Automatic Linguistic Analysis for Language Teachers: The Case of Zeros

MITSUKO YAMURA-TAKEI
Graduate School of Information Sciences
Hiroshima City University
3-4-1 Ozuka-higashi, Asaminami-ku,
Hiroshima, JAPAN 731-3194
yamuram@nlp.its.hiroshima-cu.ac.jp

MIHO FUJIWARA
Department of Japanese and Chinese
Willamette University
900 State Street, Salem,
OR. USA 97301
mfujiwar@willamette.edu

MAKOTO YOSHIE
Graduate School of Information Sciences
Hiroshima City University
yoshie@nlp.its.hiroshima-cu.ac.jp

TERUAKI AIZAWA
Faculty of Information Sciences
Hiroshima City University
aizawa@its.hiroshima-cu.ac.jp

Abstract

This paper presents the Natural Language Processing-based linguistic analysis tool that we have developed for Japanese as a Second Language teachers. This program, Zero Detector (ZD), aims to promote effective instruction of zero anaphora, on the basis of a hypothesis about ideal conditions for second language acquisition, by making invisible zeros visible. ZD takes Japanese written narrative discourse as input and provides the zero-specified texts and their underlying structures as output. We evaluated ZD's performance in terms of its zero detecting accuracy. We also present an experimental report of its validity for practical use. As a result, ZD has proven to be pedagogically feasible in terms of its accuracy and its impact on effective instruction.

Introduction

Natural Language Processing (NLP) is an emerging technology with a variety of real-world applications. Computer-Assisted Language Learning/Teaching (CALL/CALT) is one area that NLP techniques can contribute to. Such techniques range from indexing and concordancing to morphological processing with on-demand dictionary look-ups and syntactic processing with diagnostic error analysis, to name a few. But little work has been done on discourse-level phenomena, including anaphora.

Zero anaphora or zero pronouns (henceforth zeros) are referential noun phrases (NPs) that are not overtly expressed in Japanese discourse. These NPs can be omitted if they are recoverable from a given context or relevant knowledge. The use of zeros is common in Japanese and this poses a challenge for Japanese as a Second Language (JSL) learners for their accurate comprehension and natural-sounding production of Japanese discourse with zeros. Some learners fail to understand a passage correctly because of the difficulty of identifying zeros and/or their antecedents. Other learners produce grammatically correct but still unnatural-sounding Japanese due to overuse or underuse of zeros.

Yet, very few textbooks provide systematic instruction or intensive exercises to overcome these difficulties with zeros. Consequently many Japanese language teachers rely on their intuitions when explaining zeros. Intuition is a conventional tool in teaching one's native language, but from a student's perspective, a well-developed systematic method of instruction can be more convincing. Also from a teacher's standpoint, such analysis will be helpful in preparing teaching materials and evaluating students' performance.

Analysis of zeros can be divided into three phases: zero identification, zero interpretation and zero production. This paper focuses on the first phase and proposes a method of systematically identifying the presence of zeros in order that teachers might provide effective instruction of zeros, based on some pedagogical principles from relevant second language acquisition (SLA) theory. We regard teachers as primary users of the program and aim to help them enhance their instruction. We implemented the program and evaluated its potential benefits for language teachers.

In Sections 1 and 2 we discuss the pedagogical assumptions from SLA theory that motivate our program design, and present the linguistic assumptions from which our heuristics were drawn. Section 3 provides an overview of our system implementation. In Section 4, we present the results of evaluation from the viewpoints of both the accuracy and the empirical validity of the program. We conclude with a discussion of possible future work.

1 Pedagogical Assumptions

There have been many studies about how people learn foreign languages and what is responsible for successful language learning.

Recent SLA theory progresses beyond Krashen (e.g., 1982)'s emphasis on automatic processes of acquisition. Empirical research has shown that learners' consciousness-raising through explicit instruction does contribute to successful second language learning (see Norris & Ortega, 2000 for comprehensive review).

Chapelle (1998) reviewed seven hypotheses about ideal SLA conditions that are relevant for CALL program design. At the top of her list is that "the linguistic characteristics of target language input need to be made salient" (p. 23). Effective input enhancement, by prompting learners to notice particular learning items, with highlighting for example, plays a significant role in facilitating acquisition. We conjecture that this salience effect can also be realized by making zeros visible.

