Procesamiento del Lenguaje Natural, núm. 35 (2005), pp. 171-178 recibido 26-04-2005; aceptado 01-06-2005

Exploiting Rules for Word Sense Disambiguation in Machine Translation

Lucia Specia, Maria das Graças V. Nunes
ICMC - University of São Paulo
Av. do Trabalhador São-Carlense, 400
São Carlos, 13560-970, Brazil
{lspecia,gracan}@icmc.usp.br

Mark Stevenson
University of Sheffield
Regent Court, 211 Portobello Street
Sheffield, S1 4DP, UK
M.Stevenson@dcs.shef.ac.uk

Resumen: Esta comunicación describe la generación automática y la evaluación de conjuntos de reglas para la desambiguación del sentido de las palabras (DSP) en la traducción automática. El objetivo principal es identificar reglas de alta calidad que se pueden utilizar como fuentes del conocimiento en un modelo relacional de DSP. La evaluación fue realizada automáticamente, por medio de cuatro medidas objetivas (error, cobertura, suporte y novedad), y manualmente, por medio de un análisis subjetivo del nivel del interés de las mejores reglas según lo precisado por las medidas objetivas. Han sido seleccionadas 63 reglas que trataban siete verbos altamente ambiguos. La evaluación también evidenció qué clases de conocimiento fueron utilizadas con eficacia por las reglas de DSP, que no son siempre iguales aquéllas reveladas por evaluaciones tradicionales de los modelos completos de DSP. Aunque hayamos realizado experimentos con inglés-portugués, los procedimientos de generación y evaluación de las reglas se podrían aplicar a cualquier par de lenguas, siempre que un corpus de ejemplos de desambiguación esté disponible para este par de lenguas.
Palabras clave: Desambiguación del sentido de las palabras, aprendizaje de máquina, evaluación de reglas

Abstract: This paper describes the automatic generation and the evaluation of sets of rules for word sense disambiguation (WSD) in machine translation. The ultimate aim is to identify high-quality rules that can be used as knowledge sources in a relational WSD model. The evaluation was carried out both automatically, by means of four objective measures (error, coverage, support and novelty), and manually, by means of a subjective analysis of the level of interest of the best rules as pointed out by the objective measures. As a result, we selected 63 rules addressing seven highly ambiguous verbs. The evaluation also evidenced which kinds of knowledge were effectively used by the WSD rules, which are not always the same as those revealed by traditional evaluations of complete WSD models. Although we experimented with English-Portuguese, the rule generation and evaluation procedures could be applied to any language pair, provided that there is a disambiguation sample corpus for that language pair.
Keywords: Word sense disambiguation, machine learning, rule evaluation

1 Introduction

Word Sense Disambiguation (WSD) in Machine Translation (MT) is required to carry out the lexical choice in the case of semantic ambiguity during the translation, i.e., the choice for the most appropriate translation for a source language word when the target language offers more than one option, with different meanings, but the same part-of-speech. For example, assuming English-Portuguese translation, the noun *bank* can be translated as *banco* (*financial institution*) or *margem* (*land along the side of a river*), and the verb *to run* can be translated as *correr* (*to move quickly*) and *ir* (*to go*). So, in this context, "sense" means, in fact, "translation".

Different WSD paradigms have been proposed for MT, including **knowledge-based** approaches, which depend on the manual encoding of accurate linguistic knowledge and disambiguation rules, e.g., (Dorr and Katsova, 1998), **corpus-based** approaches, which make use of knowledge automatically acquired from text using machine learning techniques, e.g., (Lee, 2002), and **hybrid** approaches, which mix characteristics of the other two approaches, e.g., (Zinovjeva, 2000). Recent

ISSN: 1135-5948

works have converged to the use of corpus-based or hybrid techniques, which have shown good results, in terms of accuracy and coverage, especially those following the supervised learning. In this work we are focusing on hybrid approaches, which minimize the knowledge acquisition bottleneck, but also concern about the accuracy of the acquired knowledge. The language pair under consideration is English-Portuguese, not addressed by any other work. As stressed in Specia (2005a), the lack of effective WSD mechanisms is one of the main reasons for the unsatisfactory results of the existent English-Portuguese MT systems.

