COLING 2002 Post-Conference Workshop

Asian Language Resources and International Standardization

Proceedings of the Workshop

31 August 2002
Center of Academia Activities, Academia Sinica
Taipei, Taiwan

Preface

This volume contains the papers presented at the workshop on Asian Language Resources and International Standardization, held on 31 August 2002 in conjunction with the 19th International Conference on Computational Linguistics (COLING 2002).

Language resources play an essential role in empirical approaches to natural language processing (NLP). Previous concerted efforts on construction of language resources, particularly in the US and European countries, have laid a solid foundation for the pioneering NLP researches in these two communities over the last decade. In comparison, availability and accessibility of Asian language resources is still very limited. Moreover, there is more diversity of Asian languages from viewpoints of character sets and grammatical properties. Because of these peculiarities, Asian languages do not always fit with the existing linguistic resource standardization frameworks.

We have held two workshops on the same topic, the first was in January of 2001 at Tokyo on invited basis and the second was in conjunction with the 6th Natural Language Processing Pacific Rim Symposium (NLPRS 2001) in November of 2001 at Tokyo. In this third workshop, we would like to put emphasis on standardization of Asian language resources, and to provide a chance to discuss research results and the possibilities of international collaboration on the development of Asian language resources in the future. The workshop also aims to introduce the status of Asian language resources to researchers in other regions. We hope this workshop can achieve these goals.

Nicoletta Calzolari
Key-Sun Choi
Asanee Kawtrakul
Alessandro Lenci
Tokunaga Takenobu
Program Co-chairs

Program Committee

Nicoletta Calzolari	*Istituto di Linguistica Computazionale CNR (Italy)*
Key-Sun Choi	*Korea Advanced Institute of Science and Technology (Korea)*
Asanee Kawtrakul	*Kasetsart University (Thailand)*
Alessandro Lenci	*Dipartimento di Linguistica – Universita di Pisa (Italy)*
Tokunaga Takenobu	*Tokyo Institute of Technology (Japan)*
Steven Bird	*University of Melbourne (Australia)*
Nuria Bel	*GILCUB (Spain)*
Ehara Terumasa	*NHK (Japan)*
Christiane Fellbaum	*Princeton University (USA)*
Ralph Grishman	*New York University (USA)*
Chu-Ren Huang	*Academia Sinica (Taiwan)*
Hammam Riza	*BPPT (Indonesia)*
Kurohashi Sadao	*University of Tokyo (Japan)*
Martha Palmer	*University of Pennsylvania (USA)*
Hae-Chang Rim	*Korea University (Korea)*
Rajeev Sangal	*International Institute of Information Technology Hyderabad (India)*
Shirai Kiyoaki	*Japan Advanced Institute of Science and Tecchnology (Japan)*
Virach Sornlertlamvanich	*NECTEC (Thailand)*
Gregor Thurmair	*SAIL Labs (Germany)*
Benjamin Tsou	*City University of HongKong (China)*
Antonnio Zampolli	*Istituto di Linguistica Computazionale CNR (Italy)*

Workshop Program

31 August 2002
Center of Academia Activities, Academia Sinica
Taipei, Taiwan

8:30-9:00	*Registration*
9:00-9:30	*A State of the Art of Thai Language Resources and Thai Behavior Analysis and Modeling* Asanee Kawtrakul, Mukda Suktarachan, Patcharee Varasai, Hutchatai Chanlekha
9:30-10:00	*Broadening the Scope of the EAGLES/ISLE Lexical Standardization Initiative* Nicoletta Calzolari, Alessandro Lenci, Francesca Bertagna, Antonio Zampolli
10:00-10:30	*Lexicon-based Orthographic Disambiguation in CJK Intelligent Information Retrieval* Jack Halpern
10:30-11:00	⟨**Break**⟩
11:00-11:30	*Decomposition for ISO/IEC 10646 Ideographic Characters* Lu Qin, Chan Shiu Tong, Li Yin, Li Ngai Ling
11:30-12:00	*Efficient Deep Processing of Japanese* Melanie Siegel, Emily M. Bender
12:00-12:30	*Urdu and the Parallel Grammar Project* Miriam Butt, Tracy Holloway King
12:30-13:30	⟨**Lunch**⟩
13:30-14:00	*A Study in Urdu Corpus Construction* Dara Becker, Kashif Riaz
14:00-14:30	*Automatic Word Spacing Using Hidden Markov Model for Refining Korean Text Corpora* Do-Gil Lee, Sang-Zoo Lee, Hae-Chang Rim, and Heui-Seok Lim
14:30-15:00	*Constructing of a Large-Scale Chinese-English Parallel Corpus* Le Sun, Weimin Qu, Song Xue, Xiaofeng Wang,Yufang Sun
15:00-15:30	*AnnCorra: Building Tree-banks across Indian Languages* Rajeev Sangal, Vineet Chaitanya, Amba Kulkarni, Dipti Misra Sharma
15:30-16:00	⟨**Break**⟩
16:00-16:30	*OLACMS: Comparisons and Applications in Chinese and Formosan Languages* Ru-Yng Chang, Chu-Ren Huang
16:30-17:30	*Discussion*

Table of Contents

A State of the Art of Thai Language Resources
and
Thai Language Behavior Analysis and Modeling

Asanee Kawtrakul, Mukda Suktarachan, Patcharee Varasai,
Hutchatai Chanlekha
Department of Computer Engineering,
Faculty of Engineering, Kasetsart University, Bangkok, Thailand 10900.
E-mail: ak, mukda, pom, aim@vivaldi.cpe.ku.ac.th,

Abstract

As electronic communications is now increasing, the term Natural Language Processing should be considered in the broader aspect of Multi-Language processing system. Observation of the language behavior will provide a good basis for design of computational language model and also creating cost-effective solutions to the practical problems. In order to have a good language modeling, the language resources are necessary for the language behavior analysis.

This paper intended to express what we have and what we have done by the desire to make a bridge between the languages and to share and make maximal use of the existing lexica, corpus and the tools. Three main topics are, then, focussed: A State of the Art of Thai language Resources, Thai language behaviors and their computational models.

1. Introduction

As electronic communications are now increasing, the term Natural Language Processing should be considered in the broader aspect of Multi- Language Processing system. An important phase in the system development process is requirement engineering, which can define as the process of analyzing the problems in a certain language. An essential part of the requirement-engineering phase is computational language modeling which is an abstract representation of the behavior of the language. In order to have a good language model for creating cost-effective solutions to the practical problems, the language resources are necessary for the language behavior analysis.

This paper intended to express what we have and what we have done by the desire to make a bridge between the languages and to share and make maximal use of the existing lexica, corpus and the tools. Three main topics are, then, focussed:

- A State of the Art of Thai language Resources that will give an overview of what we have in Corpus, Lexicon and tools for corpus processing and analysis.
- Thai language behaviors (only in word and phrase level) analyzed from the varieties of corpus which consist of Lexicon growth, New word formation and Phrase/Sentence construction, and
- The computational models providing for those behaviors, which consist of Unknown Word Extraction and Name Entities identification, New word generation and Noun phrase recognition.

The remainder of the paper is organized as follows. In section 2, we give the gateway of Thai language resources. Thai Language behaviors are discussed in section 3. In section 4, then, provides Thai Language Computational Modeling as a basis for creating cost-effective solutions to those practical problems.

2. A State of the Art of Thai Language Resource

This section gives a survey of a state of the art of Thai Language Resources consisting of Corpus, Lexicon and Tools. Here, we will present only the resources that open for public access.

2.1 Corpus

The existing Thai corpus is divided into 2 types; speech and text corpus developed by many Thai Universities. Thai Language Audio Resource Center of Thammasart University (ThaiARC) (http:// thaiarc.ac.th) developed speech corpus aimed to provide digitized audio information for dissemination via Internet. The project pioneers the production and collection of various types of audio information and various styles of Thai speech, such as royal speeches, academic lectures, oral literature, etc.

For Text corpus, originally, the goal of the corpus collecting is used only inside the laboratory. Until 1996, National Electronics and Computer Technology Center (NECTEC) and Communications Research Laboratory (CRL) had a collaboration project with the purpose of preparing Thai language corpus from technical proceedings for language study and application research. It named ORCHID corpus (NECTEC, 1997). NAiST Corpus began in 1996 with the primary aim of collecting document from magazines for training and testing program in Written Production Assistance (Asanee, 1995). The existing corpus can be summarized as shown in Table 1.

Table 1: The List of Thai Corpus

List	Corpus	Type	Amount	Status
NECTEC	Orchid Corpus	POS-Tagged Text	2,560,000 words	Online
Kasetsart Univ.	NAiST Corpus	Text	60,511,974 words	Online
Thammasart Univ.	Thai ARC	Digitized audio	4000 words++	online

2.2 Lexicon

There are a number of Thai lexicons, which has been developed as shown in Table 2.

Table 2: The List of Thai Dictionaries

Dictionary	Type	Size (word)	status	Web site
Royal Institute Dictionary	Mono	33,582	Online	http://rirs3.royin.go.th/riThdict/lookup.html
Lexitron	Bi	50,000	Online	http://www.links.nectec.or.th/lexit/lex_t.html
NaiST Lexibase	Mono	15,000	Online	http://beethoven.cpe.ku.ac.th/
So Sethaputra Dictionary	Bi	- 48,000 Eng words - 38,000 Thai words	Online	http://www.thaisoftware.co.th/
Narin's Thailand homepage	Bi	-	Online	http://www.wiwi.uni-frankfurt.de/~sascha/thailand/dictionary/dictionary_index.html
Saikam online	Bi	133,524	Online	http://saikam.nii.ac.jp/.
Lao-Thai-English Dic.	Multi	5,000	Offline	-

From the table 2, Only Lexitron (from NECTEC) and NAiST Lexibase (from Kasetsart University) that were applied to NLP. NAiST Lexibase has been developed based on relational model for managing and maintaining easily in the future. It contains 15,000 words list with their syntax and the semantic concept information in the concept code.

2.3 Corpus and Language Analysis Tools

Corpus is not only the resource of Linguistic Knowledge but is used for training, improving and evaluating the NLP systems. The tools for corpus manipulation and knowledge acquisition become necessary.

NAiST Lab. has developed the toolkit for sharing via the Internet. It has been designed for corpus collecting, annotating, maintaining and analyzing. Additionally, it has been designed as the engine, which the end user could use with their data. (See a service on **http://naist.cpe.ku.ac.th**).

3. Thai Language Behavior Analysis

In order to have a good language model for creating cost-effective solutions to the practical problems in application development, language behavior must be observed. Next is Thai language behavior analysis based on NAiST corpus consisting of Lexicon growth, Thai word formation and Phrase construction.

3.1 Lexicon Growth

The lexicon growth is studied by using Word list Extraction tool to extract word lists from a large-scale corpus and mapping to the Royal Institute Dictionary (RID). It is noticeable that there are two types of lexicon: common and unknown words. The common word lists are some words in RID, which occur in almost every document, and use in daily life. They are primitive words but not being proper names or colloquial words. The unknown or new words occur much in the real document such as Proper names, Colloquial words, Abbreviations, and Foreign words.

The lexicon growth is observed from corpus size, 400,000, 2,154,700 and 60,511,974 words from Newspaper, Magazine and Agriculture text. We found that common word lists increased from 111,954 to 839,522 and 49,136,408 words according to the corpus size, while the unknown word lists increased from 288,046 to 1,315,178 and 11,375,566 words respectively as shown in table3.

Table 3 : Lexicon-growth

Size of Corpus/ words	Common words	Unknown words
400,000	111,954	288,046
2,154,700	839,522	1,315,178
60,511,974	49,136,408	11,375,566

Regarding to 60,511,974 words corpus in the table 3, it composes of 35,127,012 words from Newspaper, 18,359,724 words from Magazine and 7,025,238 words from Agricultural Text. Unknown words occur in each category as shown in table 4.

Table 4: The Categories of Unknown words according to the various corpus genres

Types of unknown word	Newspaper (words)	Magazine (words)	Agricultural Text (words)
Proper name	4,809,160	1,272,747	1,170,076
Spoken words	58,335	8,787	0
Abbreviation	70,109	43,056	0
Foreign words	304,519	239,107	3,399,670
Total	5,242,123	1,563,697	4,569,746

According to table 3 and 4, we could observe that not only unknown words increase but common words also increase and the main categories of increasing unknown word are proper names and foreign words. Consequently, a computational model of unknown word extraction and name entity identification has been developed and also of new word construction.

3.2 New Word Formation and Core Noun

Regarding to the growth of common word shown in table 3, we studied how the new words come from.

3.2.1 Basic Information about Thai

Thai words are multi-syllabic words which stringing together could form a new word. Since Thai has no inflection and no word delimiters, Thai morphological processing is mainly to recognize word boundaries instead of recognizing a lexical form from a surface form as in English.

Let C be a sequence of characters

C = c1c2c3…cn : n>=1

Let W be a sequence of words

W = w1w2w3…wm : m>=1

Where wi = c1ic2i…ci r: i>=1, r>=2

Since Thai sentences are formed with a sequence of words with a stream of characters, i.e., c1c2c3…cn mostly without explicit delimiters, the word boundary in “c1c2c3c4c5“ pattern as shown below could have two ambiguous forms. One is “c1c2“ and “c3c4c5“. The other one is “c1c2c3“ and “c4c5“ (Kawtrakul, 1997)

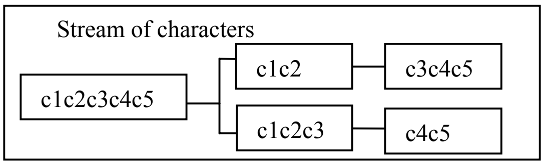


Figure 1: Word Boundary Ambiguity

From the figure 1, if characters were grouped differently, the meaning of words will be changed too. For example, “กอดอก” can be grouped to “กอด-อก(fold one's arms across the chest)” and “กอ-ดอก (a clump of flower)”. From our corpus, we found that the sentence with 45 characters has 30 combinations of words sequence.

3.2.2 New word construction

Almost all-Thai new words are formed by means of compounding and nominalization, by using a set of prefixes.