2 Linguistic Assumptions

Japanese is a head-final language. A sentence or a clause is headed by a predicate, which takes a set of arguments and adjuncts. Predicates in Japanese include verbs, adjectives, nominal adjectives and copula, and usually consist of a core predicate and some auxiliary elements. Arguments are classified into three types: Topic Phrase (TP), headed by a topic marker *wa*, Focus Phrase (FP), headed by focus particles *mo*, *koso*, *dake*, *sae*, *shika*, etc., and Kase Phrase (KP), headed by case particles *ga*, *wo*, *ni*, *e*, *to*, *yori*, *de*, *kara*, and *made*. We regard adjuncts as non-particle-headed phrases.

We define zeros as unexpressed obligatory arguments of a core predicate. What is "obligatory" is the next question to arise. Obligatoriness is a controversial issue, and there is no set agreement among linguists on its definition. Somers (1984) proposed a six-level scale of valency binding that reflects the degree of closeness of an element to the predicate. The levels are (i) integral complements, (ii) obligatory complements, (iii) optional complements, (iv) middles, (v) adjuncts and (vi) extraperipherals. Ishiwata (1999) suggests that in Japanese group (i) is often treated as part of idioms and is not omissible, and Japanese nominative *–ga* and accusative *–wo* fall into the category (ii), while dative *–ni* belongs to (iii). In light of this, we assume that obligatory arguments that can be zero-pronominalized are phrases headed by nominative-case particle *ga* and accusative *wo*, and *ni*, excluding dative *ni* in an indirect object position.

3 Zero Detector

Zero Detector (henceforth ZD) is an automatic zero identifying tool, which takes Japanese written narrative texts as input and provides the zero-specified texts and their underlying structures as output. This aims to draw learners' and teachers' attention to zeros, by making these invisible elements visible in effectively enhanced formats.

3.1 System Overview

ZD employs a rule-based approach, with theoretically sound heuristics. Our heuristics are drawn from the linguistic assumptions described in Section 2.

ZD reuses and integrates two existing natural language analysis tools and an electronic dictionary, none of which were intended for a language learning purpose, into its architecture, attempting to make the best possible use of their

capabilities for our purpose. Morphological analysis is done by *ChaSen* 2.2.8 (NAIST, Matsumoto, Y. *et al*., 2001), and dependency structure analysis by *CaboCha* 0.21 (NAIST, Kudo, K., 2001). The Goi-Taikei Valency Dictionary (hereafter GTVD; Ikehara *et al*., 1997) serves as a source for valency pattern search.

The flow of the system is illustrated in Figure 1.

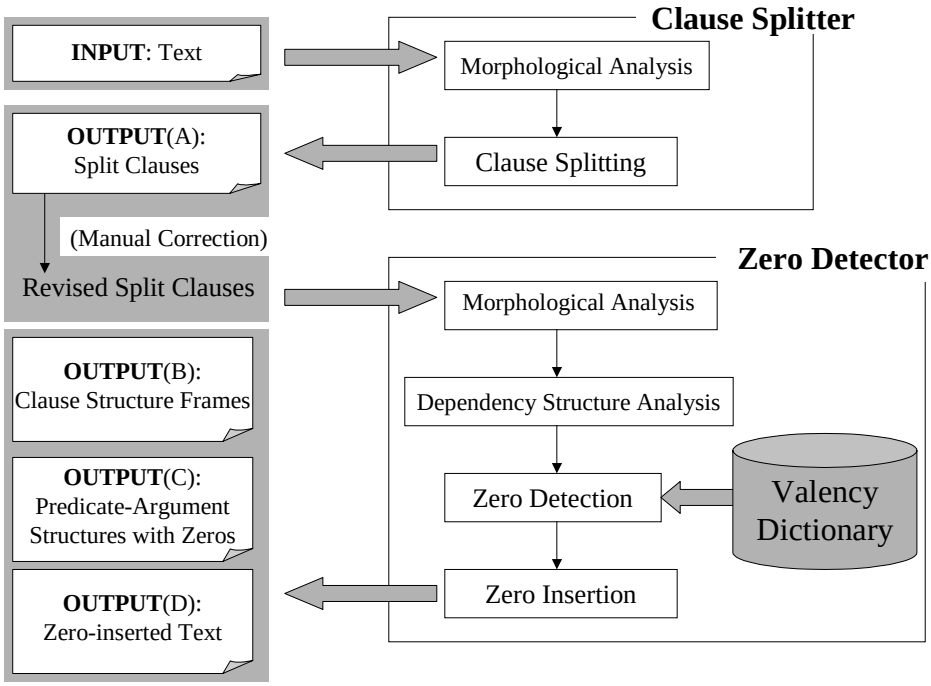

Figure 1: Flow diagram of zero detecting processes

3.2 ZD Output

As shown in Figure 1, ZD produces four different types of output: (A) split clauses, (B) clause structure frames, (C) predicate-argument structures with zeros, and (D) zero-inserted texts. We will show how these outputs are structured using the example text in Figure 2.