One key issue in corpus-based and hybrid WSD is the knowledge sources (KSs) used in the machine learning process. Several studies have been carried out to discover the best KSs, e.g. (Zinovjeva, 2000), (Stevenson and Wilks, 2001), (Agirre and Martínez, 2001), (Lee and Ng, 2002), and (Mohammad and Pedersen, 2004). These have explored several KSs, including part-of-speech tags, collocations, syntagmatic relations, topical associations, selectional preferences, frequency of senses, etc. As a result, they have shown that different KSs convey models with different accuracies and, in general, that combinations of KSs are more effective than individual sources. A few sources have been agreed to be very important by most of the works (especially collocations). However, a common conclusion is that the accuracy of the approach is also strongly influenced by many other factors, such as the algorithm being used and the words being disambiguated.

In order to identify the best KSs, all the previously mentioned studies consider the precision of the produced model, or, in some cases, also its coverage. With exception of the Zinovjeva's work, which analyzes the WSD rules produced for three words in a MT context, all the others consider monolingual WSD and the analysis of the complete model, but not of the individual rules. In fact, most of the works evaluated algorithms of paradigms other than symbolic, and so this analysis would not be possible.

Current approaches to WSD, even in monolingual contexts, use propositional formalisms to represent knowledge and examples, that is, the attribute-value format. This formalism makes unfeasible the representation of substantial knowledge, mainly if it is relational (e.g., distance, syntactic, and semantic relations among words), and its use during the learning process (Mooney, 1997). Relational knowledge is especially important for WSD, given that it is necessary to analyze different aspects about the context of the ambiguous word.

The work we present in this paper is part of a major research project, which aims at the creation of a new hybrid symbolic approach to WSD, to be applied to English-Portuguese MT (Specia, 2005a). The main innovative feature of this approach is the relational formalism to be used to represent instances and background knowledge. Before developing such approach, we first experimented with some propositional machine learning algorithms, in order to gain some insights to the proposed work, namely: (1) find out the accuracies of those algorithms, considering several KSs, to compare them to the ones to be obtained by the proposed approach; (2) identify the best KSs and filters for the proposed approach; and (3) extract rules from the predicted models that may be used as KS in the proposed approach.

The first two goals were already addressed through a set of experiments with seven highly ambiguous verbs (*to come*, *to get*, *to give*, *to go*, *to look*, *to make* and *to take*), four propositional algorithms and features representing syntactic, semantic and topical knowledge, either individually or in combinations of two or three (Specia, 2005b). In this paper we focus on the third goal. Our hypothesis is that, since the proposed approach will allow the use of explicit knowledge about disambiguation, along with the disambiguation instances, a good source of knowledge may be provided by other kind of empirical data: automatically acquired disambiguation rules. This strategy could be thought as an iterative learning. Although some works have adopted iterative learning to monolingual WSD, they are all constrained to propositional environments. Consequently, the mentioned hypothesis has not been explored in WSD so far. Furthermore, the main idea here is not to bootstrap from propositional to relational approaches, but to gather significant knowledge evidences that could contribute to the relational approach.

In order to produce and evaluate the rules, we experimented with the same set of verbs and features, and the decision tree algorithm C4.5, considering each branch of the tree as a rule. In contrast to other works, we analyzed the rules individually and explored other measures in addition to accuracy, namely, coverage, support and novelty, employing specific criteria for each measure. We also analyzed the rules manually, assessing subjective criteria revealing the level of interest of the rules, namely, the usefulness and unexpectedness of those rules. During the manual analysis, some rules were also improved in a few aspects. As a consequence of selecting the best rules according to those measures and criteria, the experiments also evidenced the kinds of knowledge effectively used in the produces models. This result can be thought as a farther investigation of our second goal.