3.2.2.1 Nominalization

Nominalization is a process by which a word can be formed as a noun by using prefixes added. Noun words formed by using prefixes “การ(ka:n)” and “ความ(k^{h}wa:m)”are nouns which signal state or action. Words formed by using prefixes “ผู้(p^{h}u:)” “ชาว(tç^{h}a:w)” and “นัก(nak)”are nouns which signal human or profession.

Prefix “ การ(ka:n)” “ ความ(k^{h}wa:m)” are used in the process of forming a noun from verb or verb phrase and sometimes from noun (Nominalization). การ(ka:n) that co-occur with noun, represents the meaning about duty or function of noun it relates to. การ(ka:n) that co-occur with verbs, always occur with action verbs. ความ(k^{h}wa:m) always co-occur with state verbs.

Prefix " ผู้(p^{h}u:) " " นัก(nak)" and "ชาว(tç^{h}a:w)" are used in the process of new word formation. ผู้(p^{h}u:) and นัก(nak) co-occur with verb phrase. นัก(nak) sometimes can occur with a few fields of nouns, such as sport and music. So at the first time we kept words, which constructed from prefix "นัก(nak)" plus noun in the lexicon for solving the problem. Prefix "ชาว(tç^{h}a:w)" can co-occur with noun only.

3.2.2.2 Compounding

Thai new words can, also, be combined to form compound nouns and are invented almost daily. They normally have at least two parts. The first part represents a pointed object or person such as คน(man), หม้อ(pot), หาง(tail), พืช(plant). The second part identifies what kind of object or person it is, or what its purpose is like ขับรถ(drive a car), หุงข้าว(cook rice), เสือ(tiger), น้ำ(water). Table 5 shows the examples of compound noun in Thai.

Table 5: The Examples of Thai Compound Noun

What or who	What type / what purpose
คน(man)	ขับรถ(drive a car),
หม้อ(pot)	หุงข้าว(cook rice)
หาง(tail)	เสือ(tiger)
พืช(plant)	น้ำ(water)

Table 6 shows the patterns of compound noun.

Table 6: Compound noun pattern

Compound noun structure	Examples	Meaning
n + n	หาง(tail)เสือ(tiger) พืช(plant)น้ำ(water)	Rudder Water Plant
n + v	ห้อง(room)นอน(sleep) เก้าอี้(chair)โยก(rock)	Bedroom rocking chair
n + v + n	คน(man)ขับ(drive)รถ (car) หม้อ(pot)หุง(cook)ข้าว(rice)	Driver A Pot For Cooking Rice
n + n + v	เด็ก(child)ผม(hair)ยาว(long) คน(human)ขา(leg)เป๋(lame)	A Long Hair Child A Lame Man
n + n + n	บ้าน(home)ทรง(shape)ไทย (Thai) ข้าว(rice)ขา(leg)หมู(pig)	Thai Style House A kind of dishes
n + v + v	ใบ(leaf)ขับ(drive)ขี่(ride) ห้อง(room)นั่ง(sit)เล่น(play)	Driving License Living Room

From Table 6, it has shown that some compound nouns maintain some parts of those primitive word meaning but some changed to a new meaning. In this paper, we are only interested in compound noun grouping from primitive words which were changed the meaning to more abstract but still maintain some parts of those primitive word meanings, e.g. "คนรถ(driver) คนครัว(cooker) etc." The word "คน" maintains its meaning which has a concept of human, but when it was compounded with "รถ(car)" and "ครัว(kitchen)", their meanings have changed to the occupation by the word relation in the equivalent level. In case of compound noun that change a whole meaning such as "ลูกเสือ (a boy scout)", it will be kept in the lexicon.

3.2.2.3 Compound noun extraction problems

There are three non-trivial problems

- Compound Noun VS Sentence Distinction
- Compound Noun Boundary Ambiguity
- Core noun Detection

Compound Noun VS Sentence

Several NP structures have the same pattern as sentences. Since Thai language is flexible and has no word derivation, including to preposition in compound noun can be omitted, etc. This causes a compound noun having the same pattern as sentence. Thus, Thai NP analysis in IR system is more difficult than English. (See Figure 2)

Sentence: นกกินผลไม้ (birds eat fruit)

In Thai:	นก	กิน	ผลไม้
	Birds	eat	fruit
Syntactic Category	cn	tv	cn

Compound Noun: โต๊ะกินข้าว (a dining table)

In Thai:	โต๊ะ(สำหรับ)	กิน	ข้าว
	table	eat	rice
Syntactic Category	cn	tv	cn

Figure 2: The comparison of noun phrase and sentence structure

In figure 2, compound noun "โต๊ะกินข้าว" (a dining table) actually omit the preposition "สำหรับ (for)", which is a relation that point to the purpose of the first noun "โต๊ะ(table)".

The Compound Noun Boundary Ambiguity

After we have extracted noun phrase aiming for enhancing the IR system, we have to segment that noun phrase into sub noun phrase or compound noun in order to specify the core noun as index and its modifier as sub-index. For example, compound noun with “noun + noun + verb” structure: เด็ก(child/N)ผม(hair/N)ยาว (long/V) etc. In this case, the second noun and verb have to be grouped firstly since it behaves similarly to a modifier by omitting the relative pronoun that represents its purpose, i.e., “who has”.

Another case of Compound Noun Boundary Ambiguity is word combination. Consider the sequence of words as the example of NP that composes of four words as follows:

$$NP = N_1N_2N_3N_4$$

There are 8 word combinations of compound noun as shown in figure 3.

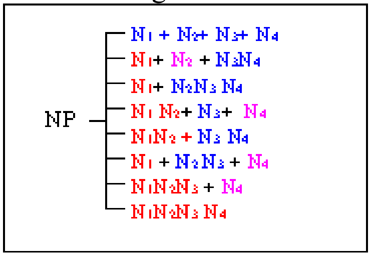


Figure 3: Patterns of noun phrase analysis

In figure 3, word string has to be grouped correctly for the correct meaning.

The ambiguity of noun phrase boundary has also directly effected the efficiency of text retrieval.

Core Noun detection

Due to the Information Retrieval, a head or core of noun phrase detection is necessary. In this paper, core noun refers to the most important and specific word that the information retrieval and extraction can directly retrieve or extract without over generating candidate words. However, by the observation, the core of noun phrase needs not to be the initial words. Some of them are at the final position and some have word relation in the equivalent level (As shown in Table 7).

Table 7: The examples of core noun in NP

Noun phrase(NP)	Core noun
W1 W2 โครงสร้าง + ประโยค structure + sentence	be W1 located at the initial position
W1 W2 รอย + วงปี stain annual ring	be W2 located at the final position
W1 W2 W3 ผล + มะละกอ + ดิบ fruit papaya green	be W2 located at the second position

As mentioned above, the models of New Word Generation and Noun Phrase Recognition become one of the interesting works in Thai processing.

3.3 Phrase and Sentence Construction

Next, we will indicate the main problems that influence to MT, IE and IR system. These are constituent movement, zero anaphora and iterative relative clause.

3.3.1 Constituent Movement

Constituent is the relationship between lexicon units, which are parts of a larger unit. Constituency is usually shown by a tree diagram or by square brackets:

Ex. [[การประชุมคณะกรรมการ] [อย่างราบรื่น]]

[[meeting committee] [very smoothly]].

Constituent acts as a chunk that can be moved together and it often occurs in Thai language (see Fig. 4). The constituents can be moved to the front, the middle or the end of the sentence.

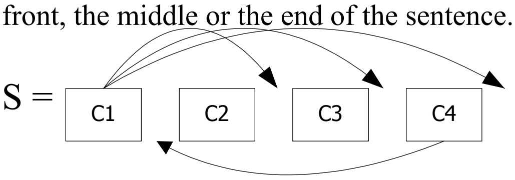


Figure 4 The movements of constituent

Ex.: ตอนเช้า ชาวประมง ออกเรือ หาปลา

In the morning, the fisherman goes to catch the fish

ชาวประมง ออกเรือ หาปลา ตอนเช้า

The fisherman goes to catch the fish in the morning.

ชาวประมง ออกเรือ ตอนเช้า หาปลา

The fisherman goes to in the morning, catch the fish

ตอนเช้า หาปลา ชาวประมง ออกเรือ

In the morning, catch the fish, the fisherman goes to.

Noun, adverb, and prepositional phrase are often move while verb phrases are.

3.3.2 Zero Anaphora

To make the cohesion in the discourse, the anaphora is used as a reference to “point back” to some entities called referent or antecedent, given in the preceding discourse. Halliday, M.A.K. and Hasan, Rugaiya (1976) divided cohesion in English into 5 categories as shown in Table 8:

Table 8: Categories of anaphora

Reference	- Personal Reference, Demonstrative Reference, Comparative Reference
Substitution	- Nominal Substitution, Verbal Substitution, Causal Substitution
Ellipsis	- Nominal Ellipsis, Verbal Ellipsis, Causal Ellipsis
Conjunction	- Additive, Adversative, Casual, Temporal
Lexical Cohesion	- Reiteration(Repetition, Synonym or Near Synonym, Super ordinate, General word) - Collocation

Observing from the corpus in: news, magazine and agricultural text, there are 4 types of anaphora. Ellipsis or zero anaphora was found most frequently in Thai documents and other anaphora happened as show in table 9.

Table 9: Types of reference

Type of Anaphora	Magazine	news	agriculture
Zero anaphora	49.88%	52.38%	50.04%
repetition	32.04%	27.78%	34.49%
personal reference	12.18%	12.70%	1.87%
nominal substitution	5.90%	6.08%	13.60%

Zero anaphora is the use of a gap, in a phrase or clause that has an anaphoric function similar to a pro-form. It is often described as "referring back" to an expression that supplies the information necessary for interpreting the gap

The following is a sentence that illustrates zero anaphora:

มีถนนสองสายที่ต้องไป ตรงแต่แคบ และกว้างแต่คดเคี้ยว

- *There are two roads to eternity, straight but narrow, and broad but crooked.*

In this sentence, the gaps in *straight but narrow [gap], and broad but crooked [gap]* have a zero anaphoric relationship to *two roads to eternity.*

Table 10 also shows the occurrence of zero anaphora in various parts of a sentence.

Table 10: Position of reference in sentences

Position	Frequency
Subject	49.88%
Object	32.04%
Possessive Pronoun	12.18%
Following a Preposition	5.90%

It is noticeable that zero anaphora in the position of the subject occurs with high frequency (49.88%). It shows that in Thai language, the position of subject is the most commonly replaced.

3.3.3 Iterative Relative Clause

Thai relative pronouns "*ที่*" (thi) "ซึ่ง(sung)" and " อัน(un)" relate to group of nouns or other pronouns (The student "*ที่*" (thi) studies hardest usually does the best.). The word "*ที่*" (thi) connects or relates the subject, *student*, to the verb within the dependent clause (*studies*). Generally, we use "*ที่*" (thi) and " "ซึ่ง(sung)" to introduce clauses that are parenthetical in nature (i.e., that can be removed from the sentence without changing the essential meaning of the sentence. The pronoun "*ที่*" (thi) and "ซึ่ง(sung)" refers to things and people and "อัน(un)" usually refers to things, but it can also refer to event in general.

The relative pronoun is sometimes omitted because it makes the sentence more efficient and elegant.

- หนังสือ ~~ที่/ซึ่ง~~ คุณ สั่งซื้อ จาก ร้านนั้น มาถึงแล้วเมื่อ 2 วันก่อน

The book ~~that~~ you ordered from that shop arrived two days later.

Sometimes relative pronoun refers to an event that takes place repeatedly in a phrase.

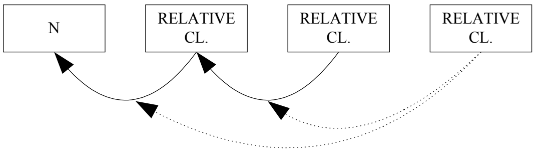


Figure 5 The structure of relative clause

Ex. [พ่อครัว]N [(ที่) ชนะการแข่งขันทำอาหาร]Rel Cl.

[The chef] [who won the cooking competition]

[(ซึ่ง) จัดขึ้นที่ประเทศฝรั่งเศส] Rel Cl. [(ที่) ฉันจ้างมา] Rel Cl.

[which compete at France] [that I employ]

Although a sentence, which has several clauses inside, will be grammatical, but it is not a good style in writing and always causes a problem for parser and noun phrase recognition.

4. The Computational Model

The computational models in word and phrase level are developed according to the phenomena mentioned in section 3.

4.1 Unknown Word Extraction

Unknown word extraction model composes of 2 sub-modules: unknown word recognition and name entity identification.

4.1.1 Unknown word recognition

The hybrid model approach has been used for unknown word recognition. The approach is the combination of a statistical model and a set of context based rules. A statistical model is used to identify unknown word's boundary. The set of context based rules, then, will be used to extract the unknown word's semantic concept. If the unknown word has no context, a set of unknown word information, which has defined through corpus analysis, will be generated and the best one will be selected, as its semantic concept, by using the semantic tagging model. Unknown word recognition process is shown in figure 6.

