Word Sense Disambiguation in Untagged Text based on Term Weight Learning

Fumiyo Fukumoto and Yoshimi Suzuki†
Department of Computer Science and Media Engineering,
Yamanashi University
4-3-11 Takeda, Kofu 400-8511 Japan
{fukumoto@skye.esb, ysuzuki@windermere.alps1.esi†}.yamanashi.ac.jp

Abstract

This paper describes unsupervised learning algorithm for disambiguating verbal word senses using term weight learning. In our method, *collocations* which characterise every sense are extracted using similarity-based estimation. For the results, term weight learning is performed. Parameters of term weighting are then estimated so as to maximise the collocations which characterise every sense and minimise the other collocations. The results of experiment demonstrate the effectiveness of the method.

1 Introduction

One of the major approaches to disambiguate word senses is supervised learning (Gale et al., 1992), (Yarowsky, 1992), (Bruce and Janyce, 1994), (Miller et al., 1994), (Niwa and Nitta, 1994), (Luk, 1995), (Ng and Lee, 1996), (Wilks and Stevenson, 1998). However, a major obstacle impedes the acquisition of lexical knowledge from corpora, i.e. the difficulties of manually sense-tagging a training corpus, since this limits the applicability of many approaches to domains where this hard to acquire knowledge is already available.

This paper describes unsupervised learning algorithm for disambiguating verbal word senses using term weight learning. In our approach, an overlapping clustering algorithm based on Mutual information-based (Mu) term weight learning between a verb and a noun is applied to a set of verbs. It is preferable that Mu is not low (Mu(x,y) $\geq$ 3) for a reliable statistical analysis (Church et al., 1991). However, this suffers from the problem of data sparseness, i.e. the co-occurrences which are used to represent every distinct senses does not appear in the test data. To attack this problem, for a low Mu value, we distinguish between unobserved co-occurrences that are likely to occur in a new corpus and those that are not, by using similarity-based estimation between two co-occurrences of words. For the results, term weight learning is performed. Parameters of term weighting are then estimated so as to maximise the collocations which characterise every sense and minimise the other collocations.

In the following sections, we first define a polysemy from the viewpoint of clustering, then describe how to extract collocations using similarity-based estimation. Next, we present a clustering method and a method for verbal word sense disambiguation using the result of clustering. Finally, we report on an experiment in order to show the effect of the method.

2 Polysemy in Context

Most previous corpus-based WSD algorithms are based on the fact that semantically similar words appear in a similar context. Semantically similar verbs, for example, co-occur with the same nouns. The following sentences from the *Wall Street Journal* show polysemous usages of <u>take</u>.

(s1) Coke has typically <u>taken</u> a minority <u>stake</u> in such ventures.
(s1') Guber and pepers tried to <u>buy</u> a <u>stake</u> in mgm in 1988.
(s2) That process of sorting out specifies is likely to <u>take</u> <u>time</u>.
(s2') We <u>spent</u> a lot of <u>time</u> and money in building our group of stations.

Let us consider a two-dimensional Euclidean space spanned by the two axes, each associated with <u>stake</u> and <u>time</u>, and in which <u>take</u> is assigned a vector whose value of the i-th dimension is the value of Mu between the verb and the noun assigned to the i-th axis. <u>Take</u> co-occurs with the two nouns, while <u>buy</u> and <u>spend</u> co-occur only with one of the two nouns. Therefore, the distances between <u>take</u> and these two verbs are large and the synonymy of take with them disappears.

