Multilingual Generation of Numeral Classifiers using a Common Ontology

Kyonghee Paik
ATR Spoken Language Translation Research Lab
2-2-2 Hikari-dai, Kyoto 619-0237, JAPAN
kpaik@slt.atr.co.jp

Francis Bond
NTT Communication Science Labs
2-4 Hikari-dai, Kyoto 619-0237, JAPAN
bond@cslab.kecl.ntt.co.jp

Abstract

In this paper, we present a solution to the problem of generating both Japanese and Korean numeral classifiers using semantic classes from an ontology. Most nouns must use a numeral classifier when they are quantified in languages such as Chinese, Japanese, Korean, Malay and Thai. In order to select an appropriate classifier, we propose an algorithm which associates classifiers with semantic classes and uses inheritance to list only exceptional classifiers with individual nouns. The algorithm generates sortal classifiers with an accuracy of 81%. We reuse the ontology provided by Goi-Taikei — a Japanese lexicon, and show that it is a reasonable choice for this task, requiring information to be entered for less than 6% of individual nouns.

Keywords: multilingual generation, numeral classifiers, Japanese, Korean

1 Introduction

In this paper we consider two questions. The first is: how to generate numeral classifiers such as *piece* in *2 pieces of paper*? To do this we use a semantic hierarchy originally developed for a different task. The second is: how far can such a hierarchy be used with different languages?

In English, uncountable nouns cannot be directly modified by numerals, instead the noun must be embedded in a noun phrase headed by a classifier. Knowing when to do this is a language specific property. For example, French *deux renseignement* must be translated as *two pieces of information* in English.[1] In many languages, including most South-East Asian languages, Chinese, Japanese and Korean, the majority of nouns are uncountable and must be quantified by numeral-classifier combinations. These languages typically have many different classifiers. There has been some work on the analysis of numeral classifiers in natural language processing, particularly for Japanese [1, 2, 3, 4], but less on their generation [5, 6]. One immediate application for the generation of classifiers is machine translation, and we shall take examples from there, but it is in fact needed for the generation of any quantified noun phrase with an uncountable head noun.

The second question we address is: how far can an ontology be reused for a different task and language than the one it was originally designed for. There are several large ontologies now in use (WordNet [7]; Goi-Taikei [8]; Mikrokosmos [9]) and it is impractical to rebuild one for every language and application. However, there is no guarantee that an ontology built for one task will be useful for another.

The paper is structured as follows. In Section 2, we discuss the properties of numeral classifiers in more detail and suggest an improved algorithm for generating them. Section 3 introduces the ontology we have chosen, the Goi-Taikei ontology [8]. Then we show how to use the ontology to generate classifiers in Section 4. Finally, we discuss how well it performs in Section 5.

2 Generating Numeral Classifiers

In this section we introduce the properties of numeral classifiers, focusing on Japanese and Korean; then give an algorithm to generate classifiers. Japanese was chosen because of the wealth of published data on Japanese classifiers [10, 11, 12, 13, 14] and the availability of a large lexicon with semantic classes marked; Korean was chosen to test the applicability of the algorithm to multiple languages. Unless otherwise indicated, examples will be in Japanese.

2.1 What are Numeral Classifiers

Japanese and Korean are languages where most nouns can not be directly modified by numerals. Instead, nouns are modified by numeral-classifier combinations as shown in (1).[2]

(1) 2-tsū-no denshimēru (Japanese)
2-tong-ui imeil (Korean)
2-CL-ADN email
"2 pieces of email"
"2 emails"

Numeral classifiers are a subclass of nouns. The main property distinguishing them from prototypical nouns is that they cannot stand alone. Typically they postfix to numerals, forming a quantifier phrase. Japanese also allows them

[1] Numeral-classifier combinations are shown underlined by double underlines, the noun phrases they quantify are single-underlined.

[2] We use the following abbreviations: NOM = nominative; ACC = accusative; ADN = adnominal; CL = classifier; ARGSTR = argument structure; ARG = argument; D-ARG = default argument, QUANT = quantification.

to combine with the quantifier *sū* "some" or the interrogative *nani* "what" (2). Korean allows them to postfix to numerals, the interrogative *myech* "what" and the quantifier *yeoreo* "many". We will call all such combinations of a numeral/quantifier/interrogative with a numeral classifier a numeral-classifier combination.