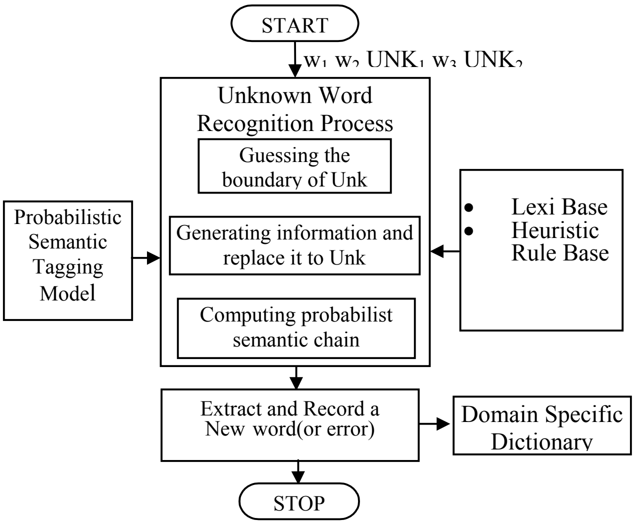

START
w1 w2 UNK1 w3 UNK2
Unknown Word Recognition Process
Guessing the boundary of Unk
Generating information and replace it to Unk
Computing probabilist semantic chain
Probabilistic Semantic Tagging Model
Lexi Base
Heuristic Rule Base
Extract and Record a New word(or error)
Domain Specific Dictionary
STOP

Figure 6: Unknown word recognition process

4.1.2 Name Entity Identification

After unknown words have been extracted, Named Entity (NE) Identification will define the category of unknown word. The model based on heuristic rules and mutual information. Mutual information or statistical analysis of word collocation is used to solve boundary ambiguity when names were composed with known and unknown words. We use Knowledge based such as list of known name (such as country names), clue word list (such as person's title) to support the heuristic rules. Using clue word or common noun that precedes the name can specify NE categorization. Based on the case grammar, NE categories can also defined. Moreover, the lists of the names from predefined NE Ontology can be used for predicting category too. The overview of our system is shown in figure 7. More detail sees (Chanlekha, H. et al, 2002)

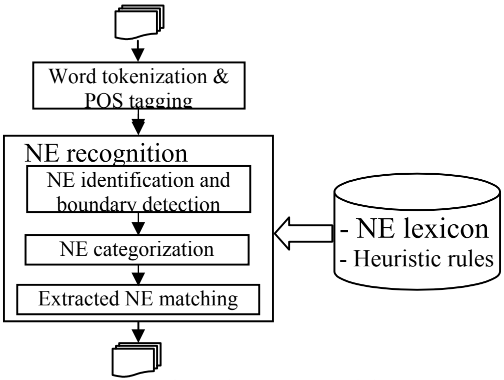


Figure 7 : Named Entity Recognition System

4.2 New Word Generation

Word formation is proposed to reduce the lexicon size by constructing new words or compound noun from the existing words. Based on word formation rules and common dictionary, the shallow parser will extract a set of candidate compound nouns. Then probabilistic approach based on syntactic structure and statistical data is used to solve the problem of over- and under-generation of new word construction and prune the erroneous of compound noun from the candidate set. The process of new word construction is shown in figure 8. See more detail in (Pengphon, N. et al, 2002)

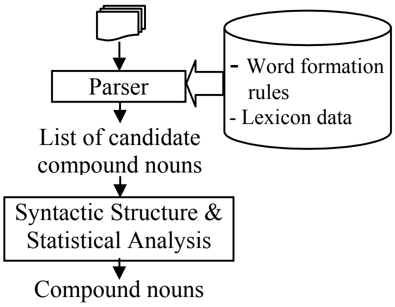

Parser
- Word formation rules
- Lexicon data
List of candidate compound nouns
Syntactic Structure & Statistical Analysis
Compound nouns

Figure 8 : New Word Construction process

4.3 Noun Phrase Recognition

Entities or concepts are usually described by noun phrases. This indicates that text chunks like noun phrases play an important role in human language processing. In order to analyze NP, both statistical and linguistic data are used. The model of NP analysis system is shown in figure 9. More detail sees (Pengphon, N. et al, 2002)

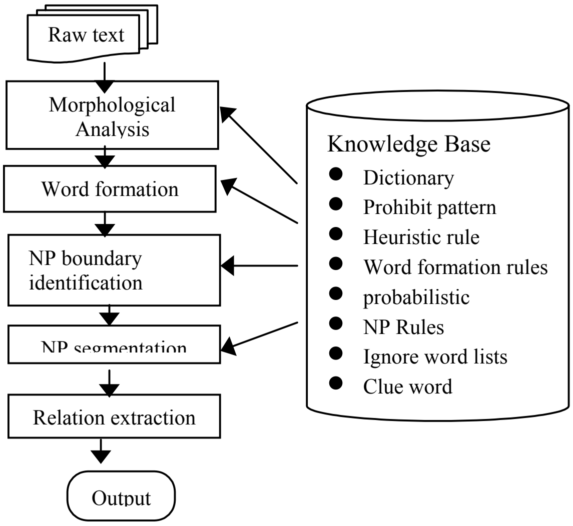


Figure 9 The architecture of system

The first step is morphological analysis for word segmentation and POS tagging. At the second step, the compound word is grouped into one word by using word formation module (see 4.2). The third step, statistical-based technique is used to identify phrase boundary. This step was provided for identifying the phrase boundary by using NP rules. Next step is Noun Phrase Segmentation. The condition of noun phrase segmentation is shown in figure 10.

$$\{w_1,w_2,R_{12}\} \begin{cases} w_2 \in \text{clue word set} & ;\ w_2=\text{start sub np} \\ R_{12} < \text{threshold} & ;\ w_2= \text{start sub np} \\ \text{else} & ;\ w_1w_2 \text{ is one unit} \end{cases}$$

Figure 10 Noun phrase Segmentation

After noun phrase is correctly detected, the relation in noun phrases will be extracted. There are 2 types of relation: head-head noun phrase and head-modifier noun phrase. The process is based on statistical techniques by considering the frequency (f_i) of each word (w_i) in the document (See figure 11).

$$\begin{matrix} w_1w_2w_3w_4 \\ f_1\ f_2\ f_3\ f_4 \end{matrix} \begin{cases} \text{Head modifier} \quad ;\ f_1 > f_2, f_3, f_4 \\ \quad w_1\ \ w_2w_3w_4 \\ \text{Compound} \quad ;\ f_1 = f_2 = f_3 = f_4 \\ \quad w_1w_2w_3w_4 \end{cases}$$

Figure 11: Noun phrase relation

5. Conclusion

The computational language models for Thai in word and phrase level, consisting of Unknown Word Extraction and Name Entities identification, New word generation and Noun phrase recognition, are studied on the basis of their behavior analysis from the varieties of corpus. We expected that it could create cost-effective solutions to the practical problems in the application developments especially in Thai Information Retrieval and Information extraction system. We also give the gateway to access Thai language resources with hoping that it could be the bridge of the international collaboration for developing Multi-Language Processing applications.

Acknowledgement

This project has been supported by NECTEC.

Reference

[1] Bourigault, D.“Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases”. Proc. COLING 1992, 1992.

[2] Chen Kuang-hua and Chen Hsin-His, “Extracting Noun Phrases from large-scale Texts: A Hybrid Approach and Its Automatic Evaluation”, Proc. of the 32nd ACL Annual Meeting, 1994.

[3] Chanlekha, H. et al, “ Statistical and Heuristic Rule Based Model for Thai Named Entity Recognition”, Proc. of SNLP 2002, 2002.

[4] G. Salton, “Automatic Text Processing. The Transformation, Analysis, and Retrieval of Information by Computer”, Singapore: Addison-Wesley Publishing Company, 1989.

[5] Halliday,M.A.K and Hasan,Rugaiya. “Cohesion in English”. Longman Group, London, 1976.

[6] Kawtrakul, A.et.al., “Automatic Thai Unknown Word Recognition”, Proc.of the Natural Language Processing Pacific Rim Symposium, Phuket,1997.

[7] Kawtrakul, A.et.al.,“Backward Transliteration for Thai Document Retrieval”, Proc.of The1998 IEEE Asia-Pacific Conference on Circuits and Systems, Chiangmai, 1998.

[8] Kawtrakul, A. et.al., “Toward Automatic Multilevel Indexing for Thai Text retrieval System”, In Proceedings of The 1998 IEEE Asia-Pacific Conference on Circuits and Systems, Chiangmai, 1998.

[9] Kawtrakul, A.“A Lexibase Model for Writing Production Assistant System” Proc.SNLP’95, 1995.

[10] Kawtrakul, A.“Anaphora Resolution Based On Context Model Approach In Database-Oriented Discourse”. A Doctoral Thesis to The Department of Information Engineering, School of Engineering, Nagoya University, Japan, 1991.

[11] Pengphon, N. et al, “Word Formation Approach and Noun Phrase Analysis for Thai” ”, Proc. of SNLP 2002, 2002.

[12] Sornlertlamvanich, V. et.al., “ORCHID: THAI Part of Speech Tagged Corpus. Technical Report of NECTEC, 1997.

[13] WEBSITE : http:// thaiarc.ku.ac.th

Broadening the Scope of the EAGLES/ISLE Lexical Standardization Initiative

Nicoletta CALZOLARI
Istituto di Linguistica Computazionale, CNR
Area della Ricerca, Via Moruzzi 1
Pisa, Italy, 56100
glottolo@ilc.cnr.it

Alessandro LENCI
Dipartimento di Linguistica, Università di Pisa
Via S. Maria 36
Pisa, Italy, 56100
alessandro.lenci@ilc.cnr.it

Francesca BERTAGNA
Dipartimento di Linguistica, Università di Pisa
Via S. Maria 36
Pisa, Italy, 56100
francesca.bertagna@ilc.cnr.it

Antonio ZAMPOLLI
Istituto di Linguistica Computazionale, CNR
Area della Ricerca, Via Moruzzi 1
Pisa, Italy, 56100
pisa@ilc.cnr.it

Abstract

ISLE is a continuation of the long standing EAGLES initiative and it is supported by EC and NSF under the Human Language Technology (HLT) programme. Its objective is to develop widely agreed and urgently demanded standards and guidelines for infrastructural language resources, tools, and HLT products. EAGLES itself is a well-known trademark and point of reference for HLT projects and products and its previous results have already become *de facto* widely adopted standards. Multilingual computational lexicons, natural interaction and multimodality, and evaluation are the three areas targeted by ISLE. In the first section of the paper we describe the overall goals and methodology of EAGLES/ISLE, in the second section we focus on the work of the Computational Lexicon Working Group, introducing its work strategy and the preliminary guidelines of a standard framework for multilingual computational lexicons, based on a general schema for the "Multilingual ISLE Lexical Entry" (MILE).

1 Introducing EAGLES/ISLE

ISLE (*International Standards for Language Engineering*) is a continuation of the long standing European EAGLES (*Expert Advisory Group for Language Engineering Standards)* initiative (Calzolari *et al.*, 1996), carried out through a number of subsequent projects funded by the European Commission (EC) since 1993. ISLE is an initiative under the Human Language Technology (HLT) programme within the EU-US International Research Co-operation with the aim to develop and promote widely agreed and urgently demanded HLT standards, common guidelines and best practice recommendations for infrastructural language resources (Zampolli, 1998), (Calzolari, 1998), tools that exploit them, and language engineering products.

Object of EAGLES/ISLE are large-scale language resources (such as text corpora, computational lexicons, speech corpora, multimodal resources), means of manipulating such knowledge via computational linguistic formalisms, mark-up languages and various software tools and means of assessing and evaluating resources, tools and products (EAGLES EWG final report, 1996).

EAGLES was set up to determine which aspects of our field are open to short-term *de facto* standardisation and to encourage the development of such standards for the benefit of consumers and producers of language technology, through bringing together representatives of major collaborative European R&D projects, and of HLT industry, in relevant areas. In this respect, more than 150 leading industrial and academic players in the HLT field have actively participated in the definition of this initiative and have lent invaluable support to its execution.

Successful standards are those which respond to commonly perceived needs or aid in overcoming common problems. In terms of offering workable, compromise solutions, they must be based on some solid platform of accepted facts and acceptable practices.

The current ISLE project[1] targets the three areas of :

-*multilingual computational lexicons*[2],

-*natural interaction and multimodality* (*NIMM*)[3],

-*evaluation of HLT systems*[4].

For *multilingual computational lexicons*, ISLE goals are: i) extending EAGLES work on lexical semantics, necessary to establish inter-language links; ii) designing and proposing standards for multilingual lexicons; iii) developing a prototype tool to implement lexicon guidelines and standards; iv) creating exemplary EAGLES-conformant sample lexicons and tagging exemplary corpora for validation purposes; v) developing standardised evaluation procedures for lexicons.

For *NIMM*, ISLE work is targeted to develop guidelines for: i) the creation of NIMM data resources; ii) interpretative annotation of NIMM data, including spoken dialogue in NIMM contexts; iii) metadata descriptions for large NIMM resources; iv) annotation of discourse phenomena.

For *evaluation*, ISLE is working on: i) quality models for machine translation systems; ii) maintenance of previous guidelines - in an ISO based framework (ISO 9126, ISO 14598).

Three Working Groups, and their sub-groups, carry out the work, according to the EAGLES methodology, with experts from both the EU and US, acting as a catalyst in order to pool concrete results coming from major international/national/industrial projects. Relevant common practices or upcoming standards are being used where appropriate as input to EAGLES/ISLE work. Numerous theories, approaches, and systems are being taken into account as any recommendation for harmonisation must take into account the needs and nature of the different major contemporary approaches.

Results are widely disseminated, after due validation in collaboration with EU and US HLT R&D projects, National projects, and industry.

In the following we concentrate on the Computational Lexicon Working Group (CLWG), trying to describe its specific methodology and its goal of establishing a general and consensual standardized environment for the development and integration of multilingual resources. The general vision adheres to the idea of enhancing the sharing and reusability of multilingual lexical resources, by promoting the definition of a common parlance for the community of multilingual HLT and computational lexicon developers. The CLWG pursues this goal by proposing a general schema for the encoding of multilingual lexical information, the *MILE* (Multilingual ISLE Lexical Entry). This has to be intended as a meta-entry, acting as a common representational layer for multilingual lexical resources.

We describe the preliminary proposals of guidelines for the MILE, highlighting some methodological principles applied in previous EAGLES.

2 The Computational Lexicon Working Group

Existing EAGLES results in the Lexicon and Corpus areas are currently adopted by an impressive number of European - and recently also National – projects and has became the "*de-facto* standard" for LR in Europe. This is a very good measure of the impact – and of the

[1] Coordinated by A. Zampolli for EU and M. Palmer for US, see http://www.ilc.pi.cnr.it/EAGLES96/isle/ISLE_Home_Page.htm.

[2] EU chair: N. Calzolari; US chairs: M. Palmer and R. Grishman.

[3] EU chair: N. O. Bernsen; US chair: M. Liberman.