In the rest of this paper we first describe, in Section 2, our experimental setting, including the KSs used as features, our sample corpus, the objective and subjective measures, as well as the algorithm and data mining environment employed. In Section 3 we present the evaluation experiment and discuss its results. In Section 4 we conclude with some remarks and future work.

2 *Experimental setting*

2.1 Knowledge sources, features and lexical resources

According to the taxonomy of **knowledge sources**, **features**, and **lexical resources** defined by Agirre and Stevenson (2005), we explore knowledge from three sources: (1) syntactic: different collocations and their part-of-speech (POS); (2) semantic: subject and object syntactic dependencies with relation to the verb; and (3) pragmatic/topical: topical word associations.

In order to select a subset of possible feature combinations to encode these KSs, we used the results of the experiments previously carried out aiming to find out the accuracies of the algorithms and the best KSs and filters. We also used, in those experiments, one instance filter commonly employed in WSD (Lee and Ng, 2002) to tackle the feature sparseness problem: we remove from all instances the features values that occur less than a given N number of times with a certain sense. We choose the best feature settings (Table 1) and filters (N=1 – i.e., no filter –, and N=3), as pointed out by those experiments.

All features were encoded as multi-valued features, having as possible values the lemmas/POS in the sentence position that they represent. Regarding the lexical resources, all the features were extracted from a corpus (Section 2.2) previously annotated with the senses and also all the needed information.

2.2 Sample data

Our sample corpus consists of English sentences containing the seven verbs under consideration: *to come*, *to get*, *to give*, *to go*, *to look*, *to make* and *to take*. The sentences were collected from the Compara corpus (Frankenberg-Garcia and Santos, 2003), which comprises texts of fiction books. Each sentence has a sense tag, which corresponds to the translation of the verb in that sentence. The sense tagging process was carried out automatically, as described in Specia et al. (2005), and then manually reviewed. Besides the sense tags, the corpus presents: (1) POS tags of all words; (2) lemmas of all words; and (3) subject-object syntactic relations.

No.	Setting
S1	Bag-of-words and POS of ± 5 lemmas of words surrounding the verb.
S2	Bag-of-words and POS of ± 5 lemmas of words surrounding the verb, and subject and object relations.
S3	Lemmas of the first and second words to left and right, first noun, first adjective, and first verb to left and right of the verb, and first preposition to the right of the verb.
S4	Lemmas of the first and second words to left and right, first noun, first adjective, and first verb to left and right of the verb, and first preposition to the right of the verb, and subject and object relations.
S5	Lemmas and POS of content words in a ± 5 word window, and subject and object relations.

Table 1: Features tested in the experiments

To minimize the sparseness of our original set of instances (1,400) with respect to the classes (senses), i.e., the number of senses with only one sentence as instance, we filtered the data selecting only the instances for which the sense occurred at least three times. The initial number of senses and the number of remaining instances and senses after the filter are shown in Table 2, along with the resultant percentage of instances with the most frequent sense.

Verb	Initial senses	remaining instances	remaining senses	most frequent sense
come	26	183	11	50.3%
get	51	157	17	21.0%
give	27	180	5	88.8%
go	25	197	11	68.5%
look	16	191	7	50.3%
make	39	170	11	70.0%
take	63	142	13	28.5%

Table 2: Sample data after the instance filter

A feature extractor was developed to extract the features values from the sample corpus and represent them and their headers in the attribute file format of the data mining environment used to run the experiments.

2.3 Algorithm, data mining environment and rule generation

To produce the rules we chose the C4.5 algorithm (Quinlan, 1988), using the original implementation provided by the Sniffer system (Batista and Monard, 2004), as part of the Discover data mining environment (Prati et al., 2003). This environment offers integrated tools to produce and evaluate models according to different algorithms, evaluation measures, and other data mining characteristics. Besides being one of the most commonly used symbolic algorithms, C4.5 was chosen because, differently from other machine learning algorithms, it makes a clear distinction among known and unknown data, that is, among features that have values for all the instances and features with undefined values. This distinction is important in this work: we intend to evaluate only rules based on known data, since these rules explicitly indicate the KSs being used.

