we define the entropies and the conditional entropies:

$$H(C) = -\sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} I_{ck}}{N} log \frac{\sum_{k=1}^{|K|} I_{ck}}{N}$$

$$H(C|K) = -\sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{I_{ck}}{N} log \frac{I_{ck}}{\sum_{c=1}^{|C|} I_{ck}}$$

$H(K)$ and $H(K|C)$ are defined similarly.

In Section 5 we use two IT measures for token level evaluation, V (Rosenberg and Hirschberg, 2007) and NVI (Reichart and Rappoport, 2009) (a normalized version of VI (Meila, 2007)). The appealing properties of these measures have been extensively discussed in these references; see also (Pfitzner et al., 2008). V and NVI are defined as follows:

$$h = \begin{cases} 1 & H(C) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & H(C) \neq 0 \end{cases}$$

$$c = \begin{cases} 1 & H(K) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & H(K) \neq 0 \end{cases}$$

$$V = \frac{2hc}{h + c}$$

$$NVI(C,K) = \begin{cases} \frac{H(C|K)+H(K|C)}{H(C)} & H(C) \neq 0 \\ H(K) & H(C) = 0 \end{cases}$$

In the monosemous case (type or token), the application of the measures described in this section to type level evaluation is straightforward. In the polysemous case, however, they suffer from serious shortcomings.

Consider a case in which each item is assigned exactly r gold clusters and each gold cluster has the exact same number of items (i.e., each has a size of $\frac{l \cdot r}{|C|}$, where l is the number of items). Now, consider an induced clustering where there are $|C|$ induced clusters ($|K| = |C|$) and each item is assigned to all induced clusters. The co-occurrence matrix in this case should have identical values in all its entries. Even if we allow the weight each item contributes to the matrix to depend on its gold and induced entry sizes, the situation will remain the same. This is because all items have the exact same entry size and both gold and induced clusterings have uniform cluster sizes.

In this case, the random variables defined by the induced and gold clustering assignments are independent (this easily follows from the definition of independent events, since the joint probability is the multiplication of the marginals). Hence, $H(K|C) = H(K)$ and $H(C|K) = H(C)$, and both V and NVI obtain their worst possible values[1]. However, the score should surely depend on r (the size of each word's gold entry). Specifically, when $r = |C|$ we get that the induced and gold clusterings are identical. This case should not get the worst score, and it should definitely score higher than the case in which $r = 1$, where K is dramatically different from C.

The problem can in theory be solved by providing the number of clusters per item as an input to the algorithm. However, in NLP this is unrealistic (even if the total number of clusters can be provided) and the number should be determined by the algorithm. We therefore do not consider IT-based measures in this paper, deferring them to future work.

4 Mapping Based Measures for Polysemous Type Evaluation

In this section we present new type level evaluation measures for the polysemous case. As we show below, these measures do not suffer from the problems discussed for IT measures in Section 3.

All measures are mapping-based: first, a mapping between the induced and gold clusters is performed, and then a measure E is computed. As is common in the clustering evaluation literature (Section 3), we use M-1 and 1-1 greedy mappings, defined to be those that maximize the corresponding measure E.

Let $C = \{c_1, ..., c_n\}$ be the set of gold classes and $K = \{k_1, ..., k_m\}$ be the set of induced clusters. Denote the number of words types by l. Let $A_i \subset C, B_i \subset K, i = 1...l$ be the set of gold classes and set of induced clusters for each word. The polysemous nature of task is reflected by the fact that A_i and B_i are subsets, rather than members, of C and K respectively.

Our measures address quality from two persectives, that of the individual items clustered (Section 4.1) and that of the clusters (Section 4.2). Item-based measures especially suit evaluation of clustering quality for the purpose of lexicon induction, and have no counterpart in the monosemous case. Cluster-based measures are a direct generalization of existing mapping based measures to the polysemous case.

The difficulty in designing item-based and cluster-based measures is that the number of clusters assigned to each item is determined by the clustering algorithm. Below we show how to overcome this.