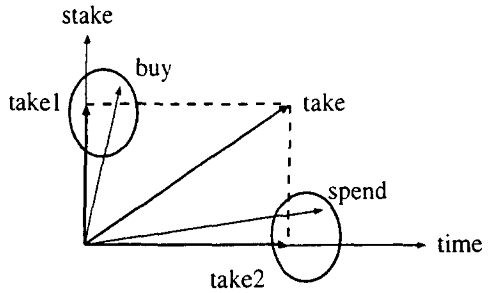

Figure 1: The decomposition of the verb take

In order to capture the synonymy of take with the two verbs correctly, one has to decompose the vector assigned to take into two component vectors, **take1** and **take2**, each of which corresponds to one of the two distinct usages of take (in Figure 1). (we call them *hypothetical verbs* in the following). The decomposition of a vector into a set of its component vectors requires a proper decomposition of the context in which the word occurs. Furthermore, in a general situation, a polysemous verb co-occurs with a large group of nouns and one has to divide the group of nouns into a set of subgroups, each of which correctly characterises the context for a specific sense of the polysemous word. Therefore, the algorithm has to be able to determine when the context of a word should be divided and how.

The approach proposed in this paper explicitly introduces new entities, i.e. *hypothetical verbs* when an entity is judged polysemous and associates them with contexts which are sub-contexts of the context of the original entity. Our algorithm has two basic operations, *splitting* and *lumping*. *Splitting* means to divide a polysemous verb into two *hypothetical verbs* and *lumping* means to combine two *hypothetical verbs* to make one verb out of them (Fukumoto and Tsujii, 1994).

3 Extraction of Collocations

Given a set of verbs, v_1, v_2, $\cdots$, v_m, the algorithm produces a set of semantic clusters, which are ordered in the ascending order of their semantic deviation values. Semantic deviation is a measure of the deviation of the set in an n-dimensional Euclidean space, where n is the number of nouns which co-occur with the verbs.

In our algorithm, if v_i is non-polysemous, it belongs to at least one of the resultant semantic clusters. If it is polysemous, the algorithm splits it into several hypothetical verbs and each of them belongs to at least one of the clusters. Table 1 summarises the sample result from the set {**close**, open, end}.

Table 1: Distinct senses of the verb '**close**'

v_i	n	Mu(v_i,n)
close1	account	2.116
(open)	banking	2.026
	acquisition	1.072
	book	4.427
	bottle	3.650
	...	...
close2	announcement	1.692
(end)	connection	2.745
	conversation	4.890
	period	1.876
	practice	2.564
	...	...

In Table 1, subsets 'open' and 'end' correspond to the distinct senses of '**close**'. Mu(v_i,n) is the value of mutual information between a verb and a noun. If a polysemous verb is followed by a noun which belongs to a set of the nouns, the meaning of the verb within the sentence can be determined accordingly, because a set of the nouns characterises one of the possible senses of the verb.

The basic assumption of our approach is that a polysemous verb could not be recognised correctly if collocations which represent every distinct senses of a polysemous verb were not weighted correctly. In particular, for a low Mu value, we have to distinguish between those unobserved co-occurrences that are likely to occur in a new corpus and those that are not. We extracted these collocations which represent every distinct senses of a polysemous verb using similarity-based estimation. Let (w_p, n_q) and (w'_{pi}, n_q) be two different co-occurrence pairs. We say that w_p and n_q are semantically related if w'_{pi} and n_q are semantically related and (w_p, n_q) and (w'_{pi}, n_q) are semantically similar (Dagan et al., 1993). Using the estimation, collocations are extracted and term weight learning is performed. Parameters of term weighting are then estimated so as to maximise the collocations which characterise every sense and minimise the other collocations.

