A Second Language Acquisition Model Using Example Generalization and Concept Categories

Ari Rappoport
Institute of Computer Science
The Hebrew University
Jerusalem, Israel
arir@cs.huji.ac.il

Vera Sheinman
Institute of Computer Science
The Hebrew University
Jerusalem, Israel
vera46@cl.cs.titech.ac.jp

Abstract

We present a computational model of acquiring a second language from example sentences. Our learning algorithms build a construction grammar language model, and generalize using form-based patterns and the learner's conceptual system. We use a unique professional language learning corpus, and show that substantial reliable learning can be achieved even though the corpus is very small. The model is applied to assisting the authoring of Japanese language learning corpora.

1 Introduction

Second Language Acquisition (SLA) is a central topic in many of the fields of activity related to human languages. SLA is studied in cognitive science and theoretical linguistics in order to gain a better understanding of our general cognitive abilities and of first language acquisition (FLA)[1]. Governments, enterprises and individuals invest heavily in foreign language learning due to business, cultural, and leisure time considerations. SLA is thus vital for both theory and practice and should be seriously examined in computational linguistics (CL), especially when considering the close relationship to FLA and the growing attention devoted to the latter by the CL community.

In this paper we present a computational model of SLA. As far as we could determine, this is the first model that simulates the learning process computationally. Learning is done from examples, with no reliance on explicit rules. The model is unique in the usage of a conceptual system by the learning algorithms. We use a unique professional language learning corpus, showing effective learning from a very small number of examples. We evaluate the model by applying it to assisting the authoring of Japanese language learning corpora.

We focus here on basic linguistic aspects of SLA, leaving other aspects to future papers. In particular, we assume that the learner possesses perfect memory and is capable of invoking the provided learning algorithms without errors.

In sections 2 and 3 we provide relevant background and discuss previous work. Our input, learner and language models are presented in section 4, and the learning algorithms in section 5. Section 6 discusses the authoring application.

2 Background

We use the term 'second language acquisition' to refer to any situation in which adults learn a new language[2]. A major concept in SLA theory [Gass01, Mitchell03] is that of **interlanguage**: when learning a new language (L2), at any given point in time the learner has a valid partial L2 language system that differs from his/her native language(s) (L1) and from the L2. The SLA process is that of progressive enhancement and refinement of interlanguage. The main trigger for interlanguage modification is when the learner notices a **gap** between interlanguage and L2 forms. In order for this to happen, the learner must be provided with **com-**

[1] Note that the F stands here for 'First', not 'Foreign'.

[2] Some SLA texts distinguish between 'second' and 'foreign' and between 'acquisition' and 'learning'. We will not make those distinctions here.

Proceedings of the Second Workshop on Psychocomputational Models of Human Language Acquisition, pages 45–52,
Ann Arbor, June 2005.

prehensible input. Our model directly supports all of these notions.

A central, debated issue in language acquisition is whether FLA mechanisms [Clark03] are available in SLA. What is clear is that SL learners already possess a mature conceptual system and are capable of explicit symbolic reasoning and abstraction. In addition, the amount of input and time available for FLA are usually orders of magnitude larger than those for SLA.

The general linguistic framework that we utilize in this paper is that of **Construction Grammar (CG)** [Goldberg95, Croft01], in which the building blocks of language are words, phrases and phrase templates that carry meanings. [Tomasello03] presents a CG theory of FLA in which children learn whole constructions as 'islands' that are gradually generalized and merged. Our SLA model is quite similar to this process.

In language education, current classroom methods use a combination of formal rules and communicative situations. Radically different is the Pimsleur method [Pimsleur05], an audio-based self-study method in which rules and explanations are kept to a minimum and most learning occurs by letting the learner infer L2 constructs from translations of contextual L1 sentences. Substantial anecdotal evidence (as manifested by learner comments and our own experience) suggests that the method is highly effective. We have used a Pimsleur corpus in our experiments. One of the goals of our model is to assist the authoring of such corpora.

3 Previous Work

There is almost no previous CL work explicitly addressing SLA. The only one of which we are aware is [Maritxalar97], which represents interlanguage levels using manually defined symbolic rules. No language model (in the CL sense) or automatic learning are provided.

Many aspects of SLA are similar to first language acquisition. Unsupervised grammar induction from corpora is a growing CL research area ([Clark01, Klein05] and references there), mostly using statistical learning of model parameters or pattern identification by distributional criteria. The resulting models are not easily presentable to humans, and do not utilize semantics.