4.1 Item-Based Evaluation

For a given mapping $h : K \rightarrow C$, denote $h(B_i) = \{h(x) : x \in B_i\}$. A fundamental quantity for item-based evaluation is the number of correct clusters for each item (word type) under this mapping, denoted by IM_i (IM stands for 'item match'):

$$IM_i = |A_i \cap h(B_i)|$$

The total item match IM is defined to be:

$$IM = \sum_{i=1}^{l} IM_i = \sum_{i=1}^{l} |A_i \cap h(B_i)|$$

In the monosemous case, IM is normalized by the number of items, yielding an accuracy score. Applying a similar definition in the polysemous case, normalizing instead by the total number of gold clusters assigned to the items, can be easily manipulated. Even a clustering which has the correct number of induced clusters (equal to the number of gold classes) but which assigns each item to

[1] V values are in $[0, 1]$, 0 being the worst. NVI obtains its highest and worst possible value, $1 + \frac{log(|K|)}{H(C)}$.

all induced clusters, receives a perfect score under both greedy M-1 and 1-1 mappings. This holds for any induced clustering for which $\forall i, A_i \subset h(B_i)$. Note that using a mapping from C to K (or a combination of both directions) would exhibit the same problem.

To overcome the problem, we use the harmonic average of two normalized terms (F-score). We use two average variants, micro and macro. Macro average computes the total number of matches over all words and normalizes in the end. Recall (R), Precision (P) and their harmonic average (F-score) are accordingly defined:

$$R = \frac{IM}{\sum_{i=1}^{l} |A_i|} \qquad P = \frac{IM}{\sum_{i=1}^{l} |h(B_i)|}$$

$$MacroI = \frac{2RP}{R+P} =$$

$$= \frac{2IM}{\sum_{i=1}^{l} |A_i| + \sum_{i=1}^{l} |h(B_i)|} = F(h) \cdot \sum_{i=1}^{l} IM_i$$

$F(h)$ is a constant depending on h. As all items are equally weighted, those with larger gold and induced entries have more impact on the measure.

The micro average, aiming to give all items an equal status, first computes an F-score for each item and then averages over them. Hence, each item contributes at most 1 to the measure. This *MicroI* measure is given by:

$$R_i = \frac{IM_i}{|A_i|} \quad P_i = \frac{IM_i}{|h(B_i)|} \quad F_i = \frac{2R_iP_i}{R_i+P_i} = \frac{2IM_i}{|A_i|+|h(B_i)|}$$

$$MicroI = \frac{1}{l}\sum_{i=1}^{l} F_i = \frac{1}{l}\sum_{i=1}^{l} \frac{2IM_i}{|A_i| + |h(B_i)|} =$$

$$= \frac{1}{l}\sum_{i=1}^{l} w_i(h) \cdot IM_i$$

Where $w_i(h)$ is a weight depending on h but also on i.

For both measures, the maximum score is 1. It is obtained if and only if $A_i = h(B_i)$ for every i.

In 1-1 mapping, when the number of induced clusters is larger than the number of gold clusters, some of the induced clusters are not mapped. To preserve the nature of 1-1 mapping that punishes for excessive clusters[2], we define $|h(B_i)|$ to be equal to $|B_i|$ even for these unmapped clusters.

Recall that any induced clustering in which $\forall i, A_i \subset h(B_i)$ gets the best score under a greedy mapping with the accuracy measure. In MacroI and MicroI the obtained recalls are perfect, but the precision terms reflect deviation from the correct solution.

In the example in Section 3 showing an unreasonable behavior of IT-based measures, the score depends on r for both MacroI and MicroI. With our new measures, recall is always 1, but precision is $\frac{r}{n}$. This is true both for 1-1 and M-1 mappings. Hence, the new measures show reasonable behavior in this example for all r values.

MicroI was used in (Dasgupta and Ng, 2007) with a manually compiled mapping. Their mapping was not based on a well-defined scheme but on a heuristic. Moreover, providing a manual mapping might be impractical when the number of clusters is large, and can be inaccurate, especially when the clustering is not of very high quality.

In the following we discuss how to compute the 1-1 and M-1 greedy mappings for each measure.