Let v be two senses, w_p and w_1, but not be judged correctly. Let N_Set_1 be a set of nouns which co-occur with both v and w_p, but do not co-occur with w_1. Let also N_Set_2 be a set of nouns which co-occur with both v and w_1, but do not co-occur with w_p, and N_Set_3 be a set of nouns which co-occur with v, w_p and w_1. Extraction of collocations using similarity-based estimation

```
begin
(a)   for all n_q ∈ N_Set1 - N_Set3 such that Mu(w_p,n_q) < 3
      Extract w'_pi (1 ≤ i ≤ s) such that Mu(w'_pi, n_q) ≥ 3. Here, s is the number of verbs which
      co-occur with n_q
       for all w'_pi
         if w'_pi exists such that Sim(w_p,w'_pi) > 0
(a-1)      then parameters of Mu of (w_p,n_q) and (v,n_q) are set to α (1 < α)
(a-2)    else parameters of Mu of (w_p,n_q) and (v,n_q) are set to β (0 < β < 1)
        end_if
       end_for
      end_for
(b)   for all n_r ∈ N_Set3 such that Mu(w_p,n_r) ≥ 3 and Mu(w_1,n_r) ≥ 3
      Extract w'_pi (1 ≤ i ≤ t) such that Mu(w'_pi, n_r) ≥ 3. Here, t is the number of verbs which
      co-occur with n_r
       for all w'_pi
         if w'_pi exists such that Sim(w_p,w'_pi) > 0 and Sim(w_1,w'_pi) > 0
          then parameters of Mu of (v,n_r), (w_p,n_r) and (w_1,n_r) are set to β (0 < β < 1)
        end_if
       end_for
      end_for
end
```

Figure 2: Extraction of collocations

is shown in Figure 2 [1].
In Figure 2, (a-1) is the procedure to extract collocations which were not weighted correctly and (a-2) and (b) are the procedures to extract other words which were not weighted correctly. $Sim(v_i, v_i')$ in Figure 2 is the similarity value of v_i and v_i' which is measured by the inner product of their normalised vectors, and is shown in formula (1).

$$\begin{aligned} Sim(v_i, v_i') &= \frac{v_i \times v_i'}{\mid v_i \mid\mid v_i' \mid} \\ v_i &= (v_{i1}, \cdots, v_{ik}) \end{aligned} \tag{1}$$

$$v_{ij} = \begin{cases} \text{Mu}(v_i, n_j) & \text{if Mu}(v_i,n_j) \geq 3 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

In formula (1), k is the number of nouns which co-occur with v_i. v_{ij} is the Mu value between v_i and n_j.

We recall that w_p and n_q are semantically related if w_{pi}' and n_q are semantically related and (w_p,n_q) and (w_{pi}',n_q) are semantically similar. (a) in Figure 2, we represent w_{pi}' and n_q are semantically related when $\text{Mu}(w_{pi}',n_q) \geq 3$. Also, (w_p,n_q) and (w_{pi}',n_q) are semantically similar if $Sim(w_p, w_{pi}') > 0$. In (a) of Figure 2, for example, when (w_p,n_q) is judged to be a collocation which represents every distinct senses, we set Mu values of (w_p,n_q) and (v,n_q) to $\alpha \times \text{Mu}(w_p,n_q)$ and $\alpha \times \text{Mu}(v,n_q)$, $1 < \alpha$. On the other hand, when n_q is judged not to be a collocation which represents every distinct senses, we set Mu values of these co-occurrence pairs to $\beta \times \text{Mu}(w_p,n_q)$ and $\beta \times \text{Mu}(v,n_q)$, $0 < \beta < 1$ [2].

4 Clustering a Set of Verbs

Given a set of verbs, $\text{VG} = \{v_1, \cdots, v_m\}$, the algorithm produces a set of semantic clusters, which are sorted in ascending order of their semantic deviation. The deviation value of VG, $Dev(\text{VG})$ is shown in formula (3).

$$Dev(\text{VG}) = \frac{1}{\mid \bar{g} \mid (\beta * m + \gamma)} \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} (v_{ij} - \bar{g}_j)^2} \tag{3}$$

β and γ are obtained by *least square estimation* [3]. v_{ij} is the Mu value between v_i and n_j. $\bar{g}_j = \frac{1}{m} \sum_{i=1}^{m} v_{ij}$

[1] For w_1, we can replace w_p with w_1, $n_q \in N\_Set_1$ - N_Set_3 with $n_q \in N\_Set_1$ - N_Set_2, and $Sim(w_p, w_{pi}') > 0$ with $Sim(w_1, w_{pi}') > 0$.