(2) Japanese

a. *2-hiki* "2 animals" (Numeral)

b. *sū-hiki* "some animals" (Quantifier)

c. *nan-biki* "how many animals" (Interrogative)

(3) Korean

a. *2-mari* "2 animals" (Numeral)

b. *myetch-mari* "some animals" (Quantifier)

c. *myetch-mari* "how many animals" (Interrogative)

Classifiers have different properties depending on their use. There are five major types: **sortal** which classify the kind of the noun phrase they quantify (such as *-tsu* "piece"); **event** which are used to quantify events (such as *-kai* "time"); **mensural** which are used to measure the amount of some property (such as *senchi* "-cm"), **group** which refer to a collection of members (such as *-mure* "group"); and **taxonomic** which force the noun phrase to be interpreted as a generic kind (such as *-shu* "kind"). In this paper we are concerned with the generation of sortal classifiers.

We propose the following basic structure for sortal classifiers (4). The lexical structure we adopt is an extension of Pustejovsky's (1995) generative lexicon, with the addition of an explicit quantification relationship [16].

(4)
$$\textit{classifier}\left[\begin{array}{ll}\text{ARGSTR} & \left[\begin{array}{ll}\text{ARG1} & \texttt{x:numeral+}\\ \text{D-ARG1} & \texttt{y: ?}\end{array}\right]\\ \text{QUANT} & \texttt{quantifies(x,y)}\end{array}\right]$$

There are two variables in the argument structure: the numeral, quantifier or interrogative (represented by `numeral+`), and the noun phrase being classified. Because the noun phrase being classified can be omitted in context, it is a default argument, one which participates in the logical expressions in the word's semantics (its qualia structure), but is not necessarily expressed syntactically.

Sortal classifiers differ from each other in the restrictions they place on the quantified variable `y`. For example the classifier *-nin* adds the restriction `y:human`. That is, it can only be used to classify human referents.

Japanese has two number systems: a Sino-Japanese one based on Chinese (*ichi* "one", *ni* "two", *san* "three", ...), and an alternative native system (*hitotsu* "one" *futatsu* "two", *mitsu* "three", ...). In Japanese the native system only exists for the numbers from one to ten. Most classifiers combine with the Chinese forms, for example, *ni-hiki* "two-cl", and most classifiers undergo some form of sound change (such as *-hiki* to *-biki* in (2)). However, some classifiers can select native forms for some numerals: e.g., *shichi-nin* "seven people" (Chinese) vs *nana-nin* "seven people" (native). We will not be concerned with these morphological changes, we refer interested readers to Backhouse [17, 118–122] for more discussion.

Korean also has two number systems: a Sino-Korean one based on Chinese (*il* "one", *i* "two", *sam* "three", ...) and the native system (*han* "one", *tu* "two", *sey* "three", ...), which can count up to ninety-nine in modern Korean. However, old Korean had more numerals: *on* "one-hundred", *zumun* "thousand", *gol* "ten-thousand", *jal* "million", ... [18]. In general, Sino-Korean numeral classifiers tend to be used with the Sino-Korean numerals and native numeral classifiers with the native numerals. Some classifiers have two variants: a Sino-Korean form and a native form (e.g. the classifer for people which can be either *-myeong* or *-saram*). The Chinese form is normally used in more formal registers, the native form in more relaxed ones. Events can be counted with two classifiers: *-hoi* and *-beon*. *-hoi* is Sino-Korean, and is normally used with Sino-Korean numerals. *-beon* has two separate uses. The combination of a native numeral and *-beon* such as *han-beon* "once" is used for counting events. However, a combination of a Sino numeral and *-beon* such as *il-beon* "first", is used as an ordinal counter.

Numeral classifiers characteristically premodify the noun phrases they quantify, linked by an adnominal case marker, as in (5); or appear 'floating' as adverbial phrases, typically to before the verb: (6). The choice between pre-nominal and floating quantifiers is largely driven by discourse related considerations [13]. In this paper we concentrate on the semantic contribution of the quantifiers, and ignore the discourse effects.