1-1 Mapping. We compute h by finding the maximal weighted matching in a bipartite graph. In this graph one side represents the induced clusters, the other represents the gold classes and the matchings correspond to 1-1 mappings. The problem can be efficiently solved by the Kuhn-Munkres algorithm (Kuhn, 1955; Munkres, 1957).

To be able to use this technique, edge weights must not depend upon h. In 1-1 mapping, $|h(B_i)| = |B_i|$, and therefore $F(h) = F$ and $w_i(h) = w_i$. That is, both quantities are independent of h[3]. For MacroI, the weight on the edge between the s-th gold class and the j-th induced cluster is: $W(e_{sj}) = \sum_{i=1}^{l} F \cdot I_{s \in A_i} I_{j \in B_i}$. For MicroI it is: $W(e_{sj}) = \sum_{i=1}^{l} w_i \cdot I_{s \in A_i} I_{j \in B_i}$. $I_{s \in A_i}$ is 1 if $s \in A_i$ and 0 otherwise.

M-1 Mapping. There are two problems in applying the bipartite graph technique to finding an M-1 mapping. First, under such mapping $w_i(h)$ and $F(h)$ do depend on h. The problem may be solved by selecting some constant weighting scheme. However, a more serious problem also arises.

Consider a case in which an item x has a gold entry $\{C_1\}$ and an induced entry $\{K_1, K_2\}$. Say the chosen mapping mapped both K_1 and K_2 to C_1. By summing over the graph's edges selected by the mapping, we add weight ($F(h)$ for MacroI and $w_i(h)$ for MicroI) both to the edge between K_1 and C_1 and to the edge between K_2 and C_1. However, the item's IM_i is only 1. This prohibits

[2] And to allow us to compute it accurately, see below.

[3] Consequently, the increase in MacroI and MicroI following an increase of 1 in an item's gold/induced intersection size (IM_i) is independent of h.

the use of the bipartite graph method for the M-1 case.

Since we are not aware of any exact method for solving this problem, we use a hill-climbing algorithm. We start with a random mapping and a random order on the induced clusters. Then we iterate over the induced clusters and map each of them to the gold class which maximizes the measure given that the rest of the mapping remains constant. We repeat the process until no improvement to the measure can be obtained by changing the assignment of a single induced cluster. Since the score depends on the initial random mapping and random order, we repeat this process several times and choose the maximum between the obtained scores.

4.2 Cluster-Based Evaluation

The cluster-based evaluation measures we propose are a direct generalization of existing monosemous mapping based measures to the polysemous type case.

For a given mapping $h : K \to C$, we define $\bar{h} : K^h \to C$. K^h is defined to be a clustering which is obtained by performing set union between every two clusters in K that are mapped to the same gold cluster. The resulting $\bar{h}$ is always 1-1. We denote $|K^h| = m^h$.

Our motivation for using $\bar{h}$ in the definition of the measures instead of h is to stay as close as possible to accuracy, the most common mapping-based measure in the monosemous case. M-1 (monosemous) accuracy does not punish for splitting classes. For instance, in a case where there is a gold cluster c_i and two induced clusters k_1 and k_2 such that $c_i = k_1 \cup k_2$, the M-1 accuracy is the same as in the case where there is one cluster k_1 such that $c_i = k_1$. M-1 accuracy, despite its indifference to splitting, was shown to reflect better than 1-1 accuracy the clustering's applicability for subsequent applications (at least in some contexts) (Headden III et al., 2008).

Recall that in item-based evaluation, IM_i measures the intersection between the induced and gold entries of each item. Therefore, the set union operation is not needed for that case, since when an item appears in two induced clusters that are mapped to the same gold cluster, its IM_i is increased only by 1.