[2] In the experiment, we set increment value of α and decrease value of β to 0.001.

[3] Using *Wall Street Journal*, we obtained $\beta = 0.964$ and $\gamma = -0.495$.

is the j-th value of the centre of gravity. $| \bar{g} | = \frac{1}{m}\sqrt{\sum_{j=1}^{n}(\sum_{i}^{m} v_{ij})^2}$ is the length of the centre of gravity. In formula (3), a set with a smaller value is considered *semantically less deviant.*

Figure 3 shows the flow of the clustering algorithm. As shown in '(' in Figure 3, the function **Make-Initial-Cluster-Set** applies to VG and produces all possible pairs of verbs with their semantic deviation values. The result is a list of pairs called the ICS (Initial Cluster Set). The CCS (Created Cluster Set) shows the clusters which have been created so far. The function **Make-Temporary-Cluster-Set** retrieves the clusters from the CCS which contain one of the verbs of Set_i. The results (Set_β) are passed to the function **Recognition-of-Polysemy**, which determines whether or not a verb is polysemous. Let v be an element included in both Set_i and Set_β. To determine whether v has two senses w_p, where w_p is an element of Set_i, and w_1, where w_1 is an element of Set_β, we make two clusters, as shown in (4) and their merged cluster, as shown in (5).

$$\{v_1, w_p\}, \{v_2, w_1, \cdots, w_n\} \quad (4)$$

$$\{v, w_1, \cdots, w_p, \cdots, w_n\} \quad (5)$$

Here, v and w_p are verbs and w_1, $\cdots$, w_n are verbs or hypothetical verbs. w_1, $\cdots$, w_p, $\cdots$, w_n in (5) satisfy $Dev(v, w_i) \leq Dev(v, w_j)$ $(1 \leq i \leq j \leq n)$. v_1 and v_2 in (4) are new hypothetical verbs which correspond to two distinct senses of v.

If v is a polysemy, but is not recognised correctly, then **Extraction-of-Collocations** shown in Figure 2 is applied. In **Extraction-of-Collocations**, for (4) and (5), α and β are estimated so as to satisfy (6) and (7).

$$Dev(v_1, w_p) \leq Dev(v, w_1, \cdots, w_p, \cdots, w_n) \quad (6)$$

$$Dev(v_2, w_1, \cdots, w_n) \leq Dev(v, w_1, \cdots, w_p, \cdots, w_n) \quad (7)$$

The whole process is repeated until the newly obtained cluster, Set_γ, contains all the verbs in the input or the ICS is exhausted.

5 Word Sense Disambiguation

We used the result of our clustering analysis, which consists of pairs of collocations of a distinct sense of a polysemous verb and a noun.

Let v has senses v_1, v_2, $\cdots$, v_m. The sense of a polysemous verb v is v_i $(1 \leq i \leq m)$ if $\sum_j^t Mu(v_i, n_j)$ is largest among $\sum_j^t Mu(v_1, n_j)$, $\cdots$ and $\sum_j^t Mu(v_m, n_j)$. Here, t is the number of nouns which co-occur with v within the five-word distance.

6 Experiment

This section describes an experiment conducted to evaluate the performance of our method.

6.1 Data

The data we have used is 1989 *Wall Street Journal* (WSJ) in ACL/DCI CD-ROM which consists of 2,878,688 occurrences of part-of-speech tagged words (Brill, 1992). The inflected forms of the same nouns and verbs are treated as single units. For example, 'book' and 'books' are treated as single units. We obtained 5,940,193 word pairs in a window size of 5 words, 2,743,974 different word pairs. From these, we selected collocations of a verb and a noun.

As a test data, we used 40 sets of verbs. We selected at most four senses for each verb, the best sense, from among the set of the Collins dictionary and thesaurus (McLeod, 1987), is determined by a human judge.