[Edelman04] presents an elegant FLA system in which constructions and word categories are identified iteratively using a graph. [Chang04] presents an FLA system that truly supports construction grammar and is unique in its incorporation of general cognitive concepts and embodied semantics.

SLA is related to machine translation (MT), since learning how to translate is a kind of acquisition of the L2. Most relevant to us here is modern example-based machine translation (EBMT) [Somers01, Carl03], due to its explicit computation of translation templates and to the naturalness of learning from a small number of examples [Brown00, Cicekli01].

The Computer Assisted Language Learning (CALL) literature [Levy97, Chapelle01] is rich in project descriptions, and there are several commercial CALL software applications. In general, CALL applications focus on teacher, environment, memory and automatization aspects, and are thus complementary to the goals that we address here.

4 Input, Learner and Language Knowledge Models

Our ultimate goal is a comprehensive computational model of SLA that covers all aspects of the phenomenon. The present paper is a first step in that direction. Our goals here are to:

- Explore what can be learned from **example-based, small, beginner-level input corpora tailored** for SLA;
- Model a learner having a mature **conceptual system**;
- Use an L2 **language knowledge** model that supports sentence enumeration;
- Identify cognitively plausible and effective SL **learning algorithms**;
- Apply the model in **assisting the authoring** of corpora tailored for SLA.

In this section we present the first three components; the learning algorithms and the application are presented in the next two sections.

4.1 Input Model

The input potentially available for SL learners is of high variability, consisting of meta-linguistic rules, usage examples isolated for learning purposes, usage examples partially or fully understood in context, dictionary-like word definitions, free-form explanations, and more.

One of our major goals is to explore the relationship between first and second language acquisition. Methodologically, it therefore makes sense to first study input that is the most similar linguistically to that available during FLA, usage examples. As noted in section 2, a fundamental property of SLA is that learners are capable of mature understanding. Input in our model will thus consist of an ordered set of **comprehensible usage examples**, where an example is a pair of L1, L2 sentences such that the former is a translation of the latter in a certain understood context.

We focus here on modeling **beginner-level proficiency**, which is qualitatively different from native-like fluency [Gass01] and should be studied before the latter.

We are interested in **relatively small** input corpora (thousands of examples at most), because this is an essential part of SLA modeling. In addition, it is of great importance, in both theoretical and computational linguistics, to explore the limits of what can be learned from meager input.

One of the main goals of SLA modeling is to discover which input is most effective for SLA, because a substantial part of learners' input can be controlled, while their time capacity is small. We thus allow our input to be **optimized** for SLA, by containing examples that are sub-parts of other examples and whose sole purpose is to facilitate learning those (our corpus is also optimized in the sense of covering simpler constructs and words first, but this issue is orthogonal to our model). We utilize two types of such sub-examples. First, we require that new words are always presented first on their own. This is easy to achieve in controlled teaching, and is actually very frequent in FLA as well [Clark03]. In the present paper we will assume that this completely solves the task of segmenting a sentence into words, which is reasonable for a beginner level corpus where the total number of words is relatively small. Word boundaries are thus explicitly and consistently marked.

Second, the sub-example mechanism is also useful when learning a construction. For example, if the L2 sentence is 'the boy went to school' (where the L2 here is English), it could help learning algorithms if it were preceded by 'to school' or 'the boy'. Hence we do not require examples to be complete sentences.

In this paper we do not deal with phonetics or writing systems, assuming L2 speech has been consistently transcribed using a quasi-phonetic writing system. Learning L2 phonemes is certainly an important task in SLA, but most linguistic and cognitive theories view it as separable from the rest of language acquisition [Fromkin02, Medin05].

The input corpus we have used is a transcribed Pimsleur Japanese course, which fits the input specification above.

4.2 Learner Model

A major aspect of SLA is that learners already possess a mature conceptual system (CS), influenced by their life experience (including languages they know). Our learning algorithms utilize a CS model. We opted for being conservative: the model is only allowed to contain concepts that are clearly possessed by the learner before learning starts. Concepts that are particular to the L2 (e.g., 'noun gender' for English speakers learning Spanish) are not allowed. Examples for concept classes include fruits, colors, human-made objects, physical activities and emotions, as well as meta-linguistic concepts such as pronouns and prepositions. A single concept is simply represented by a prototypical English word denoting it (e.g., 'child', 'school'). A concept class is represented by the concepts it contains and is conveniently named using an English word or phrase (e.g., 'types of people', 'buildings', 'language names').