[4] EU chair: M. King; US chair: E. Hovy.

need – of such a standardisation initiative in the HLT sector. To mention just a few key examples:

- the LE PAROLE/SIMPLE resources (morphological/syntactic/semantic lexicons and corpora for 12 EU languages (Zampolli, 1997) (Ruimy *et al*., 1998) (Lenci *et al*., 1999) (Bel *et al*., 2000) rely on EAGLES results (Sanfilippo *et al*., 1996) (Sanfilippo *et al*., 1999), and are now being enlarged to real-size lexicons through many National Projects, thus building a really large infrastructural platform of harmonised lexicons in Europe, sharing the same model;
- the ELRA Validation Manuals for Lexicons (Underwood and Navarreta, 1997) and Corpora (Burnard *et al*., 1997) are based on EAGLES guidelines;
- morpho-syntactic encoding of lexicons and tagging of corpora in a very large number of EU, international and national projects – and for more than 20 languages — is conformant to EAGLES recommendations (Monachini and Calzolari, 1996) (Monachini and Calzolari, 1999) (Leech and Wilson, 1996).

Standards must emerge from state-of-the-art developments. The process of standardisation, although by its own nature not intrinsically innovative, must – and actually does – proceed shoulder to shoulder with the most advanced research. Since ISLE involves many bodies active in EU-US NLP and speech projects, close collaboration with these projects is assured and, significantly, free manpower has been contributed by the projects, as a sign of both their commitment and of the crucial importance they place on reusability issues.

Lexical semantics has always represented a sort of *wild frontier* in the investigation of natural language. In fact, the number of open issues in lexical semantics both on the representational, architectural and content level might induce an actually unjustified negative attitude towards the possibility of designing standards in this difficult territory. Rather to the contrary, standardisation must be conceived as enucleating and singling out the areas in the open field of lexical semantics, that already present themselves with a clear and high degree of stability, although this is often hidden behind a number of formal differences or representational variants, that prevent the possibility of exploiting and enhancing the aspects of commonality and the already consolidated achievements.

With no intent of imposing any constraints on investigation and experimentation, the ISLE CLWG rather aims at selecting mature areas and results in computational lexical semantics and in multilingual lexicons, which can also be regarded as stabilised achievements, thus to be used as the basis for future research. Therefore, consolidation of a standards proposal must be viewed, by necessity, as a slow process comprising, after the phase of putting forward proposals, a cyclical phase involving ISLE external groups and projects with: i) careful evaluation and testing of recommendations in concrete applications; ii) application, if appropriate, to a large number of European languages; iii) feedback on and readjustment of the proposals until a stable platform is reached; dissemination and promotion of consensual recommendations.

The process of standard definition undertaken by CLWG represents an essential interface between advanced research in the field of multilingual lexical semantics, and the practical task of developing resources for HLT systems and applications. It is through this interface that the crucial trade-off between research practice and applicative needs will actually be achieved.

In what follows we briefly describe the two-step strategy adopted in the journey towards standards design: a first activity of survey of existing multilingual resources both in the European and American research and industrial scenarios. A second ongoing phase aiming at individuating hot areas on the domains of multilingual lexical resources, which call – and *de facto* can access to – a process of standardisation.

2.1 Preliminary Step: the Survey Phase

Following the well established EAGLES methodology, the first priority was to do a wide-range survey of bilingual/multilingual (or semantic monolingual) lexicons, so as to reach a fair level of coverage of existing lexical resources. With respect to this target, one of the first objectives is to discover and list the (maximal) set of (granular) *basic notions* needed to describe the multilingual level. The *Survey* of existing lexicons (Calzolari, Grishman and Palmer, 2001) has been accompanied by the analysis of the requirements of a few multilingual applications, and by the parallel analysis of typical cross-lingually complex phenomena[5]. The main issue is how to state in the most proper way the translation correspondences among entries in the multilingual lexicon. The passage from source language (SL) to target language (TL) makes it necessary to express very complex and articulated transfer conditions, which have to take into account as difficult and pervasive phenomena as argument switching, multi-word expressions, collocational patterns, etc.

The function of an entry in a multilingual lexicon is to supply enough information to allow the system to identify a distinct sense of a word or phrase in SL, in many different contexts, and reliably associate each context with the most appropriate translation. The first step is to determine, of all the information that can be associated with SL lexical entries, what is the most relevant to a particular task. We decided to focus the work of survey and subsequent recommendations around two major broad categories of application: Machine Translation and Cross-Language Information Retrieval. They have partially different/complementary needs, and can be considered to represent the requirements of other application types. It is necessary in fact to ensure that any guidelines meet the requirements of industrial applications and that they are implementable.

In the Survey, some Korean and Japanese examples were present in the *case study* dedicated to relevant cross-linguistic phenomena, (e. g. sense distinctions according to variation in syntactic frames/semantic type/ domain information, differences in predicate argument structure, argument incorporation, conflation, head switching etc).

2.2 Towards the Recommendation Phase: designing the MILE Architecture

Since the architecture of the PAROLE-SIMPLE lexicons has been selected to provide the necessary bootstrapping basis for the stepwise refinement cycle leading to MILE, we briefly provide here some information about these resources. The design of the SIMPLE lexicons (Bel *et al.*, 2000) complies with the EAGLES Lexicon/Semantics Working Group guidelines (Sanfilippo *et al.*, 1999), and the set of recommended semantic notions.

The SIMPLE lexicons are built as a new layer connected to the PAROLE syntactic layer, and encode structured "semantic types" and semantic (subcategorization) frames. They cover 12 languages (Catalan, Danish, Dutch, English, Finnish, French, German, Greek, Italian, Portuguese, Spanish, Swedish). The common model is designed to facilitate future cross-language linking: they share the same *core ontology* and the same set of *semantic templates*.

The "conceptual core" of the lexicons consists of the basic structured set of "semantic types" (the *SIMPLE ontology*) and the basic set of notions to be encoded for each Semantic Unit (*SemU*): domain information, lexicographic gloss, argument structure, selectional restrictions/preferences on the arguments, event type, links of the arguments to the syntactic subcategorization frames as represented in the PAROLE lexicons, 'qualia' structure, following the Generative Lexicon (Pustejovsky, 1995), semantic relations, etc..

SIMPLE and PAROLE lexicons are layered resources, with links between the morphological and syntactic layers expressed in PAROLE and the semantic information present in SIMPLE.

In its general design, also MILE is envisaged as a highly *modular* and *layered* architecture as described in Calzolari *et al.* (2001). Modularity concerns the "horizontal" MILE organization, in which independent and yet linked modules

[5] Contributors are: Atkins, Bel, Bertagna, Bouillon, Calzolari, Dorr, Fellbaum, Grishman, Habash, Lange, Lehmann, Lenci, McCormick, McNaught, Ogonowski, Palmer, Pentheroudakis, Richardson, Thurmair, Vanderwende, Villegas, Vossen, Zampolli.

target different dimensions of lexical entries. On the other hand, at the "vertical" level, a layered organization is necessary to allow for different degrees of granularity of lexical descriptions, so that both "shallow" and "deep" representations of lexical items can be captured. This feature is particularly crucial in order to stay open to the different styles and approaches to the lexicon adopted by existing multilingual systems.

At the top level, MILE includes two main modules, *mono-MILE*, providing monolingual lexical representations, and *multi-MILE*, where multilingual correspondences are defined. With this design choice the ISLE-CLWG intends also to address the particularly complex and yet crucial issue of multilingual resource development through the integration of monolingual computational lexicons. As in the reference model, PAROLE/SIMPLE, Mono-MILE is organized into independent modules, respectively providing *morphological*, *syntactic* and *semantic* descriptions. The latter surely represents the core and the most challenging part of the ISLE-CLWG activities, together with the two other crucial topics of *collocations* and *multi-word expressions*, which have often remained outside astandardization initiatives, and nevertheless have a crucial role at the multilingual level. This bias is motivated by the necessity of providing an answer to the most urgent needs and desiderata of next generation HLT, as also expressed by the industrial partners participating to the project. With respect to the issue of the representation of multi-word expressions in computational lexicons, the ISLE-CLWG is actively cooperating with the NSF sponsored XMELLT project (Calzolari *et al.*, 2002).

Multi-MILE specifies a formal environment for the characterization of multilingual correspondences between lexical items. In particular, source and target lexical entries can be linked by exploiting (possibly combined) aspects of their monolingual descriptions. Moreover, in multi-MILE both syntactic and semantic lexical representations can also be enriched, so as to achieve the granularity of lexical description required to establish proper multilingual correspondences, and which is possibly lacking in the original monolingual lexicons.

According to the ISLE approach, monolingual lexicons can thus be regarded as *pivot lexical repositories*, on top of which various language-to-language multilingual modules can be defined, where lexical correspondences are established by partly exploiting and partly enriching the monolingual descriptions. This architecture guarantees the independence of monolingual descriptions while allowing for the maximum degree of flexibility and consistency in reusing existing monolingual resources to build new bilingual lexicons.

The MILE architecture is intended to provide the common representational environment needed to implement such an approach to multilingual resource development, with the goal of maximizing the reuse, integration and extension of existing monolingual computational lexicons.

In the process of specifying the various components of MILE, the ISLE-CLWG has adopted a two-track strategy:

1) identifying the lexical dimensions and the various types of information which are relevant to establish multilingual correspondences;
2) idefining a suitable formal data model to encode this information as well as the operations required at the multilingual level.

To tackle point 1) the survey of the available computational lexicons (see section 2.1) has been complemented with a more lexicographic-based effort, to identify the types of information used in bilingual dictionaries to establish translation equivalents. To this purpose, the CLWG has organized two "task forces" with the responsibility respectively of creating a sample of lexical entries and investigating the use of the so-called *sense indicators* in traditional bilingual dictionaries. The work on *sense indicators* has been carried out mainly by S. Atkins and P. Bouillon: sense indicators are the 'clues' given by the lexicographer to the bilingual dictionary users in order to guide them to the most appropriate choice of equivalence in the foreign language. The source word with its syntactic category, the target words and the sense indicators were automatically extracted from an English-French dictionary and then the sense indicators have been classified on the basis of lexical relevant facts (cf. Atkins *et al.*, 2002).

The aim of these activities has been twofold: on one hand, we wanted to be able to highlight the various types of information useful to determine the transfer conditions; on the other, we had to explore and evaluate the full expressive potentialities provided by the reference computational model (i.e. the PAROLE-SIMPLE architecture).

3.2 The MILE Data Structure and Lexicographic Environment

The CLWG is setting up a lexicographic environment consisting of the following four main components: i) the *MILE Entry Skeleton*, ii) the *MILE Lexical Data Categories*, iii) the *MILE Shared Lexical Objects*, iv) the *ISLE Lexicographic Station*.

The *MILE Entry Skeleton*, formalized as an XML DTD, is an Entity Relationship model that will define the general constraints for the construction of multilingual entries, as well as the grammar to build the whole array of lexical elements needed for a given lexical description.

The *MILE Lexical Data Categories* will provide the lexical objects (syntactic and semantic features, semantic relations, syntactic constructions, predicates and arguments etc..) that are the basic components of MILE-conformant lexical entries. Lexical Data Categories will be organized in a hierarchy and will be defined using RDF schema (Brickley and Guha, 2000) to formalize their properties and make their "semantics" explicit.

The *MILE Shared Lexical Objects* will instantiate the MILE Lexical Data Categories, to be used to build in an easy and straightforward way lexical entries. These will include main syntactic constructions, basic operations and conditions to establish multilingual links, macro-semantic objects, such as lexical conceptual templates acting as general constraints for the encoding of semantic units.

For instance, at the multilingual level it is possible to identify a first set of basic operations that are at the basis of multilingual transfer tests and actions. This would include: i) adding to a monolingual lexical entry a new syntactic position (required for a given translation correspondence); ii) adding to a monolingual semantic description a new semantic feature (required for a given translation correspondence); iii) constraining the source-target correspondence to apply only if an existing syntactic position is realized by a certain type of phrase, etc.

Lexical objects will be identified by an URI and will act as a common resources for lexical representation, to be in turn described by RDF metadata. The defined lexical objects will be used by the lexicon (or applications) developers to build and target lexical data at a higher level of abstraction. Thus, they have to be seen as a step in the direction of simplifying and improving the usability of the MILE recommendations.

The ISLE Lexicographic Station is a development platform used to automatically generate a prototype tool starting from the MILE DTD. The aim of this prototype tool is to i) exemplify the MILE entry ii) make extensive use of already existing monolingual resources, and iii) eventually test the guidelines in a real scenario. This situation led us to define a lexicographic station development platform that guarantees the portability of the final prototype to the final specifications as well as to existing monolingual resources which will serve as the basic data for MILE (for a detailed description, cf. Villegas and Bel, 2002).

Both at monolingual and multilingual level (but with particular emphasis on the latter), ISLE intends to start up the incremental definition of a more Object-Oriented layer for lexical description and to foster the vision of open and distributed lexicons, with elements possibly residing in different sites of the web.

3 Enlargement to Asian Languages

An enlargement of the group to involve also Asian languages is going on and representatives of Chinese, Japanese, Korean, and Thai languages have contributed to ISLE work and participated in some ISLE workshops.

The cooperation between Asia and Europe has to be pursued also through new common initiatives, as the expression of interest for the creation of an *Open Distributed Lexical Infrastructure* that has been submitted to the European Commission for the 6th Framework Programme for Research.

This expression of interest is supported by many non-EU participants, as the newly formed Asian Federation of Natural Language Processing Associations (AFNLPA), the Department of Computer Science of the University of Tokyo, the Korean KAIST and KORTERM, the Taiwanese Institute of Linguistics of the Academia Sinica.

The *Open Distributed Lexical Infrastructure,* a natural development of the ISLE model, can be seen as a new paradigm of distributed lexicon creation and maintenance and it would be a step of great importance for the fulfilment of the vision of the Semantic Web (Berners-Lee, 1998). The creation of such infrastructure has to be *consensual* and in this regard needs the collaboration of a group of languages as large as possible (for example the AFNLPA brings into the initiative many Asian languages, such as Chinese, Hindi, Indonesian, Japanese, Korean, Malay, Tamil, Thai and Urdu). A prerequisite in order to reach interoperability is the existence of best practices and standards that have been consensually agreed on or have been submitted to the international community as *de-facto* standards.