(5) 2-tsū-no tegami-o yonda
2-CL-ADN letter-ACC read
"I read two letters"

(6) tegami-o 2-tsū yonda
letter-ACC 2-CL read
"I read two letters"

Quantifier phrases can also function as noun phrases on their own, with anaphoric or deictic reference, when what is being quantified is recoverable from the context. For example (8) is acceptable if the letters have already been referred to, or are clearly visible.

(7) [some background with letters salient]

(8) 2-tsū-o yonda (Japanese)
2-CL-ACC read
"I read two letters"

In the pre-nominal construction the relation between the target noun phrase and quantifier is explicit. For numeral-classifier combinations the quantification can be of the object denoted by the noun phrase itself as in (9); or of a sub-part of it as in (10) (see Bond and Paik [16] for a fuller discussion).

(9) 3-tsū-no tegami
3-CL-ADN letter
"3 letters"

(10) 3-mai-no tegami
3-CL-ADN letter
"a 3 page letter"

2.2 An Algorithm to Generate Numeral Classifiers

We will use the algorithm given in Bond and Paik [6], an extension of the algorithm proposed by Sornlertlamvanich et al. [5]. The algorithm is shown in Figure 1.

The algorithm can be used when a noun is a member of more than one semantic class or of no semantic class. In the lexicon we used, nouns are, on average, members of 2 semantic classes. However, the semantic classes are ordered so that the most basic class comes first [8, vol 1, p25]. For example, *usagi* "rabbit" is marked as both `animal` and `meat`, with `animal` coming first (Figure 3).[3] During contextual processing, other semantic classes may become more salient, in which case they will be used to select the default classifier.

The algorithm can also handle the generation of classifiers that quantify coordinate noun phrases. These commonly appear in appositive noun phrases such as *ABC-to XYC-no 2-sha* "the two companies, ABC and XYZ".

1. For a simple noun phrase
   (a) If the head noun has a default classifier in the lexicon:
   use the noun's default classifier
   (b) Else if it exists, use the default classifier of the head noun's most salient semantic class (the class's default classifier)
   (c) Else use the **residual** classifier
   (つ *-tsu* for Japanese; 개 *-kae* for Korean)
2. For a coordinate noun phrase
   generate the classifier for each noun phrase
   use the most frequent classifier

Figure 1: Algorithm to generate numeral classifiers

If a noun's default classifier is the same as the default classifier for its semantic class, then there is no need to list it in the lexicon. This makes the lexicon smaller and it is easier to add new entries. Any display of the lexical item (such as for maintenance or if the lexicon is used as a human aid), should automatically generate the classifier from the semantic class. Alternatively (and equivalently), in a lexicon with multiple inheritance and defaults, the class's default classifier can be added as a defeasible constraint on all members of the semantic class.

3 The Goi-Taikei Ontology

We used the ontology provided by Goi-Taikei — A Japanese Lexicon [8]. We choose it because of its rich ontology, its extensive use in many other NLP applications, its wide coverage of Japanese, and the fact that it is being extended to other numeral classifier languages, such as Malay and Chinese.

The ontology has several hierarchies of concepts: with both `is-a` and `has-a` relationships. 2,710 semantic classes (12-level tree structure) for common nouns, 200 classes (9-level tree structure) for proper nouns and 108 classes for predicates. We show the top four levels of the common noun ontology in Figure 2. Words can be assigned to semantic classes anywhere in the hierarchy. Not all semantic classes have words assigned to them.

The semantic classes are used in the Japanese word semantic dictionary to classify nouns, verbs and adjectives. The dictionary includes 100,000 common nouns, 70,000 technical terms, 200,000 proper nouns and 30,000 other words: 400,000 words in all. The semantic classes are also used as selectional restrictions on the arguments of predicates in a separate predicate dictionary, with around 17,000 entries.