Our learners can explicitly reason about concept inter-relationships. Is-a relationships between classes are represented when they are beyond any doubt (e.g., 'buildings' and 'people' are both 'physical things').

A basic conceptual system is assumed to exist before the SLA process starts. When the input is controlled and small, as in our case, it is both methodologically valid and practical to prepare the CS manually. CS design is discussed in detail in section 6.

In the model described in the present paper we do not automatically modify the CS during the learning process; CS evolution will be addressed in future models.

As stated in section 1, in this paper we focus on linguistic SLA aspects and do not address issues such as human errors, motivation and attention. We thus assume that our learner possesses perfect memory and can invoke our learning algorithms without any mistakes.

4.3 Language Knowledge Model

We require our model to support a basic capability of a grammar: enumeration of language sentences (parsing will be reported in other papers). In addition, we provide a degree of certainty for each. The model's quality is evaluated by its applicability for learning corpora authoring assistance (section 6).

The representation is based on construction grammar (CG), explicitly storing a set of constructions and their inter-relationships. CG is ideally suited for SLA interlanguage because it enables the representation of partial knowledge: every language form, from concrete words and sentences to the most abstract constructs, counts as a construction. The generative capacity of language is obtained by allowing constructions to replace arguments. For example, (child), (the child goes to school), (<x> goes to school), (<x> <v> to school) and (X goes Z) are all constructions, where <x>, <v> denote word classes and X, Z denote other constructions.

SL learners can make explicit judgments as to their level of confidence in the grammaticality of utterances. To model this, our learning algorithms assign a **degree of certainty (DOC)** to each construction and to the possibility of it being an argument of another construction. The certainty of a sentence is a function (e.g., sum or maximum) of the DOCs present in its derivation path.

Our representation is equivalent to a graph whose nodes are constructions and whose directed, labeled arcs denote the possibility of a node filling a particular argument of another node. When the graph is a-cyclic the resulting language contains a finite number of concrete sentences, easily computed by graph traversal. This is similar to [Edelman04]; we differ in our partial support for semantics through a conceptual system (section 5) and in the notion of a degree of certainty.

5 Learning Algorithms

Our general SLA scheme is that of incremental learning – examples are given one by one, each causing an update to the model. A major goal of our model is to identify effective, cognitively plausible learning algorithms. In this section we present a concrete set of such algorithms.

Structured categorization is a major driving force in perception and other cognitive processes [Medin05]. Our learners are thus driven by the desire to form useful generalizations over the input. A generalization of two or more examples is possible when there is sufficient similarity of form and meaning between them. Hence, the basic ingredient of our learning algorithms is identifying such similarities.

To identify concrete effective learning algorithms, we have followed our own inference processes when learning a foreign language from an example-based corpus (section 6). The set of algorithms described below are the result of this study.

The basic form similarity algorithm is **Single Word Difference (SWD)**. When two examples share all but a single word, a construction is formed in which that word is replaced by an argument class containing those words. For example, given 'eigo ga wakari mas' and 'nihongo ga wakari mas', the construction (<eigo, nihongo> ga wakari mas) ('I understand English/Japanese'), containing one argument class, is created. In itself, SWD only compresses the input, so its degree of certainty is maximal. It does not create new sentences, but it organizes knowledge in a form suitable for generalization.

The basic meaning-based similarity algorithm is **Extension by Conceptual Categories (ECC)**. For an argument class W in a construction C, ECC attempts to find the smallest concept category U' that contains W', the set of concepts corresponding to the words in W. If no such U' exists, C is removed from the model. If U' was found, W is replaced by U, which contains the L2 words corresponding to the concepts in U'. When the replacement occurs, it is possible that not all such words have already been taught; when a new word is taught, we add it to all such classes U (easily implemented using the new word's translation, which is given when it is introduced.)

In the above example, the words in W are 'eigo' and 'nihongo', with corresponding concepts 'English' and 'Japanese'. Both are contained in W', the 'language names' category, so in this case U' equals W'. The language names category contains concepts for many other language names, including Korean, so it suffices to teach our learner the Japanese word for Korean ('kankokugo') at some point in the future in order to update the construction to be (<eigo, nihongo, kankokugo> ga wakari mas). This creates a new sentence 'kankokugo ga wakari mas' meaning 'I understand Korean'. An

example in which U' does not equal W' is given in Table 1 by 'child' and 'car'.