4 Conclusions

In this paper we presented overall goals and methodological principles of the standardization activity of EAGLES/ISLE. In particular, we describe the work of the Computational Lexicon Working Group and its effort towards recommendations, focussing on the MILE, the multilingual lexical meta-entry proposed as the standard representational format for multilingual computational lexical resources. Lexical representation is articulated over different information layers, each factoring out different, but possibly inter-related, linguistic facets of information, relevant in order to establish multilingual lexical links. We also pointed out the necessity to involve a broader group of languages in order to ensure the achievement of a real consensual standard.

Acknowledgements

We thank all the members of the ISLE CLWG for their active participation and contribution to this enterprise.

References

Atkins S., Bel N., Bertagna F., Bouillon P., Calzolari N., Fellbaum C., Grishman R., Lenci A., MacLeod C., Palmer M., Thurmair R., Villegas M., Zampolli A. (2002) *From Resources to Applications. Designing The Multilingual ISLE Lexical Entry*. In Proceedings of LREC 2002, Las Palmas, Canary Islands, Spain.

Bel N., Busa, F., Calzolari, N., Gola, E., Lenci, A., Monachini, M., Ogonowski, A., Peters, I., Peters, W., Ruimy, N., Villegas, M., Zampolli A. (2000) *Simple: A General Framework for the Development of Multilingual Lexicons*. In: LREC Proceedings. Athens.

Berners-Lee T. (1998) *Semantic Web road map*. Personal note. Available at: http://www.w3.org/DesignIssues/Semantic.html.

Burnard L., Baker P., McEnery A., Wilson A. (1997) *An analytic framework for the validation of language corpora*. Report of the Elra Corpus Validation Group.

Calzolari N (1998) An Overview of Written Language Resources in Europe: a few Reflections, Facts, and a Vision. In: Rubio, A., Gallardo, N., Castro, R., Tejada A. (eds.) Proceedings of the First International Conference on Language Resources and Evaluation. Granada 217-224.

Calzolari N., Grishman R., Palmer M. (eds.) (2001) Survey of major approaches towards Bilingual/Multilingual Lexicons. ISLE Deliverable D2.1-D3.1. Pisa

Calzolari N., Fillmore C.J., Grishman R., Ide N., Lenci A., MacLeod C., Zampolli A. (2002) Towards Best Practice for Multiword Expressions in Computational Lexicons. In Proceedings of LREC 2002, Las Palmas, Canary Islands, Spain.

Calzolari N., Lenci A, Zampolli A., Bel N., Villegas M., Thurmair G. (2001) The ISLE in the Ocean. Transatlantic Standards for Multilingual Lexicons (with an eye to Machine Translation). In Proceedings of Machine Translation Summit VIII, Santiago de Compostela, Spain.

Calzolari N., Mc Naught J., Zampolli A (1996) Eagles Final Report: Eagles Editors' Introduction. Pisa.

Eagles (1996) Evaluation of Natural Language Processing Systems. Final Report. CST, Copenhagen. Also at http://issco-www.unige.ch/projects/ewg96/ewg96.html.

Brickley D., Guha R. (2000) Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation. Available online at http://www.w3.org/TR/rdf-schema.

Leech G., Wilson A. (1996) Recommendations for the morphosyntactic annotation of corpora. Lancaster.

Lenci A., Busa F., Ruimy N., Gola E., Monachini M., Calzolari N., Zampolli A. (1999) Linguistic Specifications. Simple Deliverable D2.1. ILC and University of Pisa.

Monachini M., Calzolari N. (1996) Synopsis and comparison of morphosyntactic phenomena encoded in lexicons and corpora. A common proposal and applications to European languages. ILC-CNR, Pisa.

Monachini M., Calzolari N. (1999) *Standardization in the Lexicon*. In: H. van Halteren (ed.): Syntactic Wordclass Tagging. Kluwer, Dordrecht 149-173.

Pustejovsky J. (1995) *The Generative Lexicon*. Cambridge, MA, MIT Press.

Ruimy N., Corazzari O., Gola E., Spanu A., Calzolari N., Zampolli A. (1998) *The European LE-Parole Project: The Italian Syntactic Lexicon*. In: Proceedings of the First International Conference on Language resources and Evaluation. Granada 241-248.

Sanfilippo A. et al. (1996) *Eagles Subcategorization Standards.* See http://www.icl.pi.cnr.it/EAGLES96/syntax/syntax.html

Sanfilippo A. et al. (1999) *Eagles Recommendations on Semantic Encoding*.. See http://www.ilc.pi.cnr.it/EAGLES96/rep2

Underwood N., Navarretta C. (1997) *A Draft Manual for the Validation of Lexica*. Final ELRA Report. Copenhagen.

Villegas M., Bel N. (2002) *From DTDs to relational dBs. An automatic generation of a lexicographical station out off ISLE guidelines*. In Proceedings of LREC 2002, Las Palmas, Canary Islands, Spain.

Zampolli A. (1997) *The PAROLE project in the general context of the European actions for Language Resources*. In: Marcinkeviciene, R., Volz, N. (eds.): Telri Proceedings of the Second European Seminar: Language Applications for a Multilingual Europe. IDS/VDU, Manheim/Kaunas.

Zampolli A. (1998) *Introduction of the General Chairman*. In: Rubio, A., Gallardo, N. Castro, R., Tejada A. (eds.): Proceedings of the First International Conference on Language Resources and Evaluation. Granada, Spain.

Lexicon-based Orthographic Disambiguation in CJK Intelligent Information Retrieval
面向中日韩文智能信息检索的基于词典的异形词排歧

Jack Halpern（春遍雀來）jack@cjk.org
The CJK Dictionary Institute（日中韓辭典研究所）
34-14, 2-chome, Tohoku, Niiza-shi, Saitama 352-0001, Japan

Abstract

The orthographical complexity of Chinese, Japanese and Korean (CJK) poses a special challenge to the developers of computational linguistic tools, especially in the area of **intelligent information retrieval**. These difficulties are exacerbated by the lack of a standardized orthography in these languages, especially the highly irregular Japanese orthography. This paper focuses on the typology of CJK orthographic variation, provides a brief analysis of the linguistic issues, and discusses why lexical databases should play a central role in the disambiguation process.

1 Introduction

Various factors contribute to the difficulties of CJK information retrieval. To achieve truly "intelligent" retrieval many challenges must be overcome. Some of the major issues include:

1. The lack of a standard orthography. To process the extremely large number of orthographic variants (especially in Japanese) and character forms requires support for advanced IR technologies such as **cross-orthographic searching** (Halpern 2000).
2. The accurate conversion between Simplified Chinese (SC) and Traditional Chinese (TC), a deceptively simple but in fact extremely difficult computational task (Halpern and Kerman 1999).
3. The morphological complexity of Japanese and Korean poses a formidable challenge to the development of an accurate morphological analyzer. This performs such operations as canonicalization, *stemming* (removing inflectional endings) and *conflation* (reducing morphological variants to a single form) on the morphemic level.
4. The difficulty of performing accurate word segmentation, especially in Chinese and Japanese which are written without interword spacing. This involves identifying word boundaries by breaking a text stream into meaningful semantic units for dictionary lookup and indexing purposes. Good progress in this area is reported in Emerson (2000) and Yu et al. (2000).
5. Miscellaneous retrieval technologies such as lexeme-based retrieval (e.g. 'take off' + 'jacket' from 'took off his jacket'), identifying syntactic phrases (such as 研究する from 研究をした), synonym expansion, and cross-language information retrieval (CLIR) (Goto et al. 2001).
6. Miscellaneous technical requirements such as transcoding between multiple character sets and encodings, support for Unicode, and input method editors (IME). Most of these issues have been satisfactorily resolved, as reported in Lunde (1999).
7. Proper nouns pose special difficulties for IR tools, as they are extremely numerous, difficult to detect without a lexicon, and have an unstable orthography.
8. Automatic recognition of terms and their variants, a complex topic beyond the scope of this paper. It is described in detail for European languages in Jacquemin (2001), and we are currently investigating it for Chinese and Japanese.

Each of the above is a major issue that deserves a paper in its own right. Here, the focus is on **orthographic disambiguation,** which refers to the detection, normalization and conversion of CJK orthographic variants. This paper summarizes the typology of CJK orthographic variation, briefly analyzes the linguistic issues, and discusses why lexical databases should play a central role in the disambiguation process.

2 Orthographic Variation in Chinese

2.1 One Language, Two Scripts

As a result of the postwar language reforms in the PRC, thousands of character forms underwent drastic simplifications (Zongbiao 1986). Chinese written in these simplified forms is called **Simplified Chinese** (SC). Taiwan, Hong Kong, and most overseas Chinese continue to use the old, complex forms, referred to as **Traditional Chinese** (TC).

The complexity of the Chinese writing system is well known. Some factors contributing to this are the large number of characters in common use, their complex forms, the major differences between TC and SC along various dimensions, the presence of numerous orthographic variants in TC, and others. The numerous variants and the difficulty of converting between SC and TC are of special importance to Chinese IR applications.

2.2 Chinese-to-Chinese Conversion

The process of automatically converting SC to/from TC, referred to as **C2C conversion**, is full of complexities and pitfalls. A detailed description of the linguistic issues can be found in Halpern and Kerman (1999), while technical issues related to encoding and character sets are described in Lunde (1999). The conversion can be implemented on three levels in increasing order of sophistication, briefly described below.

2.2.1 Code Conversion The easiest, but most unreliable, way to perform C2C conversion is on a codepoint-to-codepoint basis by looking the source up in a mapping table, such as the one shown below. This is referred to as **code conversion** or **transcoding.** Because of the numerous one-to-many ambiguities (which occur in both the SC-to-TC and the TC-to-SC directions), the rate of conversion failure is unacceptably high.

Table 1. Code Conversion

SC	TC1	TC2	TC3	TC4	Remarks
门	們				one-to-one
汤	湯				one-to-one
发	發	髮			one-to-many
暗	暗	闇			one-to-many
干	幹	乾	干	榦	one-to-many

2.2.2 Orthographic Conversion The next level of sophistication in C2C conversion is referred to as **orthographic conversion,** because the items being converted are orthographic units**,** rather than codepoints in a character set. That is, they are meaningful linguistic units, especially multi-character lexemes. While code conversion is ambiguous, orthographic conversion gives better results because the orthographic mapping tables enable conversion on the word level.

Table 2. Orthographic Conversion

English	SC	TC1	TC2	Incorrect	Comments
telephone	电话	電話			unambiguous
we	我们	我們			unambiguous
start-off	出发	出發		出髮 齣髮 齣發	one-to-many
dry	干燥	乾燥		干燥 幹燥 榦燥	one-to-many
	阴干	陰乾	陰干		depends on context

As can be seen, the ambiguities inherent in code conversion are resolved by using an orthographic mapping table, which avoids false conversions such as shown in the **Incorrect** column. Because of segmentation ambiguities, such conversion must be done with the aid of a morphological analyzer that can break the text stream into meaningful units (Emerson 2000).

2.2.3 Lexemic Conversion A more sophisticated, and far more challenging, approach to C2C conversion is called **lexemic conversion**, which maps SC and TC lexemes that are **semantically,** *not* orthographically, equivalent. For example, SC 信息 (*xìnxī*) 'information' is converted to the semantically equivalent TC 資訊 (*zīxùn*). This is similar to the difference between *lorry* in British English and *truck* in American English.

There are numerous lexemic differences between SC and TC, especially in technical terms and proper nouns, as demonstrated by Tsou (2000). For example, there are more than 10 variants for 'Osama bin Laden.' To complicate matters, the correct TC is sometimes locale-dependent. Lexemic conversion is the most difficult aspect of C2C conversion and can only be done with the help of mapping tables. Table 3 illustrates various patterns of cross-locale lexemic variation.

Table 3. Lexemic Conversion

English	SC	Taiwan TC	Hong Kong TC	Other TC	Incorrect TC (orthographic)
Software	软件	軟體	軟件		軟件
Taxi	出租汽车	計程車	的士	德士	出租汽車
Osama bin Laden	奥萨马本拉登	奧薩瑪賓拉登	奧薩瑪賓拉丹		奧薩馬本拉登
Oahu	瓦胡岛	歐胡島			瓦胡島

2.3 Traditional Chinese Variants

Traditional Chinese does not have a stable orthography. There are numerous TC variant forms, and much confusion prevails. To process TC (and to some extent SC) it is necessary to disambiguate these variants using mapping tables (Halpern 2001).

2.3.1 TC Variants in Taiwan and Hong Kong Traditional Chinese dictionaries often disagree on the choice of the standard TC form. TC variants can be classified into various types, as illustrated in Table 4.

Table 4. TC Variants

Var. 1	Var. 2	English	Comment
裏	裡	inside	100% interchangeable
敎	教	teach	100% interchangeable
著	着	particle	variant 2 not in Big5
為	爲	for	variant 2 not in Big5
沉	沈	sink; surname	partially interchangeable
泄	洩	leak; divulge	partially interchangeable

There are various reasons for the existence of TC variants, such as some TC forms are not being available in the Big Five character set, the occasional use of SC forms, and others.

2.3.2 Mainland vs. Taiwanese Variants To a limited extent, the TC forms are used in the PRC for some classical literature, newspapers for overseas Chinese, etc., based on a standard that maps the SC forms (GB 2312-80) to their corresponding TC forms (GB/T 12345-90). However, these mappings do not necessarily agree with those widely used in Taiwan. We will refer to the former as **"Simplified Traditional Chinese"** (STC), and to the latter as **"Traditional Traditional Chinese"** (TTC).

Table 5. STC vs. TTC Variants

Pinyin	SC	STC	TTC
xiàn	线	綫	線
bēng	绷	綳	繃
cè	厕	厠	廁

3 Orthographic Variation in Japanese

3.1 One Language, Four Scripts

The Japanese orthography is highly irregular. Because of the large number of orthographic variants and easily confused homophones, the Japanese writing system is significantly more complex than any other major language, including Chinese. A major factor is the complex interaction of the four scripts used to write Japanese, resulting in countless words that can be written in a variety of often unpredictable ways (Halpern 1990, 2000).

Table 6 shows the orthographic variants of 取り扱い *toriatsukai* 'handling', illustrating a variety of variation patterns.