Figure 3 shows an example of one record of the Japanese semantic word dictionary, with the addition of the new DEFAULT CLASSIFIER field (underlined for emphasis).

record

INDEX FORM	ウサギ (*usagi*)
PRONUNCIATION	うさぎ /usagi/
CANONICAL FORM	兎 (*usagi*)
PART OF SPEECH	noun
DEFAULT CLASSIFIER	羽 (*-wa*)
SEMANTIC CLASSES	NOUN `537:beast`, `843:meat/egg`

***Figure 3: Japanese Lexical Entry for** rabbit* "usagi"

Each record has an index form, pronunciation, a canonical form, part-of-speech and semantic classes. Each word can have up to five common noun classes and ten proper noun classes. In the case of *usagi* "rabbit", there are two common noun classes and no proper noun classes. The semantic classes are listed in order of salience (as judged by the dictionary compilers).

4 Mapping Classifiers to the Ontology

In this section we investigate how far the semantic classes can be used to predict default classifiers for Japanese and Korean. Because most sortal classifiers select for some kind of semantic class, nouns grouped together under the same semantic class will typically share the same classifier.

We associated classifiers with each of the 2,710 semantic classes by hand. This took around two weeks for Japanese. We found that, while some classes were covered by a single classifier, around 20% required more than one. For example, `1056:song` is counted only by *-kyoku* "tune", and `989:water vehicle` only by *-seki* "ship", but the class `961:weapon` had members counted by *-hon* "long thin", *-chō* "knife", *-furi* "swords", *-ki* "machines" and more. Adding the mapping for Korean took 3 days. It was faster than mapping the Japanese because we started off by translating the Japanese classifiers to Korean, and then checking them, which was quicker than assigning them from scratch.

[3] However, in the case of *usagi* it is not counted with the default classifier for animals *-hiki*, but with that for birds *-wa*, this must be listed as an exception.

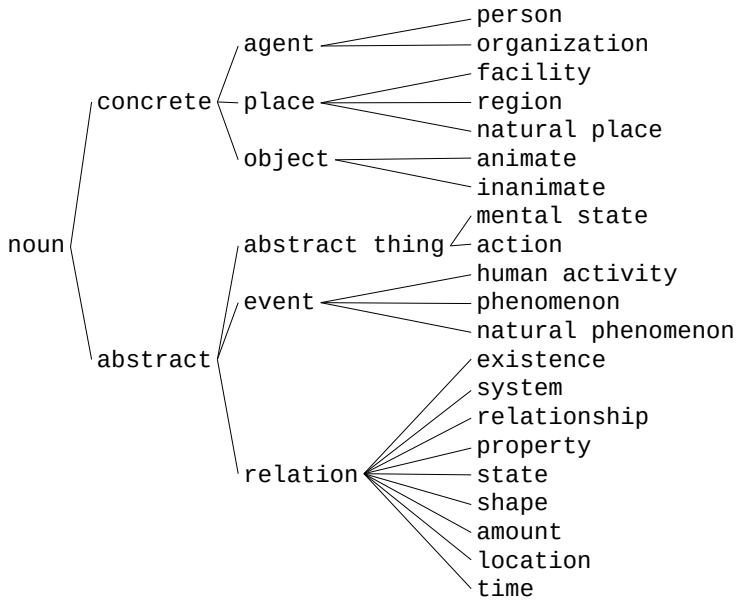

Figure 2: Top four levels of the Goi-Taikei Common Noun Ontology

We show the most frequent numeral classifiers for Japanese in Table 1, and for Korean in Table 2. We ended up with 47 classifiers used as semantic classes' default classifiers for Japanese. This is in line with the fact that most speakers of Japanese know and use between 30 and 80 sortal classifiers [13]. Note that we expect to add more specific classifiers at the noun level.

794 semantic classes were not assigned classifiers. This included classes with no words associated with them, and those that only contained nouns with referents so abstract we considered them to be uncountable, such as greed, lethargy, etc.

The mapping we created is not complete because some of the semantic classes have nouns which do not share the same classifiers. In order to generate classifiers accurately it is necessary to add more specific defaults at the noun level (noun default classifiers). As well as more specific sortal classifiers, there are cases where a group classifier may be more appropriate. For example, among the nouns counted with *-nin* there are entries such as couple, twins and so on which are often counted with *-kumi* "pair".