L2 words might be ambiguous – several concepts might correspond to a single word. Because example semantics are not explicitly represented, our system has no way of knowing which concept is the correct one for a given construction, so it considers all possibilities. For example, the Japanese 'ni' means both 'two' and 'at/in', so when attempting to generalize a construction in which 'ni' appears in an argument class, ECC would consider both the 'numbers' and 'prepositions' concepts.

The degree of certainty assigned to the new construction by ECC is a function of the quality of the match between W and U'. The more abstract is U, the lower the certainty.

The main form-based induction algorithm is **Shared Prefix, Generated Suffix (SPGS)**. Given an example 'x y' (x, y are word sequences), if there exist (1) an example of the form 'x z', (2) an example 'x', and (3) a construction K that derives 'z' or 'y', we create the construction (x K) having a degree of certainty lower than that of K. A Shared Suffix version can be defined similarly. Requirement (2) ensures that the cut after the prefix will not be arbitrary, and assumes that the lesson author presents constituents as partial examples beforehand (as indeed is the case in our corpus).

SPGS utilizes the learner's current generative capacity. Assume input 'watashi wa biru o nomi mas' ('I drink beer'), previous inputs 'watashi wa america jin des' ('I am American'), 'watashi wa' ('as to me...') and an existing construction K = (<biru, wain> o nomi mas). SPGS would create the construction (watashi wa K), yielding the new sentence 'watashi wa wain o nomi mas' ('I drink wine').

To enable faster learning of more abstract constructions, we use generalized versions of SWD and SPGS, which allow the differing or shared elements to be a *construction* rather than a word or a word sequence.

The combined learning algorithm is: given a new example, iteratively invoke each of the above algorithms at the given order until nothing new can be learned. Our system is thus a kind of inductive programming system (see [Thompson99] for a system using inductive logic programming for semantic parsing).

Note that the above algorithms treat words as atomic units, so they can only learn morphological rules if boundaries between morphemes are marked in the corpus. They are thus more useful for languages such as Japanese than, say, for Romance or Semitic languages.

Our algorithms have been motivated by general cognitive considerations. It is possible to refine them even further, e.g. by assigning a higher certainty when the focus element is a prefix or a suffix, which are more conspicuous cognitively.

6 Results and Application to Authoring of Learning Corpora

We have experimented with our model using the Pimsleur Japanese I (for English speakers) course, which comprises 30 half-hour lessons, 1823 different examples, and about 350 words. We developed a simple set of tools to assist transcription, using an arbitrary, consistent Latin script transliteration based on how the Japanese phonemes are presented in the course, which differs at places from common transliterations (e.g., we use 'mas', not 'masu'). Word boundaries were marked during transliteration, as justified in section 4.

Example sentences from the corpus are 'nani o shi mas kaa ? / what are you going to do?', 'watashi ta chi wa koko ni i mas / we are here', 'kyo wa kaeri masen / today I am not going back', 'demo hitori de kaeri mas / but I am going to return alone', etc. Sentences are relatively short and appropriate for a beginner level learner.

Evaluating the quality of induced language models is notoriously difficult. Current FLA practice favors comparison of predicted parses with ones in human annotated corpora. We have focused on another basic task of a grammar, sentence enumeration, with the goal of showing that our model is useful for a real application, assistance for authoring of learning corpora.

The algorithm has learned 113 constructions from the 1823 examples, generating 525 new sentences. These numbers do not include constructions that are subsumed by more abstract ones (generating a superset of their sentences) or those involving number words, which would distort the count upwards. The number of *potential* new sentences is much higher: these numbers are based only on the 350 words present, organized in a rather flat CS. The constructions contain many

placeholders for concepts whose words would be taught in the future, which could increase the number exponentially.

In terms of precision, 514 of the 525 sentences were judged (by humans) to be syntactically correct (53 of those were problematic semantically). Regarding recall, it is very difficult to assess formally. Our subjective impression is that the learned constructions do cover most of what a reasonable person would learn from the examples, but this is not highly informative – as indicated, the algorithms were discovered by following our own inherence processes. In any case, our algorithms have been deliberately designed to be conservative to ensure precision, which we consider more important than recall for our model and application.

There is no available standard benchmark to serve as a baseline, so we used a simpler version of our own system as a baseline. We modified ECC to not remove C in case of failure of concept match (see ECC's definition in section 5). The number of constructions generated after seeing 1300 examples is 3,954 (yielding 35,429 sentences), almost all of which are incorrect.