We first ran C4.5 with its default parameters for our 70 different instance sets (five different feature settings for each of the seven verbs, with both filters), which resulted in pruned trees as output. Then we ran other 70 experiments with the same instance sets, but taking the unpruned trees, that is, increasing the confidence factor parameter at the most possible. We evaluated both versions of the trees because for some verbs the information gain criterion employed by C4.5 was too strict to our purposes, resulting in pruned trees having only the default branch (voting for the majority class).

In both cases (pruned and unpruned), we used the same data set for training and testing, given that we have a small number of instances, with a high level of variability. It is worth noticing that we could not use an *n*-fold cross validation strategy here, since it would produce *n* separate models, which would be infeasible to analyze.

```
R18     IF col_1 = up
THEN CLASS = levantar [0.0649, 0.0065,
0.9026, 0.0260, 154] ?[0.0000, 1.0000, 0.0000,
0.0000, 3]
```

Figure 1: Example of rule produced by C4.5 through the Sniffer system

After running C4.5 for a certain set of training and test instances, the Sniffer system converts the produced rules into the Discover syntax, which is used in further steps of the rule evaluation process. One example of such output is shown in Figure 1, for the verb *to get*, considering collocations (S3) as features.

The first line represents the condition: *col_1 (first word to the right) = up*; while the second represents the class assigned in case the conditions are met, *levantar*. The *if-then* rules can be generalized by $B \rightarrow H$, where B stands for the body or antecedent of the rule, i.e., the set of conditions on one or more features, and H stands for the head or consequent of the rule, i.e., the target feature (or class). The second line also represents the relative frequency based contingency tables for instances for which the values of the features under consideration are known, and for instances for which those values are unknown (after "?"). Both contingency tables are compacted representations of the data in Table 3, as shown in what follows.

	H	$\overline{H}$	
B	fbh	$fb\overline{h}$	fb
$\overline{B}$	$f\overline{b}h$	$f\overline{bh}$	$f\overline{b}$
	fh	$f\overline{h}$	1

[fbh, $fb\overline{h}$, $f\overline{b}h$, $f\overline{bh}$, n]

Table 3: Relative frequency contingency table

In both representations, $fx = x/n$, where n is the number of instances used to generate the rule. For example, fbh stands for the frequency of instances for which the body and the head are true, that is, for which the conditions are satisfied and the class is correct. On the other hand, $fb\overline{h}$ stands for the frequency of instances for which the body is true but the head is false.

Given a set of rules represented as the example in Figure 1, the individual rules were evaluated using the Rulee system (Paula, 2003), also part of the Discover environment. Based on the contingency table for known data, this system allows the use of more than 25 objective measures through a friendly interface to consult the rule set according to one or more measures and to user defined criteria (e.g., selecting rules with a minimum value for a certain measure).

2.4 Measures

Among the objective measures provided by Rulee, we selected those we consider most pertinent to the task of WSD: error, coverage, support and novelty. The first three are commonly used to evaluate complete WSD models. The fourth can be thought as a quantitative estimative of the level of interest of the rules. In what follows, we describe such measures, based on the terminology used by Lavrac et al. (1999).

$$Error = P(\overline{H} \mid B) = fb\overline{h} / fb$$

This measure corresponds to the negative version of to the traditionally used *precision* (*1-precision*), i.e., the level of confidence of the rule. Since we are considering the evaluation of the rules and not of the complete models, it is very usual to have rules with very close precisions. In this sense, the negative version provides a clearer evidence of the difference between rules (precisions of 0.99 and 0.98 sounds more similar than the equivalent errors of 0.01 and 0.02 – the second is twice as the first).

$$Support = P(BH) = fbh$$

Support, also referred to as *recall*, measures the percentage of instances correctly classified by the rule. A high support indicates both high coverage and precision.