6.2 Results

The results of the experiment are shown in Table 2, Table 3 and Table 4.

In Table 2, 3 and 4, every polysemous verb has two, three and four senses, respectively. Column 1 in Table 2, 3 and 4 shows the test data. The verb v is a polysemous verb and the remains show these senses. For example, '**cause**' of (1) in Table 2 has two senses, 'effect' and 'produce'. 'Sentence' shows the number of sentences of occurrences of a polysemous verb, and column 4 shows their distributions. 'v' shows the number of polysemous verbs in the data. W in Table 2 shows the number of nouns which co-occur with w_p and w_1. $v \cap W$ shows the number of nouns which co-occur with both v and W. In a similar way, W in Table 3 and 4 shows the number of nouns which co-occur with $w_p \sim w_2$ and $w_p \sim w_3$, respectively. 'Correct' shows the performance of our method. 'Total' in the bottom of Table 4 shows the performance of 40 sets of verbs.

Table 2 shows when polysemous verbs have two senses, the percentage attained at 80.0%. When polysemous verbs have three and four senses, the percentage was 77.7% and 76.4%, respectively. This shows that there is no striking difference among them. Column 8 and 9 in Table 2, 3 and 4 show the results of collocations which were extracted by our method.

```
begin
  ICS := Make-Initial-Cluster-Set(VG)
  ( VG = {v_i | i = 1,···,m}    ICS = {Set_1,···,Set_{m(m-1)/2}}
    where Set_p = {v_i, v_j} and Set_q = {v_k, v_l} ∈ ICS (1 ≤ p < q ≤ m) satisfy Dev(v_i, v_j) ≤ Dev(v_k, v_l) )

  for i := 1 to m(m-1)/2 do
if CCS = φ
      then Set_γ := Set_i i.e. Set_i is stored in CCS as a newly obtained cluster
else if Set_α ∈ CCS exists such that Set_i ⊂ Set_α
      then Set_i is removed from ICS and Set_γ := φ
else if
      for all Set_α ∈ CCS do
         if Set_i ∩ Set_α = φ
            then Set_γ := Set_i i.e. Set_i is stored in CCS as a newly obtained cluster
         end_if
      end_for
else Set_β := Make-Temporary-Cluster-Set(Set_i,CCS)
            ( Set_β := Set_α ∈ CCS such that Set_i ∩ Set_α ≠ φ

      Set_γ := Recognition-of-Polysemy(Set_i,Set_β)
      if Set_γ was not recognised correctly
         then for v, w_p and w_1, do
                 Extraction-of-Collocations.
               end_for
         i := 1
      end_if
end_if
end_if
end_if
      if Set_γ = VG
         then exit from the for_loop ;
      end_if
   end_for
end
```

Figure 3: Flow of the algorithm

Mu < 3 shows the number of nouns which satisfy Mu(w_p,n) < 3 or Mu(w_1,n) <3. 'Correct' shows the total number of collocations which could be estimated correctly. Table 2 $\sim$ 4 show that the frequency of v is proportional to that of $v \cap W$. As a result, the larger the number of $v \cap W$ is, the higher the percentage of correctness of collocations is.

7 Related Work

Unsupervised learning approaches, i.e. to determine the class membership of each object to be classified in a sample without using sense-tagged training examples of correct classifications, is considered to have an advantage over supervised learning algorithms, as it does not require costly hand-tagged training data.

Schütze and Zernik's methods avoid tagging each occurrence in the training corpus. Their methods associate each sense of a polysemous word with a set of its co-occurring words (Schutze, 1992), (Zernik, 1991). If a word has several senses, then the word is associated with several different sets of co-occurring words, each of which corresponds to one of the senses of the word. The weakness of Schütze and Zernik's method, however, is that it solely relies on human intuition for identifying different senses of a word, i.e. the human editor has to determine, by her/his intuition, how many senses a word has, and then identify the sets of co-occurring words that correspond to the different senses.