A fundamental quantity for cluster-based evaluation is the intersection between each induced cluster and the gold class to which it is mapped. We denote this value by CM_j (CM stands for 'cluster match'):

$$CM_j = |k_j \cap \bar{h}(k_j)|$$

The total intersection (CM) is accordingly defined to be:

$$CM = \sum_{j=1}^{m^h} CM_j = \sum_{j=1}^{m^h} |k_j \cap \bar{h}(k_j)|$$

As with the item-based evaluation (Section 4.1), using CM or a derived accuracy as a measure is problematic. A clustering that assigns n induced classes to each word (n is the number of gold classes) will get the highest possible score under every greedy mapping (1-1 or M-1), as will any clustering in which $\forall i, A_i \subset h(B_i)$.

As in the item-based evaluation, a possible solution is based on defining recall, precision and F-score measures, computed either in the micro or in the macro level. The macro cluster-based measure turns out to be identical to the macro item-based measure MacroI[4].

The following derivation shows the equivalence for the 1-1 case. The M-1 case is similar. We note that $h = \bar{h}$ in the 1-1 case and we therefore exchange them in the definition of CM. It is enough to show that $CM = IM$, since the denominator is the same in both cases:

$$\begin{aligned} CM &= \textstyle\sum_{j=1}^{m} |k_j \cap h(k_j)| = \\ &= \textstyle\sum_{j=1}^{m} \sum_{i=1}^{l} I_{i \in k_j} I_{i \in h(k_j)} = \\ &= \textstyle\sum_{i=1}^{l} \sum_{j=1}^{m} I_{i \in k_j} I_{i \in h(k_j)} = \\ &= \textstyle\sum_{i=1}^{l} |A_i \cap h(B_i)| = IM \end{aligned}$$

The micro cluster-based measures are defined:

$$R_j = \frac{CM_j}{|\bar{h}(k_j)|} \quad P_j = \frac{CM_j}{|k_j|} \quad F_j = \frac{2R_jP_j}{R_j+P_j}$$

The micro cluster measure MicroC is obtained by taking a weighted average over the F_j's:

$$MicroC = \sum_{k \in K^h} \frac{|k|}{N^*} F_k$$

Where $N^* = \sum_{z \in K^h} |z|$ is the number of clustered items after performing the set union and including repetitions. If, in the 1-1 case where $m > n$, an induced cluster is not mapped, we define $F_k = 0$. A definition of the measure using a reverse mapping (i.e., from C to K) would have used a weighted average with weights proportional to the gold classes' sizes.

[4] Hence, we have six type level measures: MacroI (which is equal to MacroC), MicroI, and MicroC, each of which in two versions, M-1 and 1-1.

The definition of $\bar{h}$ causes a similar computational difficulty as in the M-1 item-based measures. Consequently, we apply a hill climbing algorithm similar to the one described in Section 4.1.

The 1-1 mapping is computed using the same bipartite graph method described in Section 4.1. The graph's vertices correspond to gold and induced clusters and an edge's weight is the F-score between the class and cluster corresponding to its vertices times the cluster's weight ($|k|/N^*$).

5 Evaluation of POS Induction Models

As a detailed case study for the ideas presented in this paper, we apply the various measures for the POS induction task, using seven leading POS induction algorithms.

5.1 Experimental Setup

POS Induction Algorithms. We experimented with the following models: ARR10 (Abend et al., 2010), Clark03 (Clark, 2003), GG07 (Goldwater and Griffiths, 2007), GJ08 (Gao and Johnson, 2008), and GVG09 (Van Gael et al., 2009) (three models). Additional recent good results for various variants of the POS induction problem are described in e.g., (Smith and Eisner, 2004; Graça et al., 2009).

Clark03 and ARR10 are monosemous algorithms, allowing a single cluster for each word type. The other algorithms are polysemous. They perform sequence labeling where each token is tagged in its context, and different tokens (instances) of the same type (word form) may receive different tags.

Data Set. All models were tested on sections 2-21 of the PTB-WSJ, which consists of 39832 sentences, 950028 tokens and 39546 unique types. Of the tokens, 832629 (87.6%) are not punctuation marks.

Evaluation Measures. Type level evaluation used the measures MacroI (which is equal to MacroC), MicroI and MicroC both with greedy 1-1 and M-1 mappings as described in Section 4. The type level gold (induced) entry is defined to be the set of all gold (induced) clusters with which it appears.