komatta Satsuki-wa sassoku
in trouble Satsuki-TOP immediately

gennin-wo shirabe-sase-ta.
cause-ACC investigate-CAUSATIVE-PAST

"Satsuki, who was in trouble, immediately had (someone) investigate its cause."

Figure 2: An example input text

First, output (A) provides a text divided into clauses, each consisting of one and only one predicate and its arguments. Some predicates are simplex, while others are complex, consisting of more than one core predicate (i.e., verb, adjective). Several complex predicates (e.g., *ta-beta-koto-ga-aru* ate-experience-subject marker-have, "have eaten") are predefined as simplex to avoid excessive clause splitting. The clauses are labelled with their clause types: independent (main), dependent (coordinated/subordinated) or embedded (relative/nominal/quoted). A clause serves as the basic unit for the zero detecting operation. In this study, embedded clauses are excluded from this operation and are left within their superordinate clauses. An example output (A) is given in Figure 3 (next page).

komatta EC(RC)] Satsuki-wa sassoku

gennin-wo shirabe-sase-mashita. IC]

Figure 3: Split clauses[1]

Once the text is split into clauses, each clause is analysed for its dependency structure and then converted into its clause structure frame. The noun phrases which depend on the predicate are extracted, and then classified into phrase types (TP, FP and KP) according to their accompanying particles. An example of this frame, i.e., output (B), is given in Figure 4.

Input: komatta Satsuki-wa sassoku gennin-wo shirabe-sase-ta.

Paragraph#: 2
Sentence#: 4
Clause#: 5
Clause Type: Independent with EC(RC)

[Predicate] : shirabe-sase-ta.
Core: shiraberu verb
Auxiliary: saseru verb
ta auxiliary verb
.
Voice: causative
Empathy:
Conjunction:

[Argument] :
Topic Phrase: komatta Satsuki-wa
Topic-Case: N1-ga
Focus Phrase: <none>
Focus-Case: <none>
Kase Phrase: gennin-wo
Pre-copula: <none>
[Adjunct] : sassoku

Figure 4: A clause structure frame

This frame also includes the result of valency checking, as in Figure 5, and zero identifying processes, as in Figure 6, at the bottom.

[1] Here, we use the acronyms: IC for Independent Clause, EC for Embedded Clause, and RC for Relative Clause.

Valency Selected: N1 ga N2 wo

Valency Obligatory: N1 ga N2 wo

Valency Changed: N1 ga N2 wo N3 ni

Figure 5: Valency checking

A core predicate is checked against GTVD to search for its syntactic valency pattern. GTVD is a semantic valency dictionary, originally designed for transfer-based Japanese-to-English machine translation, so it includes as many valency pattern entries for each predicate as are necessary for effective transfer. The entries are ordered according to expected frequency of occurrence. We took the naïve approach of selecting the first-ranking entry from the listing for each core predicate (i.e.,'Valency Selected' in Figure 5).

The next step is to apply the definition of 'obligatoriness' described in Section 2 to refine the selected valency pattern ('Valency Obligatory' in Figure 5). If non-*ga*, *wo*, or *ni* cases are within the first three case slots of the selected valency pattern, they are excluded. If a *ni*-case still remains in the third case slot, it is also deleted. These operations leave us two valency patterns: (i) N1-*ga* N2-*wo*, and (ii) N1-*ga* N2-*ni*, in most cases.

Then, a valency changing operation is done in the case of causatives or passives. When an auxiliary verb is added to the core predicate in the causative or passive construction, the verb then requires three arguments. In the causative case, these are a *ga*-marked causer, a *wo*-marked object and a *ni*-marked causee. The valency changing operation adds the boxed valent, N3 *ni*, in Figure 5 (Valency Changed) because the voice slot is marked as causative in Figure 4.