Table 2: The result of disambiguation experiment(two senses)

Num	$\{v, w_p, w_1\}$	Sentence	w_p(%) w_1(%)	v	$v \cap W$	Correct(%)	Mu < 3	Correct(%)
(1)	{cause, effect, produce}	232	89(38.3) 143(61.7)	464	254	158(68.1)	118	85(72.0)
(2)	{claim, require affirm}	202	158(78.2) 44(21.8)	545	328	160(79.2)	89	71(80.0)
(3)	{close, open, end}	206	92(44.6) 114(55.4)	2,025	1,498	162(78.6)	136	120(88.2)
(4)	{fall, decline, win}	278	182(65.5) 96(34.5)	697	393	215(77.3)	395	325(82.5)
(5)	{feel, think, sense}	280	178(63.5) 102(36.5)	210	53	156(55.7)	39	20(53.1)
(6)	{hit, attack, strike}	250	156(62.4) 94(37.6)	452	122	181(72.4)	83	63(77.0)
(7)	{leave, remain, go}	183	41(22.4) 142(77.6)	1,409	942	160(87.4)	132	113(86.2)
(8)	{lose,win, get}	301	70(23.3) 231(76.7)	2,244	1,920	242(80.3)	329	282(86.0)
(9)	{manage, accomplish, operate}	216	23(10.6) 193(89.4)	648	306	165(76.3)	89	71(80.6)
(10)	{occur, happen, exist}	199	137(68.9) 62(31.1)	324	180	175(87.9)	136	95(70.3)
(11)	{order, request, arrange}	240	206(85.8) 34(14.2)	1,256	831	202(84.1)	138	121(87.7)
(12)	{pass, adopt, succeed}	301	175(58.1) 126(41.9)	378	247	259(86.0)	52	41(80.5)
(13)	{post, mail, inform}	274	63(22.9) 211(77.1)	91	68	231(84.3)	44	32(72.9)
(14)	{produce, create, grow}	231	129(55.8) 102(44.2)	640	370	184(79.6)	296	258(87.1)
(15)	{push, attack, pull}	223	93(41.8) 130(58.2)	202	153	189(84.7)	126	97(77.7)
(16)	{save, keep, rescue}	216	149(68.9) 67(31.1)	396	354	168(77.7)	115	93(81.6)
(17)	{ship, put, send}	218	92(42.2) 126(57.8)	361	241	176(80.7)	92	74(80.4)
(18)	{stop, end, move}	244	52(21.4) 192(78.6)	993	886	210(86.0)	193	169(87.5)
(19)	{add, append, total}	184	35(19.0) 149(81.0)	1,164	778	147(79.8)	164	129(78.6)
(20)	{keep, maintain, protect}	378	231(61.1) 147(38.9)	1,266	905	349(92.3)	267	234(87.7)
Total (2 senses)		4,856	3,332(68.6)			3,889(80.0)		

Yarowsky used an unsupervised learning procedure to perform noun WSD (Yarowsky, 1995). This algorithm requires a small number of training examples to serve as a seed. The result shows that the average percentage attained was 96.1% for 12 nouns when the training data was a 460 million word corpus, although Yarowsky uses only nouns and does not discuss distinguishing more than two senses of a word.

A more recent unsupervised approach is described in (Pedersen and Bruce, 1997). They presented three unsupervised learning algorithms that distinguish the sense of an ambiguous word in untagged text, i.e. McQuitty's similarity analysis, Ward's minimum-variance method and the EM algorithm. These algorithms assign each instance of an ambiguous word to a known sense definition based solely on the values of automatically identifiable features in text. Their methods are perhaps the most similar to our present work. They reported that disambiguating nouns is more successful rather than adjectives or verbs and the best result of verbs was McQuitty's method (71.8%), although they only tested 13 ambiguous words (of these, there are only 4 verbs). Furthermore, each has at most three senses. In future, we will compare our method with their methods using the data we used in our experiment.