For the token level evaluation, six measures are used (see Section 3): accuracy with M-1 and 1-1 mappings, NVI, V, H(C|K) and H(K|C), using e as the logarithm's base. We use the full WSJ POS tags set excluding punctuation[5].

Punctuation. Punctuation marks occupy a large volume of the corpus tokens (12.4% in our experimental corpus), and are easy to cluster. Clustering punctuation marks thus greatly inflates token level results. To study the relationship between type and token level evaluations in a focused manner, we excluded punctuation from the evaluation (they are still used during training, so algorithms that rely on them are not harmed).

Number of Induced Clusters. The number of gold POS tags in WSJ is 45, of which 11 are punctuation marks. Therefore, for the ARR10 and Clark03 models, 34 clusters were induced. For GJ08 we received the output with 45 clusters. The iHMM models of GVG09 determine the number of clusters automatically (resulting in 47, 91 and 192 clusters, see below). For GG07, our computing resources did not enable us to induce 45 clusters and we therefore used 17[6]. Our focus in this paper is to study the type vs. token distinction rather than to provide a full scope comparison between algorithms, for which more clustering sizes would need to be examined.

Configurations. We ran the ARR10 tagger with the configuration detailed in (Abend et al., 2010). For Clark03, we ran his *neyessenmorph* model[7] 10 times (using an unknown words threshold of 5) and report the average score for each measure. The models of GVG09 were run in the three configurations reported in their paper: one with a Dirichlet process prior and fixed parameters, another with a Pittman-Yore prior with fixed parameters, and a third with a Dirichlet process prior with parameters learnt from the data. All five models were run in an optimal configuration.

We obtained the code of Goldwater and Griffiths' BHMM model and ran it for 10K iterations with an annealing technique for parameter estimation. That was the best parameter estimation technique available to us. This is the first time that this model is evaluated on such a large experimental corpus, and it performed well under these conditions.

The output of the model of GJ08 was sent to us by the authors. The model was run on sec-

[5] We use all WSJ tokens in the training stage, but omit punctuation marks during evaluation.

[6] The 17 most frequent tags cover 94% of the word instances and more than 99% of the word types in the WSJ gold standard tagging.

[7] www.cs.rhul.ac.uk/home/alexc/RHUL/Downloads.html

tions 2-21 of the WSJ-PTB using significantly inferior computing resources compared to those used for producing the results reported in their paper. While this model cannot be compared to the aforementioned six models due to the suboptimal configuration, we evaluate its output using our measures to get a broader variety of experimental results[8].

5.2 Results and Discussion

Table 1 presents the scores of the compared models under all evaluation measures (six token level, six type level). What is important here to note are the differences between type and token level evaluations for the algorithms. We are mainly interested in two things: (1) seeing how relative rankings change in the two evaluation types, thus showing that the two types are not highly correlated and are both useful; and (2) insights gained by using a type level evaluation in addition to the usual token level one.

Note that the table should not be used to deduce which algorithm is the 'best' for the task, even according to a single evaluation type. This is because, as explained above, the algorithms do not induce the same number of clusters and this affects their results.

Results indicate that type level evaluation reveals aspects of the clustering quality that are not expressed in the token level. For the Clark03 model the disparity is most apparent. While in the token level it performs very well (better than the polysemous algorithms for the 1-1, V and NVI token level measures), in the type level it is the second worst in the item-based 1-1 scores and the worst in the M-1 scores.

Here we have a clear demonstration of the value of type level evaluation. The Clark03 algorithm is assessed as excellent using token level evaluation (second only to ARR10 in M-1, 1-1, V and NVI), and only a type level one shows its relatively poor type performance. Although readers may think that this is natural due to the algorithm's monosemous nature, this is not the case, since the monosemous ARR10 generally ranked first in the type level measures (more on this below).

The disparity is also observed for polysemous algorithms. The GG07 model's token level scores are mediocre, while in the type level MicroC 1-1 measure this model is the best and in the type level MicroI and MacroI 1-1 measures it is the second best.