Valency Selected: N1 ga N2 wo

Valency Obligatory: N1 ga N2 wo

Valency Changed: ~~N1 ga~~ ~~N2 wo~~ N3 ni

Zero: N3 ni

Figure 6: Zero identifying

Now that the valency pattern for the given predicate is assigned, it is checked against overt arguments listed in the frame. The valent N2 is matched with the overt argument *gennin-wo* and removed from the zero candidates, as shown in Figure 6.

Case-less elements, such as TP and FP, also need to have their canonical case markers restored. This is done by assigning the first remaining valent to TP and/or FP. This is based on the linguistic fact that subjects are more likely to be topicalized or focused than objects. In the example, TP, *Satsuki-wa*, is assigned *ga* case. The assigned case slot N1-*ga* is then matched with *Satsuki-wa* (*ga*) and is also deleted.

Finally, the remaining valent, if any, is assumed to be a zero (i.e., N3 *ni* in Figure 6).

Once zeros are identified, ZD decides where to insert the identified zeros in the original text, by keeping canonical ordering as listed in the valency pattern. An example of the predicate-(obligatory) argument structure from Figure 6, with the identified zero, is presented in Figure 7. This is output (C). Here, the restored case marking particle is presented in parentheses.

*komatta Satsuki-wa (ga)

*gennin-wo

***[ni]**

*shirabe-sase-ta.

Figure 7: Predicate-argument structure with zeros

Finally, ZD outputs the original series of clauses with zeros inserted in the most plausible positions, along with adjuncts, output (D), as in Figure 8.

komatta Satsuki-wa sassoku gennin-wo **[ni]**

shirabe-sase-ta.

Figure 8: Zero-specified text

These outputs can later be converted into the form of a slide presentation or hard-copy handouts, etc., depending on how they are used by teachers.

4 Evaluation

The purpose of the evaluation was to assess the validity of ZD output for practical use in a language learning/teaching setting. In the following subsections, we evaluate ZD's performance in terms of its accuracy and then present an experimental report of its validity for educational use.

4.1 Performance

First, we compared the ZD output with human judgements. The test corpus consisted of two reading selections from a JSL textbook and one student written narrative monologue, all of which were representative samples for lower intermediate level Japanese. Five subjects (native speakers of Japanese and trained natural language researchers) served as our human zero detectors. They were asked to intuitively identify missing arguments in each clause. We used average human performance as a baseline against which to evaluate ZD output. Here, zeros detected by three or more, out of five, subjects were regarded as average human performance.

As Table 1 shows, ZD achieved a 73% per-clause matching rate with human output. That number represents the ratio of the number of exact matches between the two outputs over the total number of clauses.

Table 1: Per-clause matching rates

	# of clauses	**# of matched**
Reading (1)	30	22 (73%)
Reading (2)	25	18 (72%)
Writing	23	17 (74%)
Total	78	57 (73%)

A closer examination of each case element (*ga*, *wo*, *ni*) is given in Table 2 (next page). The level 'matched' includes both cases where ZD and human detect a zero and cases where neither detects it. The accuracy (89% average) is high enough for the ZD output to be put into practical use as a learning aid, without an excessive load on teachers for post-editing output errors. Releasing teachers from having to spend enormous amount of time on the tedious work of analysing educational materials is one of the biggest advantages of computerization of linguistic analysis.

Table 2: Per-case element matching rates

		が ga		を wo		に ni	
		Human	ZD	Human	ZD	Human	ZD
Matched	Detected	35	32	5	4	5	2
	Not Detected	43	39	73	68	73	63
	Total	78	**71 (91%)**	78	**72 (92%)**	78	**65 (83%)**
Not Matched	Under-detected		3		1		3
	Over-detected		4		5		10
	Total		**7 (9%)**		**6 (8%)**		**13 (17%)**

Also, we analysed 'not matched' cases to improve future performance. There were 26 cases of both underproduction and overproduction of zeros. Nearly half of them, 12 out of 26, were caused by our naïve valency selection algorithm, which selects the first entry from the GTVD valency pattern listing for each predicate. Three were caused by our canonical-case-marker-restoring heuristics, which assign a first available case marker from *ga* and *wo* in its preference order. They sometimes do not function properly when accusatives or adjuncts are topicalized (or focused). These are two major areas for future enhancement. Four cases were affected by morphological/sysntactic analyses. Also, our definition of obligatory arguments, which excludes dative *–ni*, produced three 'not matched' cases. This definition is also an issue for further consideration.