$$Coverage = P(B) = fb$$

The coverage indicates the number of instances (correctly or incorrectly) addressed by that rule. The reason for using this measure here, in addition to support, is that rules with a high coverage, but not necessarily a high support, could be manually improved, leading to new wide coverage and accurate rules.

$$Novelty = P(BH) - P(H)P(B) = fbh - fhfb$$

Novelty quantifies the correlation between B and H. It varies from –0.25 to 0.25. If $Novelty = 0$, H and B are independents and the rule does not present any novelty. The higher the value of novelty, the higher the correlation between B and H. The smaller the value of novelty, the higher the correlation between B and $\overline{H}$. Thus, absolute values other than zero indicate a rule that brings some new or interesting information, from a quantitative point of view.

In order to qualitatively assess the level of interest of the rules, we considered two aspects, as proposed by Silberschatz and Tuzhilin (1996): the actionability (i.e., usefulness) and the unexpectedness (i.e., unpredictability) of the rule. An unexpected rule presents a pattern that was contrary to the expectation of the user, while a useful rule presents a pattern that can be helpful to the user. In both cases, the rule is interesting. Although there have been some attempts to model and quantify these measures, they still rely on completely subjective criteria. Here we manually look into the rules to judge their level of interest.

3 *Evaluation criteria, experiment and results*

We divided the evaluation in two steps: first we applied the objective measures with certain criteria to automatically reduce the number of rules, selecting those relevant according to those criteria. Then we manually analyzed the resultant reduced set of rules, selecting those considered interesting to be used as KS in our WSD system. Although the first automatic filter can cause the exclusion of interesting rules, it was necessary since the number of rules was too high.

3.1 Objective measures

Before using the Rulee system, we removed all the rules generated by estimating values for the unknown features (those for which the first contingency table did not have values for fbh). We then entered the 140 Sniffer output files in the Rulee system and consulted each set of rules according to our four objective measures and the following criteria:

- Error < error of the majority class.
- abs(Novelty) ≥ 0.01.
- Coverage ≥ 0.1.
- Support ≥ 0.05.

The criterion for the error measure is commonly used: the error must be lower than the error that would be achieved by a default rule voting always by the majority class, without analyzing any feature. As for the other criteria, in order to establish their thresholds, we experimented with several values, trying to find out appropriate distinctive criteria that would lead to a number of rules feasible to be manually analyzed (for many data sets, the original number of rules was around 100). For example, choosing the novelty threshold as $abs(Novelty) \neq 0$ seems to be the most intuitive option, but it would make all except the default rules to be selected.

To select the rules, error was considered a strict criterion: rules that did not meet it were not selected. The other criteria were less strict: a rule was selected if it satisfied at least two criteria. The number of resultant selected rules for each verb, feature setting, and filter is shown in Table 4 (pruned and unpruned, respectively).

Most of the cases with zero selected rules refer to the existence of a default rule in the original set of rules, that is, a rule voting for the majority class and thus presenting the corresponding majority class error (this is usual for verbs with a highly most frequent translation). These cases are marked by “*” in Table 4. For some verbs and features, even the unpruned tree did not generate better rules than the default one, especially when no filter was used (N=1). The use of N=3 caused a higher number of rules to be generated and selected. Some feature settings with common features, such as S3 and S4, resulted in the same rules for certain verbs. There are also repeated rules derived from the unpruned and pruned versions of the trees. So, the total number of rules (349) does not imply different rules.

	pruned						
Verb **Feature**	**come**	**get**	**give**	**go**	**look**	**make**	**take**
S1, N=1	0*	2	0*	0*	3	0*	1
S2, N=1	0*	2	0*	0*	3	0*	1
S3, N=1	0*	2	0*	0*	3	0*	1
S4, N=1	0*	2	0*	0*	3	0*	1
S5, N=1	0*	0*	0*	0*	0*	0*	0
S1, N=3	5	5	0*	4	3	0*	5
S2, N=3	7	5	0*	4	3	0*	5
S3, N=3	7	6	0*	4	4	0*	9
S4, N=3	7	6	0*	4	4	0*	9
S5, N=3	0*	0*	0*	0*	0*	0*	0*