In addition, the choice of classifier can depend on factors other than just semantic class, for example, in Japanese, *hito* "people" can be counted by either *-nin* or *-mei*, the only difference being that *-mei* implies that the referent being counted is of higher status. Similarly, in Korean, people can be counted by either *-myeong* or *-saram*, with the Chinese form (*-myeong*) normally used in more formal registers, the native form in more relaxed ones.

It was difficult to assign default classifiers to the semantic classes that referred to events. These classes mainly include deverbal nouns (e.g. *konomi* "liking") and nominal verbs (e.g., *benkyō* "study"). The deverbal nouns can stand for both the action or the result of the action: e.g. *kenkyū* "a study/research". In these cases, every application we considered would distinguish between event and sortal classification in the input, so it was only necessary to choose a classifier for the result of the action.

The most frequent numeral classifiers for Korean are shown in Table 2. Even though the similarity between Japanese and Korean, we find some difference in both ranking and the kinds of numeral classifiers. First of all, as we can see, the most frequent classifier is *-kae*. This is because Korean has only one residual classifier, unlike Japanese which has *-tsu* and *-ko*. Shimojo [19] shows that the borrowing of Chinese classifiers and numerals into Japanese started from the 6th century. Since then the Chinese numeral classifier *-ko* has gradually taken the place of the native Japanese *-tsu*. Korean has only one residual classifier (the Sino-Korean *-kae*, which uses the same Chinese character as Japanese *-ko*: 個) and combines quite freely with both Sino and native Korean numerals. Another reason why *-kae* is used a lot is that many nouns counted with the Japanese shape classifiers *-hon* "long-thin object" and *-mai* "flat object" are counted by the residual classifier *-kae* in Korean.

In addition, Korean has far fewer 'other' type classifiers: 26 compared to Japanese 38. Half of these lesser used classifiers are native Korean ones. For example, *-karak* "long-thin-flexible", *-kaji* "type", *-kuru* "tree", *-al* "egg", *-beol* "clothes", *-chae* "building", *-pogi* "plant", *-jul* "line", *-teongeori* "mass", and *-songi* "flower".

5 Evaluation and Discussion

The algorithm was tested on a 3700 sentence machine translation test set of Japanese with English and Korean translations, although we did not use the English.[4]

[4]The Japanese and English parts of the test set are available at www.kecl.ntt.co.jp/icl/mtg/resources.

CLASSIFIER		Referents classified	No.	%	Sample Semantic Class
None		Uncountable referents	794	29.3	`3:agent`
-kai	(回)	events	703	25.9	`1699:visit`
-tsu	(つ)	abstract/general objects	565	20.9	`2:concrete`
-nin	(人)	people	298	11.0	`5:person`
-ko	(個)	concrete objects	124	4.6	`854:edible fruit`
-hon	(本)	long thin objects	52	1.9	`673:tree`
-mai	(枚)	flat objects	32	1.2	`770:paper`
-teki	(滴)	liquid	21	0.8	`652:tear`
-dai	(台)	mechanic items/ furniture	18	0.7	`962:machinery`
-hiki	(匹)	animals	12	0.6	`537:beast`
Other		38 classifiers	91	3.4	

Table 1: Japanese Numeral Classifiers and associated Semantic Classes

CLASSIFIER		Referents classified	No.	%	Sample Semantic Class
None		Uncountable referents	799	29.5	`3:agent`
-kae	(개)	abstract/general objects	737	27.1	`2:concrete`
-hyoi	(회)	events	707	26.1	`1699:visit`
-myong	(명)	people	296	10.9	`5:person`
-bangul	(방울)	liquid	26	1.0	`652:tear`
-jang	(장)	flat objects	24	0.9	`770:paper`
-dae	(대)	mechanic items/ furniture	20	0.7	`962:machinery`
-keun	(건)	incidents	14	0.5	`1717:contract`
-mari	(마리)	animals	14	0.5	`537:beast`
Other		26 classifiers	73	2.7	