What should be noted here, on the other hand, is that there were six ZD produced zeros which did not match our human zero detectors' decision but whose validity was later confirmed by a JSL teacher who carefully examined the result from an instructional point of view. This implies that human-recognized zeros and linguistically/pedagogically plausible zeros do not always match, and demonstrates the potential of ZD to fill this gap.

4.2 Experiment

In order to verify the pedagogical effectiveness of ZD, the output files were experimentally used in a university-level intermediate JSL classroom, through digital presentation. The aim of this lesson was to familiarize the students with zeros by making these invisible elements visible in texts and presenting their underlying structures.

In their post-lesson feedback, the students showed a positive reaction to this analytic instruction. They described this approach as "innovative", "effective", "clear" and "easy" for understanding zeros, in contrast to their past "just guessing or being lost" experiences.

The teacher who conducted this experimental lesson also acknowledged the impact of ZD on effective instruction. She pointed out the following benefits for students:

(i) The valency checking segment of output (B) helps students realize that each predicate has its own valency pattern, and as a consequence, clarifies when to use what particles,

(ii) the predicate-argument structures with zeros, output (C), help students realize that locating zeros is not a random operation, but a canonical designation, and

(iii) the clause-by-clause parallel arrangement in output (D) facilitates realizing zero distributions in discourse and tracking down antecedents for each zero.

These include positive side effects that we initially did not foresee.

From a teaching point of view, ZD helps teachers predict the difficulties with zeros that students might encounter, by analysing text in advance. This leads to the careful selection of teaching materials and the well-thought-out creation of reading comprehension questions and tests. Also, ZD output will be helpful in explaining the illegal use of zeros and particles found in students' writing.

Conclusions and Future Work

We have developed an automatic zero detecting program that is intended mainly to serve as teacher support. The program has proven to be pedagogically feasible in terms of its accuracy and its impact on effective instruction. The great contribution of ZD is to introduce consistency and systematic analysis into an area where human intuitions play a dominant, but not always accurate and effective, role.

ZD is currently a purely syntactic-based tool that utilizes only surface-level heuristics, excluding any semantic cues. As our error analysis in Section 4 indicates, more accuracy can be achieved in a semantically enhanced version, which in fact is our next project goal. Valency-pattern-selecting (from GTVD) and canonical-case-marker-restoring (from TP and FP) algorithms are two major areas to which semantic information can greatly contribute.

Also, ZD has been designed as a teaching aid in a teacher-controlled class instruction mode. To extend its use to a self-study mode, as some students suggested, clear guidance and a user-friendly interface will be required to replace teachers' explanation.

ZD is a part of the CALL program for JSL learners, Zero Checker, which supports reading comprehension and writing revision process with a focus on zeros. Thus, ZD will also serve as a pre-processing module for the models of resolving and generating zeros, created within the centering framework (e.g., Grosz *et al*., 1995).

References

Chapelle, Carol A. (1998). Multimedia CALL: Lessons to be learned from research on instructed SLA. *Language Learning and Technology*, vol.2, no.1, pp.22-34.

Grosz, B. J., A. K. Joshi and S. Weinstein. (1995). Centering: A framework for modelling the local coherence of discourse. Computational Linguistics, 21/2, pp. 203-225.

Ikehara, S., M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura and Y. Hayashi (1997). *Goi-Taikei – A Japanese Lexicon*, 5 volumes, Iwanami Shoten, Tokyo.

Krashen, S. (1982). *Principles and Practice in Second Language Acquisition*. Pergamon, Oxford.

NAIST, Kudo, K. (2001). *CaboCha* 0.21. http://cl.aist-nara.ac.jp/~taku-ku/software/cabocha/

NAIST, Matsumoto, Y. *et al*. (2001). *ChaSen* 2.2.8. http://chasen.aist-nara.ac.jp/

Ishiwata, T. (1999). *Gendai GengoRiron to Kaku*, Hituzi Shobo, Tokyo.

Norris, J. M. and L. Ortega (2000). Effectiveness of L2 instruction: A research synthesis and quantitative meta-analysis. *Language Learning* 50 (3), pp.417-528.

Somers, H. L. (1984). On the validity of the complement-adjunct distinction in valency grammar. *Linguistics* 22, pp. 507-53.