	unpruned						
Verb **Feature**	**come**	**get**	**give**	**go**	**look**	**make**	**take**
S1, N=1	4	0	0	0	1	0	0
S2, N=1	1	0	0	0	1	0	0
S3, N=1	3	1	0	1	3	0	0
S4, N=1	2	1	0	1	1	0	0
S5, N=1	0	0	2	2	1	0	0
S1, N=3	6	9	2	5	3	7	8
S2, N=3	7	9	2	4	3	7	8
S3, N=3	6	6	3	4	10	3	10
S4, N=3	7	10	3	6	10	3	10
S5, N=3	0*	2	0*	5	9	0*	7

Table 4: Number of rules resultant from the objective measures filter

3.2 Subjective measures

In this step we manually analyzed the 349 rules selected in the first step, looking for interesting rules considering the two mentioned aspects: unexpectedness and usefulness. We intended to remove from the set of rules previously selected those not meeting any of those aspects. One example of removed rule is given in Figure 2, for the verb *to come*. The rule states that if the first word to the right (brw_1) of *come* is *to*, *come* must be translated as *vir*. It does not represent a useful rule, since *come to* is also a phrasal verb with many translations other than *vir*, and we assume that if the verb can be used as phrasal verb in the sentence, the corresponding phrasal verb translations must be preferred to the individual verb translations. It is important to say, however, that many rules were kept even if they are not totally accurate. In fact, rules were kept when they were considered to be useful without conflicting more important rules. It is reasonable, since the rules will be merged with other KSs in our WSD proposed system.

```
R4        IF bwr_1 = to
 THEN CLASS = vir [0.1753, 0.0928, 0.3299,
0.4021, 97] ?[0.4186, 0.5814, 0.0000, 0.0000, 86]
```

Figure 2: Example of removed rule

After removing 161 uninteresting rules, we grouped rules with the same head and body, amounting to 68 rules. As mentioned, some repeated rules were produced due to features in common in different settings, and also to the decision of experimenting with two versions of filters and pruning. The resulting number of rules for each verb, along with some examples of selected rules for different feature settings and filters, and the values of the four measures for such rules (the filtered version with N=3), is shown in Table 5[1]. Great part of the 68 selected rules addresses phrasal verbs, but there are also other kinds of interesting rules, such as the third for *to come*, the first for *to get*, and the third for *to look*.

In most of the cases, although testing different features, the rules are in fact analyzing the 1-3 words to the right of the verb, as well as the POS of those 1-3 words, and only in a few cases, 1-2 words and POS to the left of the verb. So, even though the rules employ features referring to content-word

[1] Feature names are composed by the kind of feature (bw = bag-of-words, col = collocations, cw = content words, p = part-of-speech (of bw = bag-of-words or cw = content words) and, except for collocations, the side of the feature with relation to the verb in the sentence (r = right, l = left), and a number indicating that the feature is the *n-th* word ith relation to the verb in the sentence. As for collocations, col_1 = the first word to the right and col_11 = the first preposition to the right. The POS tags used here are: jj = adjective, nn = common noun, dt = determiner, and in = preposition.

windows or bag-of-words, those features are working much more like collocations. This gives a clear indication about which KSs are being effectively used by the rules. In this sense, analyzing the rules also contributes to identify the appropriate KSs for disambiguation models in machine translation (our second goal). Comparing the KSs used in this individual rules analysis to those with best precision in the complete model evaluation previously carried out, the individual rules analysis corroborates that evaluation with respect to the first more relevant KS: collocations (S3). However, the complete model evaluation also pointed to S4 and S2 with very similar accuracies, but here the syntactic relations, comprised by both settings, are seldom used by the rules and, as mentioned, the bag-of-words in S2 work like collocations, since they do not identify the topic of the sentence.