Table 6. Variants of *toriatsukai*

Toriatsukai	**Type of variant**
取り扱い	"standard" form
取扱い	okurigana variant
取扱	All kanji
とり扱い	replace kanji with hiragana
取りあつかい	replace kanji with hiragana
とりあつかい	All hiragana

An example of how difficult Japanese IR can be is the proverbial "A hen that lays golden eggs." The "standard" orthography would be 金の卵を産む鶏 (*Kin no tamago wo umu niwatori*). In reality, *tamago* 'egg' has four variants (卵, 玉子, たまご, タマゴ), *niwatori* 'chicken' three (鶏, にわとり, ニワトリ) and *umu* 'to lay' two (産む, 生む), which expands to 24 permutations like 金の卵を生むニワトリ, 金の玉子を産む鶏 etc. As can be easily verified by searching the web, these variants frequently occur in webpages. Clearly, the user has no hope of finding them unless the application supports orthographic disambiguation.

3.2 Okurigana Variants

One of the most common types of orthographic variation in Japanese occurs in kana endings, called 送り仮名 *okurigana,* that are attached to a kanji base or stem. Although it is possible to generate some okurigana variants algorithmically, such as nouns (飛出し) derived from verbs (飛出す), on the whole hard-coded tables are required. Because usage is often unpredictable and the variants are numerous, okurigana must play a major role in Japanese orthographic disambiguation.

Table 7. Okurigana Variants

English	**Reading**	**Standard**	**Variants**
publish	*kakiarawasu*	書き表す	書き表わす 書表わす 書表す
perform	*okonau*	行う	行なう
handling	*toriatsukai*	取り扱い	取扱い 取扱

3.3 Cross-Script Orthographic Variants

Japanese is written in a mixture of four scripts (Halpern 1990): **kanji** (Chinese characters), two syllabic scripts called **hiragana** and **katakana,** and **romaji** (the Latin alphabet). Orthographic variation across scripts, which should play a major role in Japanese IR, is extremely common and mostly unpredictable, so that the same word can be written in hiragana, katakana or kanji, or even in a mixture of two scripts. Table 8 shows the major cross-script variation patterns in Japanese.

Table 8. Cross-Script Variants

Kanji vs. Hiragana	大勢　おおぜい
Kanji vs. Katakana	硫黄　イオウ
Kanji vs. hiragana vs. katakana	猫　ねこ　ネコ
Katakana vs. hybrid	ワイシャツ　Yシャツ
Kanji vs. katakana vs. hybrid	皮膚　ヒフ　皮フ
Kanji vs. hybrid	彗星　すい星
Hiragana vs. katakana	ぴかぴか　ピカピカ

3.4 Kana Variants

Recent years have seen a sharp increase in the use of katakana, a syllabary used mostly to write loanwords. A major annoyance in Japanese IR is that katakana orthography is often irregular; it is quite common for the same word to be written in multiple, unpredictable ways which cannot be generated algorithmically. Hiragana is used mostly to write grammatical elements and some native Japanese words. Although hiragana orthography is generally regular, a small number of irregularities persist. Some of the major types of kana variation are shown in Table 9.

Table 9. Katakana and Hiragana Variants

Type	English	Reading	Standard	Variants
Macron	computer	*konpyuuta* *konpyuutaa*	コンピューター	コンピューター
Long vowels	maid	*meedo*	メード	メイド
Multiple kana	team	*chiimu* *tiimu*	チーム	ティーム
Traditional	big	*ookii*	おおきい	おうきい
づ vs. ず	continue	*tsuzuku*	つづく	つずく

The above is only a brief introduction to the most important types of kana variation. There are various others, including an optional middle dot (*nakaguro)* and small katakana variants (クォ vs. クオ), and the use of traditional (じ vs. ぢ) and historical (い vs. ゐ) kana.

3.5 Miscellaneous Variants

There are various other types of orthographic variants in Japanese, which are beyond the scope of this paper. Only a couple of the important ones are mentioned below. A detailed treatment can be found in Halpern (2000).

3.5.1 Kanji Variants Though the Japanese writing system underwent major reforms in the postwar period and the character forms have by now been standardized, there is still a significant number of variants in common use, such as abbreviated forms in contemporary Japanese (才 for 歳 and 巾 for 幅) and traditional forms in proper nouns and classical works (such as 嶋 for 島 and 發 for 発).

3.5.2 Kun Homophones An important factor that contributes to the complexity of the Japanese writing system is the existence of a large number of homophones (words pronounced the same but written differently) and their variable orthography (Halpern 2000). Not only can each kanji have many *kun* readings, but many *kun* words can be written in a bewildering variety of ways. The majority of *kun* homophones are often close or even identical in meaning and thus easily confused, i.e., *noboru* means 'go up' when written 上る but 'climb' when written 登る, while *yawarakai* ’soft’ is written 柔らかい or 軟らかい with identical meanings.

4 Orthographic Variation in Korean

4.1 Irregular Orthography

The Korean orthography is not as regular as most people tend to believe. Though hangul is often described as "logical," the fact is that in modern Korean there is a significant amount of orthographic variation. This, combined with the morphological complexity of the language, poses a challenge to developers of IR tools. The major types of orthographic variation in Korean are described below.

4.2 Hangul Variants

The most important type of orthographic variation in Korean is the use of variant hangul spellings in the writing of loanwords. Another significant kind of variation is in the writing of non-Korean personal names, as shown in Table 10.

Table 10. Hangul Variants

cake	케이크 (*keikeu*)	케잌 (*keik*)
yellow	옐로우 (*yelrou*)	옐로 (*yelro*)
Mao Zedong	마오쩌뚱 (*maojjeottung*)	모택동 (*motaekdong*)
Clinton	클린턴 (*keulrinteon*)	클린톤 (*keulrinton*)

4.3 Cross-Script Orthographic Variants

A factor that contributes to the complexity of the Korean writing system is the use of multiple scripts. Korean is written in a mixture of three scripts: an alphabetic syllabary called **hangul,** Chinese characters called **hanja** (their use is declining) and the Latin alphabet called **romaja**. Orthographic variation across scripts is not uncommon. The major patterns of cross-script variation are shown Table 11.

Table 11. Cross-Script Orthographic Variants

Type of Variation	English	Var. 1	Var. 2	Var.3
Hanja vs. hangul	many people	大勢 (*daese*)	대세 (*daese*)	
Hangul vs. hybrid	shirt	와이셔츠 (*wai-syeacheu*)	Y셔츠 (*wai-syeacheu*)	
Hangul vs. numeral vs. hanja	one o'clock	한시 (*hansi*)	1시 (*hansi*)	一時 (*hansi*)
English vs. hangul	sex	sex	섹스 (*sekseu*)	

4.4 Miscellaneous Variants

4.4.1 North vs. South Korea Another factor contributing to the irregularity of hangul orthography is the differences in spelling between South Korea (S.K.) and North Korea (N.K.). The major differences are in the writing of loanwords, a strong preference for native Korean words, and in the writing of non-Korean proper nouns. The major types are shown below.

1. **Place names:** N.K. 오사까 (*osakka*) vs. S.K. 오사카 (*osaka*) for 'Osaka'
2. **Personal names:** N.K. 부슈 (*busyu*) vs. S.K. 부시 (*busi*) for 'Bush'
3. **Loanwords:** N.K. 미싸일 (*missail*) vs. S.K. 미사일 (*misail*) for 'missile'
4. **Russian vs. English:** N.K. 그루빠 (*guruppa*) vs. S.K. 그룹 (*geurup*)
5. **Morphophonemic:** N.K. 람용 (*ramyong*) vs. S.K. 남용 (*namyong*)

4.4.2 New vs. Old Orthography The hangul script went through several reforms during its history, the latest one taking place as recently as 1988. Though the new orthography is now well established, the old orthography is still important because the affected words are of high frequency and their number is not insignificant. For example, the modern 일군 'worker' (*ilgun*) was written 일꾼 (*ilkkun*) before 1988, while 빛갈 'color' (*bitgal*) was written 빛깔 (*bitkkal*).

4.4.3 Hanja Variants Although language reforms in Korea did not include the simplification of the character forms, the Japanese occupation of Korea resulted in many simplified Japanese character forms coming into use, such as the Japanese form 発 to replace 發 (*bal*).

4.4.4 Miscellaneous Variants There are various other types of orthographic variation, which are beyond the scope of this paper. This includes the use of abbreviations and acronyms and variation in interword spacing in multiword compounds. For example, 'Caribbean Sea' (*karibeuhae*) may be written solid (카리브해) or open (카리브 해).

5 The Role of Lexical Databases

Because of the irregular orthography of CJK languages, lexeme-based procedures such as orthographic disambiguation cannot be based on probabilistic methods (e.g. bigramming) alone. Many attempts have been made along these lines, as for example Brill (2001) and Goto et al. (2001), with some claiming performance equivalent to lexicon-based methods, while Kwok (1997) reports good results with only a small lexicon and simple segmentor.

These methods may be satisfactory for pure IR (relevant document retrieval), but for orthographic disambiguation and C2C conversion, Emerson (2000) and others have shown that a robust morphological analyzer capable of processing lexemes, rather than bigrams or *n*-grams, must be supported by a large-scale computational lexicon (even 100,000 entries is much too small).

The CJK Dictionary Institute (CJKI), which specializes in CJK computational lexicography, is engaged in an ongoing research and development effort to compile comprehensive CJK lexical databases (currently about 5.5 million entries), with special emphasis on orthographic disambiguation and proper nouns. Listed below are the principal components useful for intelligent IR tools and orthographic disambiguation.

1. **Chinese to Chinese conversion.** In 1996, CJKI launched a project to investigate C2C conversion issues in-depth, and to build comprehensive mapping tables (now at 1.3 million SC and 1.2 million TC items) whose goal is to achieve near 100% conversion accuracy. These include:
    a. SC-to/from-TC code-level mapping tables

b. SC-to/from-TC orthographic and lexemic mapping tables for general vocabulary

c. SC-to/from-TC orthographic mapping tables for proper nouns

d. Comprehensive SC-to/from-TC orthographic/lexemic mapping tables for technical terminology, especially IT terms

2. **TC orthographc normalization tables**

a. TC normalization mapping tables

b. STC-to/from-TTC character mapping tables

3. **Japanese orthographic variant databases**

a. A comprehensive database of Japanese orthographic variants

b. A database of semantically classified homophone groups

c. Semantically classified synonym groups for synonym expansion (Japanese thesaurus)

d. An English-Japanese lexicon for CLIR

e. Rules for identifying unlisted variants

Conclusions

CJK IR tools have become increasingly important to information retrieval in particular and to information technology in general. As we have seen, because of the irregular orthography of the CJK writing systems, intelligent information retrieval requires not only sophisticated tools such as morphological analyzers, but also lexical databases fine-tuned to the needs of orthographic disambiguation.

Few if any CJK IR tools perform orthographic disambiguation. For truly "intelligent" IR to become a reality, not only must lexicon-based disambiguation be supported, but such emerging technologies as CLIR, synonym expansion and cross-homophone searching should also be implemented.

We are currently engaged in further developing the lexical resources required for building intelligent CJK information retrieval tools and for supporting accurate segmentation technology.

References

Brill, E. and Kacmarick, G. and Brocket, C. (2001) *Automatically Harvesting Katakana-English Term Pairs from Search Engine Query Logs*. Microsoft Research, Proc. of the Sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan.

Emerson, T. (2000) *Segmenting Chinese in Unicode.* Proc. of the 16th International Unicode Conference, Amsterdam

Goto, I., Uratani, N. and Ehara T. (2001) *Cross-Language Information Retrieval of Proper Nouns using Context Information*. NHK Science and Technical Research Laboratories. Proc. of the Sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan

Jacquemin, C. (2001) *Spotting and Discovering Terms through Natural Language Processing*. The MIT Press, Cambridge, MA

Halpern, J. (1990) *Outline Of Japanese Writing System.* In "New Japanese-English Character Dictionary*",* 6th printing, Kenkyusha Ltd., Tokyo, Japan (www.kanji.org/kanji/japanese/writing/outline.htm)

Halpern, J. and Kerman J. (1999) *The Pitfalls and Complexities of Chinese to Chinese Conversion*. Proc. of the Fourteenth International Unicode Conference in Cambridge, MA.

Halpern, J. (2000) *The Challenges of Intelligent Japanese Searching.* Working paper (www.cjk.org/ cjk/joa/joapaper.htm), The CJK Dictionary Institute, Saitama, Japan.

Halpern, J. (2001) *Variation in Traditional Chinese Orthography.* Working paper (www.cjk.org/cjk/ cjk/reference/chinvar.htm), The CJK Dictionary Institute, Saitama, Japan.

Kwok, K.L. (1997) *Lexicon Effects on Chinese Information Retrieval*. Proc. of 2nd Conf. on Empirical Methods in NLP. ACL. pp.141-8.

Lunde, Ken (1999) *CJKV Information Processing.* O'Reilly & Associates, Sebastopol, CA.

Yu, Shiwen, Zhu, Xue-feng and Wang, Hui (2000) *New Progress of the Grammatical Knowledge-base of Contemporary Chinese*. Journal of Chinese Information Processing, Institute of Computational Linguistics, Peking University, Vol.15 No.1.

Tsou, B.K., Tsoi, W.F., Lai, T.B.Y. Hu, J., and Chan S.W.K. (2000) *LIVAC,* a *Chinese synchronous corpus, and some applications.* In "2000 International Conference on Chinese Language ComputingICCLC2000", Chicago .

Zongbiao (1986) 简化字总表 (*Jianhuazi zongbiao)* (Second Edition). 国家语言文字工作委员会, 语文出版社, China.