The applicative scenario we have in mind is the following. The corpus author initially specifies the desired target vocabulary and the desired syntactical constructs, by writing examples (the easiest interface for humans). Vocabulary is selected according to linguistic or subject (e.g., tourism, sports) considerations. The examples are fed one by one into the model (see Table 1). For a single word example, its corresponding concepts are first manually added to the CS.

The system now lists the constructions learned. For a beginner level and the highest degree of certainty, the sentences licensed by the model can be easily grasped just by looking at the constructions. The fact that our model's representations can be easily communicated to people is also an advantage from an SLA theory point of view, where 'focus on form' is a major topic [Gass01]. For advanced levels or lower certainties, viewing the sentences themselves (or a sample, when their number gets too large) might be necessary.

The author can now check the learned items for errors. There are two basic error types, errors stemming from model deficiencies and errors that human learners would make too. As an example of the former, wrong generalizations may result from discrepancies between the modeled conceptual system and that of a real person. In this case the author fixes the modeled CS. Discovering errors of the second kind is exactly the point where the model is useful. To address those, the author usually introduces new full or partial examples that would enable the learner to induce correct syntax. In extreme cases there is no other practical choice but to provide explicit linguistic explanations in order to clarify examples that are very far from the learner's current knowledge. For example, English speakers might be confused by the variability of the Japanese counting system, so it might be useful to insert an explanation of the sort 'X is usually used when counting long and thin objects, but be aware that there are exceptions'. In the scenario of Table 1, the author might eventually notice that the learner is not aware that when speaking of somebody else's child a more polite reference is in order, which can be fixed by giving examples followed by an explanation. The DOC can be used to draw the author's attention to potential problems.

Preparation of the CS is a sensitive issue in our model, because it is done manually while it is not clear at all what kind of CS people have (WordNet is sometimes criticized for being arbitrary, too fine, and omitting concepts). We were highly conservative in that only concepts that are clearly part of the conceptual system of English speakers before any exposure to Japanese were included. Our task is made easier by the fact that it is guided by words actually appearing in the corpus, whose number is not large, so that it took only about one hour to produce a reasonable CS. Example categories are names (for languages, places and people), places (park, station, toilet, hotel, restaurant, shop, etc), people (person, friend, wife, husband, girl, boy), food, drink, feelings towards something (like, need, want), self motion activities (arrive, come, return), judgments of size, numbers, etc. We also included language-related categories such as pronouns and prepositions.

7 Discussion

We have presented a computational model of second language acquisition. SLA is a central subject in linguistics theory and practice, and our main contribution is in addressing it in computational linguistics. The model's learning algorithms are unique in their usage of a conceptual system, and

its generative capacity is unique in its support for degrees of certainty. The model was tested on a unique corpus.

The dominant trend in CL in the last years has been the usage of ever growing corpora. We have shown that meaningful learning can be achieved from a small corpus when the corpus has been prepared by a 'good teacher'. Automatic identification (and ordering) of corpora subsets from which learning is effective should be a fruitful research direction for CL.

We have shown that using a simple conceptual system can greatly assist language learning algorithms. Previous FLA algorithms have in effect computed a CS simultaneously with the syntax; decoupling the two stages could be a promising direction for FLA.

The model presented here is the first computational SLA model and obviously needs to be extended to address more SLA phenomena. It is clear that the powerful notion of certainty is only used in a rudimentary manner. Future research should also address constraints (e.g. for morphology and agreement), recursion, explicit semantics (e.g. parsing into a semantic representation), word segmentation, statistics (e.g. collocations), and induction of new concept categories that result from the learned language itself (e.g. the Japanese counting system).

An especially important SLA issue is L1 transfer, which refers to the effect that the L1 has on the learning process. In this paper the only usage of the L1 part of the examples was for accessing a conceptual system. Using the L1 sentences (and the existing conceptual system) to address transfer is an interesting direction for research, in addition to using the L1 sentences for modeling sentence semantics.

Many additional important SLA issues will be addressed in future research, including memory, errors, attention, noticing, explicit learning, and motivation. We also plan additional applications, such as automatic lesson generation.

Acknowledgement. We would like to thank Dan Melamed for his comments on a related document.

References

Brown Ralf, 2000, Automated Generalization of Translation Examples, COLING '00.

Carl Michael, Way Andy, (eds), 2003, Recent Advances in Example Based Machine Translation, Kluwer.

Chang Nancy, Gurevich Olya, 2004. Context-Driven Construction Learning. Proceedings, Cognitive Science '04.

Chapelle Carol, 2001. Computer Applications in SLA. Cambridge University Press. .