Looking more carefully into the body of the rules, we realized that the 1-3 words and POS to the right of the verb, though referring to as different features, sometimes are equivalent. For example, cwr_1 and col_1 will be the same if the first word to the right of the verb (col_1) is a content word (crw_1). However, we did not group this rules, since this could be harmful considering their use for new instances. On the other hand, we manually changed rules in two situations: (1) removing one or more of the tested features when they were not necessary; (2) grouping rules if they become equal after the changes in (1). For example, we had selected two rules for the verb *to go*, with the feature setting S5, both testing if the first word to the right of the verb was *out*, and testing different subjects for the verb: *I* and *he*. We consider that the subject is not important here, so we removed this test from both rules and then grouped them into one rule: *IF cwr_1 = out THEN CLASS = sair*. With this procedure, 3 rules for *to go* and 2 for *to look* were eliminated. Hence, the new number of rules to be effectively used as KS in our proposed WSD model was 63.

4 Conclusions

We described a systematic evaluation of rules automatically produced for WSD. This kind of evaluation, in which individual rules are examined using both objective and subjective criteria, has not been performed in WSD so far. Moreover, the idea behind the evaluation, i.e., getting high-quality rules to be employed as KS in a relational WSD system, has never been explored, given that all the corpus-based works in WSD make use of propositional formalisms, which do not allow rules as KS to the machine learning process.

		Examples		
Verb	**rules**	**KS**	**N**	**Rules**[2]
come	16	S1	1, 3	IF bwr_1 = back THEN CLASS = voltar
		S3, S4	1	IF col_1 = out THEN CLASS = sair
		S3	3	IF col_1 = here THEN CLASS = vir
get	8	S5	3	IF pcwr_1 = jj THEN CLASS = ficar
		S3, S4	1, 3	IF col_11 = to AND col_1 = back THEN CLASS = voltar
give	2	S5	1	IF cwr_1 = birth THEN CLASS = dar
go	17	S1, S2	3	IF bwr_1 = there THEN CLASS = ir
		S5	3	IF cwr_1 = to AND pcwr_2 = nn THEN CLASS = ir
look	12	S3, S4	1, 3	IF col_11 = like THEN CLASS = parecer
		S3, S4	1, 3	IF col_11 = for THEN CLASS = procurar
		S5	3	IF pcwr_2 = jj THEN CLASS = parecer
make	6	S1, S2	3	IF bwr_2 = mistake AND pbwr_1 = dt AND pbwr_3 = in THEN CLASS = cometer
		S3, S4	3	IF col_5 = decision AND col_11 = about THEN CLASS = decidir
take	7	S1, S2	1, 3	IF bwr_2 = to THEN CLASS = lever
		S1, S2	3	IF bwr_2 = of AND bwr_1 = advantage THEN CLASS = aproveitar
		S3, S4	1, 3	IF col_1 = off THEN CLASS = tirar

Table 5: Number of resultant rules and examples

Although the criteria for the objective measures were empirically defined and thus may be different

[2] Feature names are composed by the kind of feature (bw = bag-of-words, col = collocations, cw = content words, p = part-of-speech (of bw = bag-of-words or cw = content words) and, except for collocations, the side of the feature with relation to the verb in the sentence (r = right, l = left), and a number indicating that the feature is the *n-th* word ith relation to the verb in the sentence. As for the collocation, col_1 = the first word to the right and col_11 = the first preposition to the right. The POS tags used here are: jj = adjective, nn = common noun, dt = determiner, and in = preposition.

for other WSD works, the evaluation in two steps, one quantitative followed by one qualitative, showed to be appropriate. The first step reduced significantly the number of rules and, consequently, the amount of manual work needed. The second step allowed a deeper analysis on the quality of the rules, proving that even high accurate, new and wide coverage rules can be useless and uninteresting. Hence, we consider that objective and subjective measures are complementary.

As result, we got 63 high-quality rules satisfying criteria established for measures of both nature. We consider that they can represent important KS in our proposed model and in future work we will evaluate these rules extrinsically in the context of that model.