Monosemous vs. polysemous algorithms. The table shows that the ARR10 model achieves the best results in most type and token level evaluation measures. The fact that this monosemous algorithm outperforms the polysemous ones, even in a type level evaluation, may seem strange at first sight but can be explained as follows. Polysemous tokens account for almost 60% of the corpus (565K out of 950K), so we could expect that a monosemous algorithm should do badly in a token-level evaluation. However, for most of the polysemous tokens the polysemy is only weakly present in the corpus[9], so it is hard to detect even for polysemous algorithms. Regarding types, polysemous types constitute only 16.6% of the corpus types, so a monosemous algorithm which is quite good in assigning types to clusters has a good chance of beating polysemous algorithms in a type level evaluation.

Hence, monosemous POS induction algorithms are not at such a great disadvantage relative to polysemous ones. This observation, which was fully motivated by our type level case study, might be used to guide future work on POS induction, and it thus serves as another demonstration for the utility of type level evaluation.

Hill climbing algorithm. For the type level measures with greedy M-1 mapping, we used the hill-climbing algorithm described in Section 4. Recall that the mapping to which our algorithm converges depends on its random initialization. We therefore ran the algorithm with 10 different random initializations and report the obtained maximum for MacroI, MicroI and MicroC in Table 1. The different initializations caused very little fluctuation: not more than 1% in the 9 (7) best runs for the item-based (MicroC) measures. We take this result as an indication that the obtained maximum is a good approximation of the global maximum.

We tried to improve the algorithm by selecting an intelligent initialization heuristic. We used the M-1 mapping obtained by mapping each induced cluster to the gold class with which it has the high-

[8] We would like to thank all authors for sending us the data.

[9] Only about 27% of the tokens are instances of words that are polysemous but not weakly polysemous (we call a word *weakly polysemous* if more than 95% of its instances (tokens) are tagged by the same tag).

	Token Level Evaluation						Type Level Evaluation					
							MacroI		MicroI		MicroC	
	M-1	1-1	NVI	V	H(C|K)	H(K|C)	M-1	1-1	M-1	1-1	M-1	1-1
ARR10	**0.675**	**0.588**	**0.809**	**0.608**	1.041	**1.22**	0.579	**0.444**	**0.596**	**0.455**	**0.624**	0.403
Clark03	0.65	0.484	0.887	0.586	1.04	1.441	0.396	0.301	0.384	0.288	0.463	0.347
GG07	0.5	0.415	0.989	0.479	1.523	1.241	0.497	0.405	0.461	0.398	0.563	**0.445**
GVG09(1)	0.51	0.444	1.033	0.477	1.471	1.409	0.513	0.354	0.436	0.352	0.486	0.33
GVG09(2)	0.591	0.484	0.998	0.529	1.221	1.564	0.637	0.369	0.52	0.373	0.548	0.32
GVG09(3)	0.668	0.368	1.132	0.534	**0.978**	2.18	**0.736**	0.280	0.558	0.276	0.565	0.199
GJ08*	0.605	0.383	1.09	0.506	1.231	1.818	0.467	0.298	0.446	0.311	0.561	0.291

Table 1: Token level (left columns) and type level (right columns) results for seven POS induction algorithms (rows) (see text for details). Token and type level performance are weakly correlated and complement each other as evaluation measures. ARR10, a monosemous algorithm, yields the best results in most measures. (GJ08* results are different from those reported in the original paper because it was run with weaker computing resources than those used there.)

est weight edge in the bipartite graph. Recall from Section 4.1 that this is a reasonable approximation of the greedy M-1 mapping. Again, we ran it for the three type level measures for 10 times with a random update order on the induced clusters. This had only a minor effect on the final scores.

Number of clusters. Previous work (Reichart and Rappoport, 2009) demonstrated that in data sets where a relatively small fraction of the gold classes covers most of the items, it is reasonable to choose this number to be the number of induced clusters. In our experimental data set, this number (the 'prominent cluster number') is around 17 (see Section 5.1). Up to this number, increasing the number of clusters is likely to have a positive effect on token level M-1, 1-1, H(C|K), and H(K|C) scores. Inducing a larger number of clusters, however, is likely to positively affect M-1 and H(C|K) but to have a negative effect on 1-1 and H(K|C).