Cicekli Ilyas, Gu"venir Altay, 2001, Learning Translation Templates from Bilingual Translational Examples. Applied Intelligence 15:57-76, 2001.

Clark Alexander, 2001. Unsupervised Language Acquisition: Theory and Practice. PhD thesis, University of Sussex.

Clark Eve Vivienne, 2003. First Language Acquisition. Cambridge University Press.

Croft, William, 2001. Radical Construction Grammar. Oxford University Press.

Edelman Shimon, Solan Zach, Horn David, Ruppin Eytan, 2004. Bridging Computational, Formal and Psycholinguistic Approaches to Language. Proceedings, Cognitive Science '04.

Fromkin Victoria, Rodman Robert, Hyams Nina, 2002. An Introduction to Language, 7th ed. Harcourt.

Gass Susan M, Selinker Larry, 2001. Second Language Acquisition: an Introductory Course. 2nd ed. LEA Publishing.

Goldberg Adele, 1995. Constructions: a Construction Grammar Approach to Argument Structure. Chicago University Press.

Klein Dan, 2005. The Unsupervised Learning of Natural Language Structure. PhD Thesis, Stanford.

Levy Michael, 1997. Computer-Assisted Language Learning. Cambridge University Press.

Maritxalar Montse, Diaz de Ilarraza Arantza, Oronoz Maite, 1997. From Psycholinguistic Modelling of Interlanguage in SLA to a Computational Model. CoNLL '97.

Medin Douglas, Ross Brian, Markman Arthur, 2005. Cognitive Psychology, 4th ed. John Wiley & Sons.

Mitchell Rosamond, Myles Florence, 2003. Second Language Learning Theories. 2nd ed. Arnold Publication.

Pimsleur 2005. www.simonsays.com, under 'foreign language instruction'.

Somers Harold, 2001. Example-based Machine Translation. Machine Translation 14:113-158.

Thompson Cynthia, Califf Mary Elaine, Mooney Raymond, 1999. Active Learning for Natural Language Parsing and Information Extraction. ICML '99.

Tomasello Michael, 2003. Constructing a Language: a Usage Based Theory of Language Acquisition. Harvard University Press.

	Construction	DOC	Source	Comment
1	anata / you	0	example	
2	watashi / I	0	example	
3	anata no / your	0	example	
4	watashi no / my	0	example	
5	(<anata,watashi> no)	0	SWD(3,4)	The first words of 3 and 4 are different, the rest is identical.
6	(W no), where W is <anata, watashi, Japanese word for 'we'>	-1	ECC(5)	The concept category W'={I, you, we} was found in the CS. We know how to say 'I' and 'you', but not 'we'.
7	watashi ta chi / we	0	example	
8	(W no), where W is <anata, watashi, watashi ta chi>	-2	ECC(6,7)	We were taught how to say 'we', and an empty slot for it was found in 6.
				Now we can generate a new sentence: 'watashi ta chi no', whose meaning ('our') is inferred from the meaning of construction 6.
9	chiisai / small	0	example	
10	kuruma / car	0	example	
11	chiisai kuruma / a small car	0	example	
12	watashi ta chi no kuruma / our car	0	example	
13	((W no) kuruma)	-3	SSGP (12, 11, 10, 8)	Shared Suffix Generated Prefix: (0) new example 12 = 'y x' (x: kuruma) (1) existing example 11 = 'z x' (2) existing example 10 = 'x' (3) construction K (#8) deriving 'y' learns the new construction (K x)
				Now we can generate a new sentence: 'watashi no kuruma', meaning 'my car'.
14	kodomo / child	0	example	
...	...	0	examples	Skipping a few examples...
20	((W no) kodomo)	-3	...	This construction was learned using the skipped examples.
21	((W no) <kuruma, kodomo>)	-3	SWD (13, 20)	Note that the shared element is a construction this time, not a sub-sentence.
22	((W no) P), where P is the set of Japanese words for physical things (animate or inanimate)	-4	ECC (21)	The smallest category that contains the concepts 'car' and 'child' is P'=PhysicalThings.
				Now we can generate many new sentences, meaning 'my X' where X is any Japanese word we will learn in the future denoting a physical thing.

Table 1: A learning scenario. For simplicity, the degree of certainty here is computed by adding that of the algorithm type to that of the most uncertain construction used. Note that the notation used was designed for succinct presentation and is not the optimal one for authors of learning corpora (for example, it is probably easier to visualize the sentences generated by construction #22 if it were shown as ((<watashi, anata, watashi ta chi> no) <kuruma, kodomo>).)