Table 2: Korean Numeral Classifiers and associated Semantic Classes

We only considered sentences with a noun phrase modified by a sortal classifier. Noun phrases modified by group classifiers, such as *-soku* "pair" were not evaluated, as we reasoned that the presence of such a classifier would be marked in the input to the generator. We also did not consider the anaphoric use of numeral classifiers. Although there were many anaphoric examples, resolving them requires robust anaphor resolution, which is a separate problem. We estimate that we would achieve the same accuracy with the anaphoric examples if their referents were known, unfortunately the test set did not always include the full context, so we could not identify the referents and test this. A typical example of anaphoric use is (11: test sentence 20300) In this case, without knowing the referent, there is no principled way to choose a translation, and the best choice would be to use the residual classifier.[5]

(11) 出荷が 累積で 五百本を 突破した
출하가 누적하여 500본을 돌파했다
shipment-NOM cumulative 500-CL-ACC reached
"Cumulative shipments reached 500 units (?barrels/rolls/logs/...)"

In total, there were 90 noun phrases modified by a sortal classifier. Our test of the algorithm was done by hand, as we have no Japanese or Korean generator. We assumed as input only two features: the fact that a classifier was required; and the semantic classes of the head noun given in the lexicon. We assumed that the Korean nouns would have the same semantic classes as their Japanese translation equivalents.

Using only the default classifiers predicted by the semantic class, we were able to generate 73 Japanese classifiers (81%) and 56 Korean classifiers (62%) correctly. A classifier was only judged to be correct if it was exactly the same as that in the original test set. This was almost double the base line of generating the most common classifier (*-nin*/*-saram*) for all noun phrases, which would have achieved 41%. The results, with a breakdown of the errors, are summarised in Table 3.

Result	Japanese		Korean	
	%	No.	%	No.
Correctly generated	81%	73	62%	56
Wrong register	–	–	6%	5
Incorrectly generated	19%	17	13%	12
Generated when not needed	–	–	19%	17
Total	100%	90	100%	90
Breakdown of Errors				
Needs default classifier	—	6	—	1
Target lexicon bad	—	4	—	3
Residual or no classifier	—	2	—	8
No-classifier Construction	—	—	—	9
Other errors	—	5	—	8

Table 3: Results of applying the algorithm

[5]This was actually manually translated into Korean as *500-bon*, even though there is no such word in Korean.

Two examples of correct generation of classifiers are given below. In (12), the semantic class of the target is `511:star/planet`, and the predicted classifier is 個 *-ko* for Japanese and 개 *-kae* for Korean. In (13) the semantic class is `971:computer` and the predicted classifiers are 台 *-dai* and 대 *-dae*. Korean and Japanese use equivalent classifiers in these examples.

(12) 太陽には 少なくとも8個の 惑星が
태양에는 적어도 8개의 혹성이
sun-LOC-TOP at least 8-CL-ADN planet-NOM
ある
있다
has

"The sun has at least eight planets around it"

(13) その会社は 計算機を 3台売った
그 회사는 계산기를 3대 팔았다
that company-TOP computer-ACC 3-CL sold

"That company sold three computers"

In the Japanese test, for this small sample, 6 out of 93 (6.5%) of noun phrases needed to have the default classifier marked for the noun. In fact, there were only 4 different nouns, as two were repeated. We therefore estimate that fewer than 6% of nouns will need to have their own default classifier marked. Had the default classifier for these nouns been marked in the lexicon, our accuracy would have been 88%, the maximum achievable for our method.

We achieved worse results for Korean, mainly because of a faulty assumption: we assumed that we always had to generate a classifier, but in fact there were many cases (17: 19%) where a classifier was not needed in Korean. For 9 of the cases, a classifier was disallowed by the syntactic construction, which we had not considered. Simply adding such constructions to the Korean generation system would enable it to correctly refrain from generating the classifiers. A native speaker evaluator judged that a classifier would be acceptable for the remaining 8 cases, although it was not generated by the human translator. In Japanese, a residual classifier was used instead of the more specific default in two of these cases. Shimojo [19] predicts that this will happen in expressions where the amount is being emphasised more than what is being counted. Intuitively, this applied in both the Japanese and Korean cases, but we were unable to identify any features we could exploit to make this judgement automatically.