This tendency is reflected in Table 1. For the GG07 model the number of induced clusters, 17, approximates the number of prominent clusters and is lower than the number of induced clusters of the other models. This is reflected by its low token level M-1 and H(C|K) performance and its high quality H(K|C) and NVI token level scores. The GVG (1)-(3) models induced 47, 91 and 192 clusters respectively. This might explain the high token level M-1 and H(C|K) performance of GVG(3), as well as its high M-1 type level performance, compared to its mediocre scores in other measures.

The item based measures. The table indicates that there is no substantial difference between the two item based type level scores with 1-1 mapping. The definitions of MacroI and MicroI imply that if $|A_i| + |h(B_i)|$ (which equals $|A_i| + |B_i|$ under a 1-1 mapping) is constant for all word types, then a clustering will score equally on both 1-1 type measures. Indeed, in our experimental corpus 83.4% of the word types have one POS tag, 12.5% have 2, 3.1% have 3 and only 1% of the words have more. Therefore, $|A_i|$ is roughly constant. The ARR10 and Clark03 models assign a word type to a single cluster. For the other models, the number of clusters per word type is generally similar to that of the gold standard. Consequently, $|B_i|$ is roughly constant as well, which explains the similar behavior of the two measures.

Note that for other clustering tasks $|A_i|$ may not necessarily be constant, so the MacroI and MicroI scores are not likely to be as similar under the 1-1 mapping.

6 Summary

We discussed type level evaluation for polysemous clustering, presented new mapping-based evaluation measures, and applied them to the evaluation of POS induction algorithms, demonstrating that type level measures provide value beyond the common token level ones.

We hope that type level evaluation in general and the proposed measures in particular will be used in the future for evaluating clustering performance in NLP tasks.

References

Omri Abend, Roi Reichart and Ari Rappoport, 2010. Improved Unsupervised POS Induction through Prototype Discovery. *ACL '10.*

Thorsten Brants, 1997. The NEGRA Export Format. *CLAUS Report, Saarland University.*

Alexander Clark, 2003. Combining Distributional and Morphological Information for Part of Speech Induction. *EACL '03*.

Sajib Dasgupta and Vincent Ng, 2007. Unsupervised Part-of-Speech Acquisition for Resource-Scarce Languages. *EMNLP-CoNLL '07*.

Dmitry Davidov, Ari Rappoport, 2006. Efficient Unsupervised Discovery of Word Categories using Symmetric Patterns and High Frequency Words. *COLING-ACL '06*.

Dmitry Davidov, Ari Rappoport. 2008. Unsupervised Discovery of Generic Relationships Using Pattern Clusters and its Evaluation by Automatically Generated SAT Analogy Questions. *ACL '08*

I. S. Dhillon, S. Mallela, and D. S. Modha, 2003. Information Theoretic Co-clustering. *KDD '03*

Micha Elsner, Eugene Charniak, and Mark Johnson, 2009. Structured Generative Models for Unsupervised Named-Entity Clustering. *NAACL '09*.

Stella Frank, Sharon Goldwater, and Frank Keller, 2009. Evaluating Models of Syntactic Category Acquisition without Using a Gold Standard. *Proc. 31st Annual Conf. of the Cognitive Science Society,* 2576–2581.

E.B Fowlkes and C.L. Mallows, 1983. A Method for Comparing Two Hierarchical Clusterings. *Journal of American statistical Association,*78:553-569.

Benjamin C. M. Fung, Ke Wang, and Martin Ester, 2003. Hierarchical Document Clustering using Frequent Itemsets. *SIAM International Conference on Data Mining '03*.

Jianfeng Gao and Mark Johnson, 2008. *A Comparison of Bayesian Estimators for Unsupervised Hidden Markov Model POS Taggers. EMNLP '08.*

Sharon Goldwater and Tom Griffiths, 2007. Fully Bayesian Approach to Unsupervised Part-of-Speech Tagging. *ACL '07*.