Decomposition for ISO/IEC 10646 Ideographic Characters

LU Qin, CHAN Shiu Tong, LI Yin, LI Ngai Ling

Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

{csluqin, cstchan, csyinli, csnlli}@comp.polyu.edu.hk

Abstract

Ideograph characters are often formed by some smaller functional units, which we call character components. These character components can be ideograph radicals, ideographs proper, or some pure components which must be used with others to form characters. Decomposition of ideographs can be used in many applications. It is particularly important in the study of Chinese character formation, phonetics and semantics. However, the way a character is decomposed depends on the definition of components as well as the decomposition rules. The 12 Ideographic Description Characters (IDCs) introduced in ISO 10646 are designed to describe characters using components. The Hong Kong SAR Government recently published two sets of glyph standards for ISO10646 characters. The standards, being the first of its kind, make use of character decomposition to specify a character glyph using its components. In this paper, we will first introduce the IDCs and how they can be used with components to describe two dimensional ideograph characters in a linear fashion. Next we will briefly discuss the basic references and character decomposition rules. We will then describe the data structure and algorithms to decompose Chinese characters into components and, vice versa. We have also implemented our database and algorithms as an internet application, called the *Chinese Character Search System*, available at website http://www.iso10646hk.net/. With this tool, people can easily search characters and components in ISO 10646.

Introduction

ISO/IEC 10646 (ISO 10646) in its current version, contains more than 27,000 Han characters, or ideograph characters as it is called, in its basic multilingual plane and another 40,000 in the second plane[1-2]. The complete set of ideograph repertoire includes Han characters in all national/regional standards as well as all characters from the Kang Xi Dictionary(康熙字典) and other major references. In almost all the current encoding systems including ISO 10646 and Unicode, each Han character is treated as a separate unique symbol and given a separate code point. This single character encoding method has some serious drawbacks. Consider most of the alphabet-based languages, such as English, even though new words are created quite frequently, the alphabet itself is quite stable. Thus the newly adopted words do not have any impact on coding standards. When new Han characters are created, they must be assigned a new code point, thus all codesets supporting Han characters must leave space for extension. As there is no formal rule to limit the formation of new Han characters, the standardization process for code point assignment can be potentially endless. On the other hand, new Han characters are almost always be created using some existing character components which can be existing radicals, characters proper, or pure components which are not used alone as characters. If we can use coded components to describe a new character, we can potentially eliminate the standardization process.

Han characters can be considered as a two dimensional encoding of components. The same set of components when used in different relative positions can form different characters. For example the two components 大 and 小 can form two different characters: 尖 㚀 depending on the relative positions of the two components. However, the current internal code point assignments in no way can reveal the relationship of the these characters with respect to their component characters. Because of the limitation of the encoding system, people have to put a lot of efforts to develop different input methods. Searching for characters with similar shapes are also quite difficult. The 12

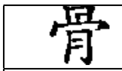
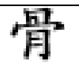
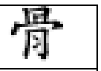
Ideographic Description Characters (IDCs) were introduced in ISO 10646 in the code range of 2FF0 - 2FFB to describe the relative positions of components as shown in Table 1. Each IDC symbol shows a typical ideograph character composition structure. For example, ⿰ (U+2FF0) indicates that a character is formed by two components, one on the left-hand side and one on the right-hand side. All IDCs except U+2FF2 and U+2FF3 have cardinality of two because the decomposition requires two components only. Details of these symbols can be found in Annex F of ISO 10646 2nd Edition [1] and in John Jenkens' report [3].

Smbl	Code point	Name in ISO 10646	Cardi-nality	Label
⿰	2FF0	IDC LEFT TO RIGHT	IDC2	A
⿱	2FF1	IDC ABOVE TO BELOW	IDC2	B
⿲	2FF2	IDC LEFT TO MIDDLE AND RIGHT	IDC3	K
⿳	2FF3	IDC ABOVE TO MIDDLE AND BELOW	IDC3	L
⿴	2FF4	IDC FULL SURROUND	IDC2	I
⿵	2FF5	IDC SURROUND FROM ABOVE	IDC2	F
⿶	2FF6	IDC SURROUND FROM BELOW	IDC2	G
⿷	2FF7	IDC SURROUND FROM LEFT	IDC2	H
⿸	2FF8	IDC SURROUND FROM UPPER LEFT	IDC2	D
⿹	2FF9	IDC SURROUND FROM UPPER RIGHT	IDC2	C
⿺	2FFA	IDC SURROUND FROM LOWER LEFT	IDC2	E
⿻	2FFB	IDC OVERLAID	IDC2	J

Table 1. The 12 Ideograph Description Characters

The IDCs can be used to describe not only unencoded characters, but also coded characters to reveal their internal structures and relationships among components. Thus applications for using these structural symbols can be quite useful. In fact the most common applications are in electronic dictionaries and on-line education [4].

In this paper, however, we introduce a new application where the IDCs and components are used in the standardization of Han character glyphs. As we all know that ISO 10646 is a character standard, which allows different glyph styles for the same character and different regions can develop different glyph styles to suit their own needs. The ideographic repertoire in ISO 10646 has a so called Horizontal Extension, where each coded ideograph character is listed under the respective CJKV columns. The glyph of each character can be different under different columns because ISO 10646 is a character standard, not a glyph standard. We normally call these different glyphs as variants. For example, the character bone 骨 can take three different forms(variants):

		
HK	Mainland	Taiwan

Even with the ISO 10646 horizontal extensions, people in Hong Kong still get confused as to which styles to use, as only some characters in the Hong Kong style deviate from both G column(mainland China) and T column(Taiwan). Consequently, the Hong Kong SAR Government has decided to develop the Hong Kong glyph standards for ISO 10646 which can serve as a reference guide for font vendors when developing products for Hong Kong. The standards, being the first of its kind, makes uses of character decomposition to specify a character glyph using its components.

The rest of the paper is organized as follows. Section 1 gives the rationale for the use of character components, the references and decomposition rules. Section 2 describes the data structure and algorithms to decompose Chinese characters into components and, vice versa. Section 3 discusses performance considerations and Section 4 is the conclusion.

1. Character Decomposition Rules

At the beginning of the glyph standardization, one important requirement was agreed by the working group, namely, *extensibility*. That is, the specifications should be easily extended by adding more characters into later versions of the ISO/IEC 10646, which we refer to as the *new characters*. The specifications should also not contain any internal inconsistency, or inconsistency in relation to the ISO/IEC 10646's source standards. In order to satisfy both consistency requirements, we have concluded that listing every character in ISO/IEC 10646 is not desirable. Instead, we decided to produce the specifications by giving the correct glyphs of character components based on a common assumption that if a component or a character is written in a certain way, all other characters using it as a component should also write it in the same way. For example if the character “bone” 骨 (U+9AA8) is written in a certain way, all characters using “bone” as a component, such as “滑” (U+6ED1) and “骼” (U+9ABC), should have the bone “骨” component follow the same style. In this way, the specification can be extended very easily for all new characters using bone “骨” as a component. In other words, we can assume that component glyphs are standardized for general usage. By using components to describe a character, we can also avoid inconsistency. That is, by avoid listing all characters with bone, “骨” as a component, we do not need to be concerned about producing inconsistent glyphs in the specifications. This is important because the working group does not have any font vendor as a member, because of an implicit rule that was specified by the Government of the HKSAR to avoid any potential conflict of interest. The glyph style is mostly based on the book published by the Hong Kong Institute of Education in 2000[5]

In principle, for producing glyph specifications, we have to produce a concrete, minimal, and unique list of basic components. In order to achieve this, we need to have a set of rules to decompose the characters systematically. In our work, we have used the GF 3001-1997 [6] as our major component reference. The following is a brief description of the rules. (For a detailed description, please refer to the paper “The Hong Kong Glyph Specifications for ISO 10646's Ideographic Characters”[7].)

- Use GF 3001-1997 specifications as the basis to construct a set of primary components. Components for simplified Chinese are removed. The shapes are modified to match the glyph style for Hong Kong.
- Characters are decomposed into components according to their structure and etymological origin.
- In some cases, an “ad-hoc” decomposition occurs if the etymological origin and its glyph shape are not consistent, or the etymological origin is not clear, or to avoid defining additional components.
- Characters are not decomposed if it appears in GF 3001-1997 as a component.
- Detached parts can be further decomposed.
- Merely touched parts that do not have overlapping or crossing can be decomposed.
- In some cases, we do not decompose some components to prevent the components from getting too small.
- In some cases, a single component will be distinguished as two different components. This is the concept of *variant* or *related* component.

This set of rules, together with 644 basic components and the set of intermediate components defined, enables us to decompose Chinese characters that appear in the first version ISO 10646 with 20,902 characters, Ext. A in the second version of ISO 10646[1] and Hong Kong Suplementary Character Set [8-9]. The 644 basic components play a very important role because they form all the Chinese characters in our scope.

In order to describe the position relationship amongst components in a character, we have used the 12 Ideographic Description Characters (IDC) in ISO/IEC 10646 Part1:2000 in the range from 2FF0 to 2FFB, and defined an extra IDC “M” (which indicates that a particular component is a basic component and will not be further decomposed), as shown in Table 1. Every character can be decomposed into up to three components depending on the cardinality of the IDC used.

Each *Character* is decomposed according to the following definition:

Character = IDC2 CC(1) CC(2)
| IDC3 CC(1) CC(2) CC(3)
| M
where
IDC2 ∈ (2FF0 – 2FFB)
CC(i) is a set of character components and *i* indicates its position in the sequence

M is a special symbol indicating
Character will not be further
decomposed

By our definition, a *CC* can be formed by three subsets: (1) coded radicals, (2) coded components and ideographs proper, and (3) intermediate components that are not coded in ISO 10646. The intermediate components are maintained by our system only. The decomposition result is stored in the database. Conceptually, every entry in the database can be treated as a Chinese component, having a data structure described above.

2. Decomposition/Formation Algorithms

As mentioned above, the decomposition database only gives information on how a character is decomposed in a minimal way. However, some characters have nested components. For instance, the character “準” can be decomposed into two components: “淮” and “十”, but “淮” being a character can be further decomposed into two components. In order to handle nesting and finding components to the most elementary form(no further decomposition), we have defined the decomposition and formation algorithms. There are mainly two algorithms, one for the decomposition of a character into a set of components(the algorithm is called *Char-to-Compnt*) , another one for the formation of a set of characters from a component (the algorithm is called *Compn-to-Charr*).

```
Let x be the seed (x = starting character for search);
Stop = false
WHILE NOT    stop DO
        IF Struct_Symbol(CD[x]) = “M”
            Stop = True
        ELSE
        LCmp ={ cc[x] ∈    CC }
ENDWHILE
```

Figure 1. Pseudo-code of “Char-to-Compnt”

Both algorithms are very similar. They recursively retrieve all characters/components appearing in the decomposition database by using the characters/components themselves as a seed, but their directions of retrieval are opposite to each other. In the *“Char-to-Compnt”*, the decomposition goes from its current level down, one level at a time, until no more decomposition can be done. Figure 1 the pseudo code of the algorithm for one level only and they can be done recursively to find all components of a character. Table 2 shows the entries related to the character “盟”. Notice that the number of components for “盟” is not two, but 4 because one of the components “明” can be further decomposed into two more components.

Character	IDC	Comp1	Comp2	Comp3
盟	B	明	皿	
明	A	日	月	
皿	M			
日	M			
月	M			

Table 2. Component Entries of character “盟”

On the other hand, the *“Compnt-to-Char”* algorithm searches from its current level up until no more character can be found using the current component. Figure 2 shows the pseudo code of the upward search algorithm where *x* is considered the seed to start the search and the variable contains all characters formed using the current component *x*.

```
Let x be the seed (x = starting component for search);
Stop = false
Char_List ={ x}
WHILE NOT    stop DO
        IF No Change to Char_List
                Stop = True
       ELSE
        FOR each x in Char_List
            Char_List = Char_List ∪{ Char[x]}
       ENDFOR

ENDWHILE
```

Figure 2. Pseudo-code of “Compnt-to-Char”

Character	IDC	Comp1	Comp2	Comp3
口	M			
吾	B	五	口	
語	A	言	吾	
齬	A	齒	吾	
唔	A	口	吾	
…				

Table 3. Example character entries of component “口”

Table 3 shows some of the search results involving the component "口". Note that the result not only find the character "吾", but also the characters using "吾" as components as well.

Further more, due to the fact that there are two IDCs with cardinality of three, the decomposition is not unique. Based Han characters formation rules, some characters should be decomposed into two components first before considering further decomposition. For instance, "鍘" should be decomposed into "金" and "則" whereas "街" should be decomposed into "行" and "圭". However, for upward search we certainly want the character "鍘" to be found if the search component is "鋇". Therefore, in addition to using the most reason decomposition at the first level, we also maintain different decompositions for applications where character formation rule are less important. In other words, we also provided composition and decompositions independent of certain particular character formal rules. Again taking the character "鍘" as an examples, its components should not only be "金" and "則", but also "鋇", "則", "釗" as well as "貝" and " ". In fact, in our system, "鍘" is decomposed into "金", "貝" and " " as shown in Table 4. The *"Char-to-Compnt"* algorithm will take the relative positions of the components into consideration based on the IDC defined in each entry to find other three possible components "鋇", "則" and "釗". This can be done because the combination of "金" and "貝" will form "鋇"; similarly "貝" and " " will form "則";, and "金" and " "will form "釗". Note that in the first two cases of the *OR* clause, "鋇" and "則" will be identified. In the third case of the *OR* clause, the character "釗" will be identified. You may argue the validity of the third case of the *OR* clause, but for the character ""衝", finding the component "行" would be very important.

Character	IDC	Comp1	Comp2	Comp3
鍘	K	金	貝	
鋇	A	金	貝	
則	A	貝		
釗	A	金		

Table 4 An example of handling a character with three components

The basic principle of the algorithm, as shown in Figure 3, is that if we see a character with an IDC {K} or {L}, or an IDC of a character that can be transformed to IDC {K} or {L}, we will try to use its components to form characters.

```
Let x be a Chinese component (x = cc);
Let LCsub be the list of sub-components c;
IF x[structure] = IDC{K} THEN
    LCsub = c : c[structure] = IDC{A}AND
    c[component(1)] = x[component(1)] AND
    c[component(2)] = x[component(2)] or
    c[component(2)] = x[component(2)] AND
    c[component(3)] = x[component(3)] or
    c[component(1)] = x[component(1)] AND
    c[component(3)] = x[component(3)]
END
**the same algorithm works when x[structure] =
IDC{L}, then the result c[structure] will become
IDC{B}
```

Figure 4 Pseudo-code for handling a character with three components

```
Let s be the seed (s = cc);
Let r be the result component;
if s[structure] = IDC{A}
    if s[component(1)][ structure] = IDC{A} then
        r = IDC{K} +
        s[component(1)][component(1)] +
        s[component(1)][component(2)] +
        s[component(2)]
    else if s[component(2)][ structure] = IDC{A}
    then
        r = IDC{K} + s[component(1)] +
        s[component(2)][component(1)] +
        s[component(2)][component(2)]
    end
end
**this algorithm also works when s[structure] =
IDC{B}, then the result structure will become
IDC{L}
```

Figure 4 Pseudo-code of For the Split Step

In many cases, we still want to maintain the characters in the right decomposition, e.g, to decompose them into two components first and then further decompose if needed. Take another character "樹" as an example. Suppose it is only decomposed into two components ("木" and "尌"). This makes the search more complex. In order to simplify the search, we need to go through an internal step which we call the *Split Step* to decompose the character into three components before we allow for component to character search. The pseudo code for the *Split Step* is shown in Figure 4. The generated result is shown in Table 5.