Five times we generated a good classifier but in the wrong register, that is we used the Sino-Korean classifier when the translator chose the native Korean one. For example, in (14) we generated 명 *-myeong* although 사람 *-saram* was used in the test set. The error was caused by our neglect of register in the generation process. We made the Sino-Korean numeral classifiers defaults because we judged that a formal register is a safer default. If we include these cases into the correctly generated category, then we were able to generate 68% correctly.

(14) 犯人は 少なくとも二人 以上 いた
범인은 적어도 두명 이상 있다
criminal-TOP at-least 2-CL more-than are

"There were at least more than two criminals involved"

There are nine cases where numeral classifiers are not allowed by the construction used in Korean, even though they are used in Japanese, as in (15).

(15) この町には 学校が ひとつも ない
이 마을에는 학교가 하나도 없다
this town-LOC school-NOM 1-(CL)-even has-not

"This town does not have a single school"

Interestingly, the English translation also does not use a numeral. In order to translate between Japanese and English, a constructions would also be useful, something like *N0-ha N1-ga 1-CL-mo nai* "N0 does not have a single N1". The equivalent Korean construction has no classifier: *N0-nun N1-ka 1-do eopda* "N0 does not have a single N1". These constructions should be listed in the grammar/lexicon.

Overall, the Goi-Taikei ontology, although initially designed for Japanese analysis, was also useful for generating not only Japanese numeral classifiers, but also Korean. The errors were not caused by flaws in the ontology. We predict that it would be equally useful for the same task with the unrelated language Malay.

In the Japanese test, we generated the residual classifier *-tsu* for nouns not in the lexicon, this proved to be a bad choice for three unknown words. If we had a method of deducing semantic classes for unknown words we could have used it to predict the classifier more successfully. For example, *kikan-tōshika* "institutional investor"[6] was not in the dictionary, and so we used the semantic class for *tōshika* "investor", which was `175:investor`, a sub-type of `5:person`. Had *kikan-tōshika* "institutional investor" been marked as a subtype of company, or if we had deduced the semantic class from the modifier, then we would have been able to generate the correct classifier *-sha*. In one case, we felt the default ordering of the semantic classes should have been reversed: `673:tree` was listed before `854:edible fruit` for *ringo* "apple": the fruit reading is the most common, and therefore the most basic.

The remaining errors were more problematic. There was one example, *80,000-nin-amari-no shōmei* "about 80,000 signatures", which could be treated as referent transfer: *shomei* "signature" was being counted with the classifier for people. Another possible analysis is that the classifier is the head of a referential noun phrase with deictic/anaphoric reference, equivalent to *the signatures of about 80,000 people*. A couple were quite literary in style: for example *10nen-no toshi* "10 years (Lit: 10 years of years)", where the *toshi* "year" part is redundant, and would not normally be used.

A more advanced semantic analysis may be able to dynamically determine the appropriate semantic class for cases of referent transfer, unknown words, or words whose semantic class can be restricted by context. Our algorithm, which ideally generates the classifier from this dynamically determined semantic class allows us to generate the correct classifier **in context**, whereas using a default listed for a noun

[6] Institutional investors are financial institutions that invest savings of individuals and non-financial companies in the financial markets.

does not. This was our original motivation for generating classifiers from semantic classes, rather than using a classifier listed with each noun as Sornlertlamvanich et al. [5] do.

So far we have concentrated on solving the problem of generating appropriate Japanese and Korean numeral classifiers using an ontology. In future work, we would like to investigate in more detail the conditions under which a classifier needs to be generated.

Finally, we would like to use the classifiers' selectional restrictions to disambiguate senses in analysis. For example, the Japanese word *denwa* can mean either "telephone call" (`1548:telephone-call`) or "telephone machine" (`970:communication machine`). The choice of classifier can be used to disambiguate them: one of the classes *-hon* selects for is `1548:telephone-call` $\subset$ `1544:communication`, as in (16); while *-dai* selects for `962:machine` $\supset$ `970:communication machine`, as in (17).

(16) 電話が　二本　来た
denwa-ga 2-hon kita
telephone 2-CL came
"Two telephone calls came."

(17) 電話が　二台　来た
denwa-ga 2-dai kita
telephone 2-CL came
"Two telephones arrived."