8 Conclusion

In this study, we proposed a method for disambiguating verbal word senses using term weight learning based on similarity-based estimation. The results showed that when polysemous verbs have two, three and four senses, the average percentage attained at 80.0%, 77.7% and 76.4%, respectively. Our method assumes that nouns which co-occur with a polysemous verb is disambiguated in advance. In future, we will extend our method to cope with this problem and also apply our

Table 3: The result of disambiguation experiment(three senses)

Num	$\{v, w_p, w_1\ w_2\}$	Sentence	w_p(%) w_1(%) w_2(%)	v	$v \cap W$	Correct(%)	Mu < 3	Correct(%)
(21)	{catch, acquire, grab, watch}	240	120(50.0) 21(9.0) 199(41.0)	447	432	180(75.0)	124	99(79.9)
(22)	{complete, end, develop, fill}	365	107(29.3) 242(66.3) 16(4.4)	727	450	280(76.7)	240	193(80.4)
(23)	{gain, win, get, increase}	334	47(14.0) 228(68.2) 59(17.8)	527	467	270(80.8)	187	152(81.4)
(24)	{grow, increase, develop become}	310	68(21.9) 132(42.5) 110(35.6)	903	651	241(77.7)	372	305(82.0)
(25)	{operate, run, act, control}	232	76(32.7) 83(35.7) 73(31.6)	812	651	187(80.6)	311	255(82.3)
(26)	{rise, increase, appear, grow}	276	51(18.4) 137(49.6) 88(32.0)	711	414	198(71.7)	372	294(79.1)
(27)	{see, look, know, feel}	318	128(40.2) 162(50.9) 28(8.9)	1,785	934	263(82.7)	497	414(83.4)
(28)	{want, desire, search, lack}	267	66(24.7) 53(19.8) 148(55.5)	590	470	208(77.9)	198	159(80.8)
(29)	{lead, cause, guide, precede}	183	139(75.9) 38(20.7) 6(3.4)	548	456	138(75.4)	274	221(80.9)
(30)	{carry, bring, capture, behave}	186	142(76.3) 39(20.9) 5(2.8)	474	440	142(76.3)	207	167(80.7)
Total (3 senses)		2,711	1,573(56.5)			2,107(77.7)		

method to not only a verb but also a noun and an adjective sense disambiguation to evaluate our method.

Acknowledgments

The authors would like to thank the reviewers for their valuable comments. This work was supported by the Grant-in-aid for the Japan Society for the Promotion of Science(JSPS).

References

E. Brill. 1992. A simple rule-based part of speech tagger. In *Proc. of the 3rd Conference on Applied Natural Language Processing*, pages 152–155.

R. Bruce and W. Janyce. 1994. Word-sense disambiguation using decomposable models. In *Proc. of the 32nd Annual Meeting*, pages 139–145.

K. W. Church, W. Gale, P. Hanks, and D. Hindle. 1991. Using statistics in lexical analysis. In *Lexical acquisition: Exploiting on-line resources to build a lexicon*, pages 115–164. (Zernik Uri (ed.)), London, Lawrence Erlbaum Associates.

I. Dagan, P. Fernando, and L. Lilian. 1993. Contextual word similarity and estimation from sparse data. In *Proc. of the 31th Annual Meeting of the ACL*, pages 164–171.

F. Fukumoto and J. Tsujii. 1994. Automatic recognition of verbal polysemy. In *Proc. of the 15th COLING, Kyoto, Japan*, pages 762–768.

W. K. Gale, K. W. Church, and D. Yarowsky. 1992. A method for disambiguating word senses in a large corpus. In *Computers and the Humanities*, volume 26, pages 415–439.

A. K. Luk. 1995. Statistical sense disambiguation with relatively small corpora using dictionary definitions. In *Proc. of the 33st Annual Meeting of ACL*, pages 181–188.