Although we addressed English-Portuguese in this work, both rule generation and evaluation methodologies are language independent. In order to be applied to other language pairs, one would only need a sample corpus of the source language, tagged with the target language translations for the words under consideration and other desired information, such as POS tags.

References

Agirre, E. and Martínez, D. 2001. Knowledge Sources for Word Sense Disambiguation. In *Proceedings of the Fourth International Conference on Text Speech and Dialogue*, Plzen.

Agirre, E. and Stevenson, M. 2005. (in press) Knowledge Sources for Word Sense Disambiguation. In Edmonds, P. and Agirre, E (eds), *Word Sense Disambiguation: Algorithms, Applications and Trends*, Kluwer.

Batista, G.E.A.P.A. and Monard, M.C. 2004. Sniffer: um Ambiente Computacional para Gerenciamento de Expe-rimentos de Aprendizado de Máquina Supervisionado. In *Proceedings of the I WorkComp Sul*, Florianópolis.

Dorr, B.J. and Katsova, M. 1998. Lexical Selection for Cross-Language Applications: Combining LCS with WordNet. In *Proceedings of AMTA'1998, Langhorne*, pp. 438-447.

Frankenberg-Garcia, A. and Santos, D. 2003. Introducing COMPARA: the Portuguese-English Parallel Corpus. *Corpora in translator education*, pp. 71-87.

Lavrac, N., Flach, P., and Zupan, B. 1999. Rule Evaluation Measures: A Unifying View. In *Proceedings of the 9th International Workshop on Inductive Logic Programming*. Lecture Notes in AI, v. 1634, pp. 174-185.

Lee, H. 2002. Classification Approach to Word Selection in Machine Translation. In *Proceedings of AMTA'2002*, Berlin, pp. 114-123.

Lee, Y.K. and Ng, H.T. 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. In *Proceedings of the Conference on Empirical Methods in NLP*, Philadelphia.

Mohammad, S. and Pedersen, T. 2004. Combining Lexical and Syntactic Features for Supervised Word Sense Disambiguation. In *Proceedings of the Conference on Computational Natural Language Learning*, Boston.

Mooney R.J. 1997. Inductive Logic Programming for Natural Language Processing. In *Proceedings of the 6th International Inductive Logic Programming Workshop*, Berlin, pp. 3-24.

Paula, M.F. 2003. *Ambiente para exploração de regras*. Dissertação de Mertrado em Ciência da Computação. Instituto de Ciências Matemáticas e de Computação, USP, São Carlos.

Prati, R.C, Geronimi, M.R., and Monard, M.C. 2003. An Integrated Environment for Data Mining. In *Proceedings of the IV Congress of Logic Applied to Technology* (LAPTEC-2003), Marília.

Quinlan, J.R. 1988. *C4.5 Programs for Machine Learning*. Morgan Kaufmann, CA.

Silberschatz, A. and Tuzhilin, A. 1996. What Makes Patterns Interesting in Knowledge Discovery Systems. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):970-974.

Specia, L. 2005a. A Hybrid Model for Word Sense Disambiguation in English-Portuguese Machine Translation. In *Proceedings of the 8th Research Colloquium of the UK Special-interest Group in Computational Linguistics*, Manchester, pp. 71-78.

Specia, L. 2005b. Knowledge sources for disambiguating highly ambiguous verbs in machine translation. In Proceedings of the 17th European Summer School in Logic, Language and Information, ESSLLI-2005, Edinburgh.

Specia, L., Oliveira-Netto, S., Nunes, M.G.V. and Stevenson, M. 2005. An Automatic Approach to Create a Sense Tagged Corpus for Word Sense Disambiguation in Machine Translation. In *Proceedings of the 2nd Meaning Workshop*, Trento, pp. 31-36.

Stevenson, M. and Wilks, Y. 2001. The Interaction of Knowledge Sources in Word Sense Disambiguation. *Computational Linguistics*, 27(3):321-349.

Zinovjeva, N. 2000. *Learning Sense Disambiguation Rules for Machine Translation*. Master's Thesis in Language Engineering. Department of Linguistics, Uppsala University.