6 Conclusion

In this paper we presented an algorithm to generate Japanese and Korean numeral classifiers using a common ontology. It was shown to select the correct sortal classifier 81% of the time. The algorithm uses the ontology provided by Goi-Taikei — a Japanese lexicon, and shows how accurately semantic classes can predict numeral classifiers for the nouns they subsume. We also show some interesting differences between the use of numeral classifiers in Japanese and Korean.

References

[1] Asahioka, Y., Hirakawa, H. and Amano, S., Semantic classification and an analyzing system of Japanese numerical expressions, *IPSJ SIG Notes 90-NL-78*, 1990, 90(64), pp. 129–136, (in Japanese).

[2] Kamei, S. and Muraki, K., An analysis of NP-like quantifiers in Japanese, in *First Natural Language Processing Pacific Rim Symposium: NLPRS-95*, vol. 1, 1995, pp. 163–167.

[3] Bond, F., Ogura, K. and Ikehara, S., Classifiers in Japanese-to-English machine translation, in *16th International Conference on Computational Linguistics: COLING-96*, Copenhagen, 1996, pp. 125–130, (http://xxx.lanl.gov/abs/cmp-lg/9608014).

[4] Yokoyama, S. and Ochiai, T., Aimai-na sūryōshi-o fukumu meishiku-no kaisekihō [a method for analysing noun phrases with ambiguous quantifiers.], in *5th Annual Meeting of the Association for Natural Language Processing*, The Association for Natural Language Processing, 1999, pp. 550–553, (in Japanese).

[5] Sornlertlamvanich, V., Pantachat, W. and Meknavin, S., Classifier assignment by corpus-based approach, in *15th International Conference on Computational Linguistics: COLING-94*, 1994, pp. 556–561, (http://xxx.lanl.gov/abs/cmp-lg/9411027).

[6] Bond, F. and Paik, K., Re-using an ontology to generate numeral classifiers, in *18th International Conference on Computational Linguistics: COLING-2000*, Saarbrücken, 2000, pp. 90–96.

[7] Fellbaum, C., ed., *WordNet: An Electronic Lexical Database*, MIT Press, 1998.

[8] Ikehara, S., Miyazaki, M., Shirai, S., Yokoo, A., Nakaiwa, H., Ogura, K., Ooyama, Y. and Hayashi, Y., *Goi-Taikei — A Japanese Lexicon*, Iwanami Shoten, Tokyo, 1997, 5 volumes/CDROM.

[9] Nirenburg, S., KBMT-89 — a knowledge-based MT project at Carnegie Mellon University, 1989, pp. 141–147.

[10] Denny, J. P., Semantic analysis of selected Japanese numeral classifiers for units, *Linguistics*, 1979, pp. 317–335.

[11] Matsumoto, Y., A semantic structure and system for Japanese classifiers — based on prototype semantics –, *Gengo Kenkyu*, 1991, 99, pp. 82–106, (in Japanese).

[12] Matsumoto, Y., Japanese numeral classifiers: a study of semantic categories and lexical organization, *Linguistics*, 1993, 31, pp. 667–713.

[13] Downing, P., *Numeral Classifier Systems, the case of Japanese*, John Benjamins, Amsterdam, 1996.

[14] Alam, Y. S., Numeral classifiers as adverbs of quantification, in Sohn, H.-M. and Haig, J., eds., *Japanese/Korean Linguistics*, CSLI, vol. 6, 1997, pp. 381–397.

[15] Pustejovsky, J., *The Generative Lexicon*, MIT Press, 1995.

[16] Bond, F. and Paik, K., Classifying correspondence in Japanese and Korean, in *3rd Pacific Association for Computational Linguistics Conference: PACLING-97*, Meisei University, Tokyo, Japan, 1997, pp. 58–67.

[17] Backhouse, A. E., *The Japanese Language: An Introduction*, Oxford University Press, 1993.

[18] Suh, C.-S., *Hyundae Kukeo Munpupron [Contemporary Korean Grammar]*, Hanyung University Press, Seoul, Korea, 1996, (in Korean).

[19] Shimojo, M., The role of the general category in the maintenance of numeral classifier systems: The case of *tsu* and *ko* in Japanese, *Linguistics*, 1997, 35(4), pp. 705–733.