João Graça, Kuzman Ganchev, Ben Taskar and Frenando Pereira, 2009. Posterior vs. Parameter Sparsity in Latent Variable Models. *NIPS '09*.

William P. Headden III, David McClosky and Eugene Charniak, 2008. *Evaluating Unsupervised Part-of-Speech Tagging for Grammar Induction.* COLING '08.

L. Hubert and J. Schultz, 1976. Quadratic Assignment as a General Data Analysis Strategy. *British Journal of Mathematical and Statistical Psychology*, 29:190-241.

L. Hubert and P. Arabie, 1985. Comparing Partitions. *Journal of Classification*, 2:193-218.

Maurice Kandall and Jean Dickinson, 1990. Rank Correlation Methods. *Oxford University Press, New York*.

Karin Kipper, Hoa Trang Dang and Martha Palmer, 2000. Class-Based Construction of a Verb Lexicon. AAAI '00.

Harold W. Kuhn, 1955. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly, 2:83-97.*

Bjornar Larsen and Chinatsu Aone, 1999. Fast and effective text mining using linear-time document clustering. *KDD '99*.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313-330.

Marina Meila and David Heckerman, 2001. An Experimental Comparison of Model-based Clustering Methods. *Machine Learning*, 42(1/2):9-29.

Marina Meila, 2007. Comparing Clustering – an Information Based Distance. *Journal of Multivariate Analysis*, 98:873-895.

C.W Milligan, S.C Soon and L.M Sokol, 1983. The Effect of Cluster Size, Dimensionality and the Number of Clusters on Recovery of True Cluster Structure. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 5:40-47.

Boris G. Mirkin, 1996. Mathematical Classification and Clustering. *Kluwer Academic Press.*

Michael Mitzenmacher , 2004. A Brief History of Generative Models for Power Law and Lognormal Distributions . *Internet Mathematics*, 1(2):226-251.

Soto Montalvo, Raquel Martnez, Arantza Casillas, and Vctor Fresno, 2006. Multilingual Document Clustering: an Heuristic Approach Based on Cognate Named Entities. *ACL '06*.

James Munkres, 1957. Algorithms for the Assignment and Transportation Problems. *Journal of the SIAM, 5(1):32-38.*

Cristina Nicolae and Gabriel Nicolae, 2006. BEST-CUT: A Graph Algorithm for Coreference Resolution. *EMNLP '06*.

Darius M. Pfitzner, Richard E. Leibbrandt and David M.W Powers, 2008. Characterization and Evaluation of Similarity Measures for Pairs of Clusterings. *Knowledge and Information Systems: An International Journal*, DOI 10.1007/s10115-008-0150-6.

William Rand, 1971. Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statstical Association*, 66(336):846-850.

Roi Reichart and Ari Rappoport, 2008. Unsupervised Induction of Labeled Parse Trees by Clustering with Syntactic Features. *COLING '08*.

Roi Reichart and Ari Rappoport, 2009. The NVI Clustering Evaluation Measure. *CoNLL '09*.

Andrew Rosenberg and Julia Hirschberg, 2007. V-Measure: A Conditional Entropy-based External Cluster Evaluation Measure. *EMNLP '07*.

Sabine Schulte im Walde, 2006. Experiments on the Automatic Induction of German Semantic Verb Classes. *Computational Linguistics*, 32(2):159-194.

Noah A. Smith and Jason Eisner, 2004. Annealing Techniques for Unsupervised Statistical Language Learning. *ACL '04*.

Stijn Van Dongen, 2000. Performance Criteria for Graph Clustering and Markov Cluster Experiments. *Technical report CWI, Amsterdam*

Jurgen Van Gael, Andreas Vlachos and Zoubin Ghahramani, 2009. The Infinite HMM for Unsupervised POS Tagging. *EMNLP '09*.

Yujing Zeng, Jianshan Tang, Javier Garcia-Frias, and Guang R. Gao, 2002. An Adaptive Meta-clustering Approach: Combining the Information from Different Clustering Results . *CSB* 00:276