W. T. McLeod. 1987. The new collins dictionary and thesaurus in one volume. London, HarperCollins Publishers.

G. Miller, C. Martin, L. Shari, L. Claudia, and R. G. Thomas. 1994. Using a semantic concordance for sense identification. In *Proc. of the ARPA Workshop on Human Language Technology*, pages 240–243.

H. T. Ng and H. B. Lee. 1996. Integrating multiple knowledge sources to disambiguate word

Table 4: The result of disambiguation experiment(four senses)

Num	$\{v, w_p, w_1, w_2, w_3\}$	Sentence	w_p(%) w_1(%) w_2(%) w_3(%)	v	$v \cap W$	Correct(%)	Mu < 3	Correct(%)
(31)	{develop, create, grow, improve, expand}	187	117(62.5) 34(18.1) 4(2.1) 32(17.3)	922	597	155(82.8)	253	218(86.1)
(32)	{face, confront, cover, lie, turn}	222	54(24.3) 103(46.3) 12(5.4) 53(24.0)	859	567	184(82.8)	178	154(86.5)
(33)	{get, become, lose, understand, catch}	302	88(29.1) 98(32.4) 34(11.2) 82(27.3)	762	513	229(75.8)	424	365(86.2)
(34)	{go, come, become, run, fit}	217	101(46.5) 66(30.4) 36(16.5) 14(6.6)	732	435	145(66.8)	374	302(80.9)
(35)	{make, create, do, get, behave}	227	123(54.1) 28(12.3) 58(25.5) 18(8.1)	783	555	178(78.4)	435	370(85.2)
(36)	{show, appear, inform, prove, express}	227	121(53.3) 16(7.0) 40(17.6) 50(22.1)	996	560	181(79.7)	258	214(83.2)
(37)	{take, buy, obtain, spend, bring}	246	20(8.1) 123(50.0) 42(17.0) 61(24.9)	2,742	1,244	179(72.7)	829	677(81.6)
(38)	{hold, keep, carry, reserve, accept}	145	7(4.8) 53(36.5) 2(1.5) 83(57.2)	727	459	111(76.5)	394	300(76.2)
(39)	{raise, lift, increase, create, collect}	204	2(1.1) 81(39.7) 86(42.1) 35(17.1)	746	491	151(74.0)	341	272(79.7)
(40)	{draw, attract, pull, close, write}	162	78(48.1) 13(8.0) 43(26.5) 28(17.4)	798	533	123(75.9)	143	119(83.2)
Total (4 senses)		2,139				1,636(76.4)		
Total		9,706				7,572(78.6)		

sense: An examplar-based approach. In *Proc. of the 34th Annual Meeting of ACL*, pages 40–47.

Y. Niwa and Y. Nitta. 1994. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *Proc. of 15th COLING, Kyoto, Japan*, pages 304–309.

T. Pedersen and R. Bruce. 1997. Distinguishing word senses in untagged text. In *Proc. of the 2nd Conference on Empirical Methods in Natural Language Processing*, pages 197–207.

H. Schutze. 1992. Dimensions of meaning. In *Proc. of Supercomputing*, pages 787–796.

Y. Wilks and M. Stevenson. 1998. Word sense disambiguation using optimised combinations of knowledge sources. In *Proc. of the COLING-ACL'98*, pages 1398–1402.

D. Yarowsky. 1992. Word sense disambiguation using statistical models of roget's categories trained on large corpora. In *Proc. of the 14th COLING*, pages 454–460.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proc. of the 33rd Annual Meeting of the ACL*, pages 189–196.

U. Zernik. 1991. Train1 vs. train2: Tagging word senses in corpus. In *Lexical acquisition: Exploiting on-line resources to build a lexicon*, pages 91–112. Uri Zernik(Ed.), London, Lawrence Erlbaum Associates.