Character	IDC	Comp1	Comp2	Comp3
樹	A	木	尌	
尌	A	壴	寸	
樹	K	木	壴	寸

Table 5. An example Output of the *Split Step*

For some characters like “衝”, the Split Step must consider the component “重” in the middle as an insertion into the character “行”. We use similar handling to decompose “衝” into “彳”, “重” and “亍”, with an IDC {K}. In order to find a character with the component “行” such as “衝” , we need additional algorithm to locate components that are potentially being split to the two sides with an inserted component. We try to decompose a component into two sub-components if their IDC is “A” or “B”. Once we get the two sub-components, we try to make different combinations to see if there are any characters with an IDC {K} or {L} that contain the two sub-components as shown in Figure 5.

```
Let x be a Chinese character (x = cc);
Let Clst be the list of results c;
if x[structure] = IDC{A} then
    Clst = c : c[structure] = IDC{K} and
    ((c[component(1)] = x[component(1)] and
    c[component(2)] = x[component(2)] ) or
    (c[component(2)] = x[component(1)] and
    c[component(3)] = x[component(2)]) or
    (c[component(1)] = x[component(1)] and
    c[component(3)] = x[component(2)]))
end
**this algorithm also works when x[structure] =
IDC{B}, then the result structure will become
IDC{L}
```

Figure 5. Pseudo-code of finding inserted component

3. Performance Evaluation

Since the algorithms have to do excessive search for many combinations in many levels recursively, performance becomes a very important issue especially if we want to make this for public access through the internet. However, since the decomposition is static, it does not need to be done in real time. as the search doesn’t need to be done online, In other words, searching of the same data will always give the same result unless the decomposition rules or algorithms are changed. Consequently, we built two pre-searched tables to store the results of both “*Compnt-to-Char*” algorithm and the “*Char-to-Compnt*”algorithm. Once we have the pre-searched tables, we can totally avoid the recursive search. Instead, the search result can be directly retrieved in a single tuple. This results in much better performance both in terms of usage of CPU time and I/O usage.

Character	Pre-searched result
鍘	鍘 鋇 則 釗 金 貝
語	語 言 吾 五 口
……	

Table 6. Examples of pre-searched results of “*Cha-to-Compnt*”Algorithm

Character	Pre-searched result
口	口 言 吾 語 谷 鰫 靈 … (total 5481 characters)
五	五 吾 語 痦 逜 齬 … (total 44 characters)
……	

Table 7. Examples of pre-searched results of “*Component to Character*”

Table 6 and table 7 shows some samples of the pre-searched tables for the downward search and the upward search, respectively.

Although the advanced control algorithms can retrieve most Chinese characters correctly, they also return some components that do not make much sense. For example, the character “章” has a structure of IDC{B}, and components “立” and “早”. However, when it is eventually decomposed into “立”, “日” and “十”. Using the algorithm “Char-to-Compnt”, the component “辛” will also be returned, even though “辛” has no cognate relationship with the character “章”. We can take into consideration of only a subset of characters that can be split in character formation, such as “行” and “衣”. This way, the insertion components will only be considered for these characters.

4. Conclusion

In this paper, we focus on the algorithms of character decomposition and formation. The results can be used for the standardization of computer fonts, glyphs, or relevant language resources. We have implemented a Chinese Character Search System based on the result of this standardization work. We can use this search system to look for character decomposition or formation results. The system comes with many handy and useful features. It provides a lot of useful information on Chinese characters, such as the code for various encodings, and pronunciations. A stand-alone version is also built. The actual implementation of these algorithms and of the database helps people to get information about Chinese characters very quickly. It further facilitates researchers' work in related areas. For more information on the system, please visit the website http://www.iso10646hk.net.

Acknowledgement

The project is partially supported by the Hong Kong SAR Government(Project code: ITF AF/212/99) and the Hong Kong Polytechnic University(Project Code: Z044).

References

[1] ISO/IEC, "ISO/IEC 10646-1 Information Technology-Universal Multiple-Octet Coded Character Set - Part 1", ISO/IEC, 2000

[2] ISO/IEC, "ISO/IEC 10646-2 Information Technology-Universal Multiple-Octet Coded Character Set - Part 1", ISO/IEC, 2001

[3] John Jenkins, "New Ideographs in Unicode 3.0 and Beyond", Proceedings of the 15th International Unicode Conference C15, San Jose, California, Sept. 1-2, 1999

[4] Dept. of Education(Taiwan), "Dictionary of Chinese Character Variants Version 2", Dept. of Education, Taiwan, 2000

[5] 李學銘(主編),《常用字字形表》,(二零零零年修訂本), 香港教育學院出版, 二零零零年(LEE Hok-ming as Chief Editor, Common Character Glyph Table 2nd Edition, Hong Kong Institute of Education, 2000)

[6] GF3001-1997《國家語言文字工作委員會語言文字規範 信息處理用 GB13000.1 字符集漢字部件規範》, 國家語言文字工作委員會, 1997 年 12 月.

[7] Lu Qin, The Hong Kong Glyph Specifications for ISO 10646's Ideographic Characters. 21st International Unicode Conference, Dublin, Ireland, May 2002

[8] Hong Kong Special Administrative Region Government, "Hong Kong Supplementary Character Set", HKSARG, September 28, 1999

[9] Hong Kong Special Administrative Region Government, "Hong Kong Supplementary Character Set – 2001", HKSARG, December 31, 2001

Efficient Deep Processing of Japanese

Melanie SIEGEL
DFKI GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
siegel@dfki.de

Emily M. BENDER
CSLI Stanford
220 Panama Street
Stanford, CA, 94305-4115, USA
bender@csli.stanford.edu

Abstract

We present a broad coverage Japanese grammar written in the HPSG formalism with MRS semantics. The grammar is created for use in real world applications, such that robustness and performance issues play an important role. It is connected to a POS tagging and word segmentation tool. This grammar is being developed in a multilingual context, requiring MRS structures that are easily comparable across languages.

Introduction

Natural language processing technology has recently reached a point where applications that rely on deep linguistic processing are becoming feasible. Such applications (e.g. message extraction systems, machine translation and dialogue understanding systems) require natural language understanding, or at least an approximation thereof. This, in turn, requires rich and highly precise information as the output of a parse. However, if the technology is to meet the demands of real-world applications, this must not come at the cost of robustness. Robustness requires not only wide coverage by the grammar (in both syntax and semantics), but also large and extensible lexica as well as interfaces to preprocessing systems for named entity recognition, non-linguistic structures such as addresses, etc. Furthermore, applications built on deep NLP technology should be extensible to multiple languages. This requires flexible yet well-defined output structures that can be adapted to grammars of many different languages. Finally, for use in real-world applications, NLP systems meeting the above desiderata must also be efficient.

In this paper, we describe the development of a broad coverage grammar for Japanese that is used in an automatic email response application. The grammar is based on work done in the *Verbmobil* project (Siegel 2000) on machine translation of spoken dialogues in the domain of travel planning. It has since been greatly extended to accommodate written Japanese and new domains.

The grammar is couched in the theoretical framework of Head-Driven Phrase Structure Grammar (HPSG) (Pollard & Sag 1994), with semantic representations in Minimal Recursion Semantics (MRS) (Copestake et al. 2001). HPSG is well suited to the task of multilingual development of broad coverage grammars: It is flexible enough (analyses can be shared across languages but also tailored as necessary), and has a rich theoretical literature from which to draw analyzes and inspiration. The characteristic type hierarchy of HPSG also facilitates the development of grammars that are easy to extend. MRS is a flat semantic formalism that works well with typed feature structures and is flexible in that it provides structures that are under-specified for scopal information. These structures give compact representations of ambiguities that are often irrelevant to the task at hand.

HPSG and MRS have the further advantage that there are practical and useful open-source tools for writing, testing, and efficiently processing grammars written in these formalisms. The tools we are using in this project include the LKB system (Copestake 2002) for grammar development, [incr tsdb()] (Oepen & Carroll 2000) for testing the grammar and tracking changes, and PET (Callmeier 2000), a very efficient HPSG parser, for processing. We also use the ChaSen tokenizer and POS tagger (Asahara & Matsumoto 2000).

While couched within the same general framework (HPSG), our approach differs from that of Kanayama et al (2000). The work described there achieves impressive coverage (83.7% on the EDR corpus of newspaper text) with an underspecified grammar consisting of a small number of lexical entries, lexical types associated with parts of speech, and six underspecified grammar rules. In contrast, our grammar is much larger in terms of the number of lexical entries, the number of grammar rules, and the constraints on both,[1] and takes correspondingly more effort to bring up to that level of coverage. The higher level of detail allows us to output precise semantic representations as well as to use syntactic, semantic and lexical information to reduce ambiguity and rank parses.

1 Japanese HPSG Syntax

The fundamental notion of an HPSG is the sign. A sign is a complex feature structure representing information of different linguistic levels of a phrase or lexical item. The attribute-value matrix of a sign in the Japanese HPSG is quite similar to a sign in the LinGO English Resource Grammar (henceforth ERG) (Flickinger 2000), with information about the orthographical realization of the lexical sign in PHON, syntactic and semantic information in SYNSEM, information about the lexical status in LEX, nonlocal information in NONLOC, head information that goes up the tree in HEAD and information about subcategorization in SUBCAT.

The grammar implementation is based on a system of types. There are 900 lexical types that define the syntactic, semantic and pragmatic properties of the Japanese words, and 188 types that define the properties of phrases and lexical rules. The grammar includes 50 lexical rules for inflectional and derivational morphology and 47 phrase structure rules. The lexicon contains 5100 stem entries. As the grammar is developed for use in applications, it treats a wide range of basic constructions of Japanese. Only some of these phenomena can be described here.

1.1 Subcategorization

The structure of SUBCAT is different from the ERG SUBCAT structure. This is due to differences in subcategorization between Japanese and English. A fundamental difference is the fact that, in Japanese, verbal arguments are frequently omitted. For example, arguments that refer to the speaker, addressee, and other arguments that can be inferred from context are often omitted in spoken language. Additionally, optional verbal arguments can scramble. On the other hand, some arguments are not only obligatory, but must also be realized adjacent to the selecting head.

To account for this, our subcategorization contains the attributes SAT and VAL. The SAT value encodes whether a verbal argument is already saturated (such that it cannot be saturated again), optional or adjacent. VAL contains the agreement information for the argument. When an argument is realized, its SAT value on the mother node is specified as *sat* and its SYNSEM is unified with its VAL value on the subcategorizing head. The VAL value on the mother is *none*. Adjacency must be checked in every rule that combines heads and arguments or adjuncts. This is the *principle of adjacency*, stated as follows:

In a headed phrase, the SUBCAT.SAT value on the non-head daughter must not contain any adjacent arguments. In a head-complement structure, the SUBCAT.SAT value of the head daughter must not contain any adjacent arguments besides the non-head daughter. In a head-adjunct structure, the SUBCAT.SAT value of the head daughter must not contain any adjacent arguments.

1.2 Verbal inflection

Japanese verb stems combine with endings that provide information about honorification, tense, aspect, voice and mode. Inflectional rules for the different types of stems prepare the verb stems for combination with the verbal endings. For example, the verb stem *yomu* must be inflected to *yon* to combine with the past tense ending *da*. Morphological features constrain the

[1] We do also make use of generic lexical entries for certain parts of speech as a means of extending our lexicon. See section 3 below.

combination of stem and ending. In the above example, the inflectional rule changes the *mu* character to the *n* character and assigns the value *nd-morph* to the morphological feature RMORPH-BIND-TYPE. The ending *da* selects for a verbal stem with this value.

Endings can be combined with other endings, as in *-sase-rare-mashi-ta* (causative-potential-honorific-past), but not arbitrarily:

**-sase-mashi-rare-ta*
**-sase-ta-mashi-rare*
-sase-ta
-rare-mashi-ta

This is accounted for with two kinds of rules which realize mutually selected elements. In the combination of stem and ending, the verb stem selects for the verbal ending via the head feature SPEC. In the case of the combination of two verbal endings, the first ending selects for the second one via the head feature MARK. In both cases, the right element subcategorizes for the left one via SUBCAT.VAL.SPR. Using this mechanism, it is possible to control the sequence of verbal endings: Verb stems select verbal endings via SPEC and take no SPR, derivational morphemes (like causative or potential) select tense endings or other derivational morphemes via MARK and subcategorize for verb stems and/or verb endings via SPR (*sase* takes only verb stems), and tense endings take verb stems or endings as SPR and take no MARK or SPEC (as they occur at the end of the sequence).

1.3 Complex Predicates

A special treatment is needed for Japanese verbal noun + light verb constructions. In these cases, a word that combines the qualities of a noun with those of a verb occurs in a construction with a verb that has only marginal semantic information. The syntactic, semantic and pragmatic information on the complex is a combination of the information of the two.

Consider example 1. The verbal noun *benkyou* contains subcategorization information (transitive), as well as semantic information (the *benkyou*-relation and its semantic arguments). The light verb *shi-ta* supplies tense information (past). Pragmatic information can be supplied by both parts of the construction, as in the formal form *o-benkyou shi-mashi-ta*. The rule that licenses this type of combination is the *vn-light-rule*, a subtype of the *head-marker-rule.*

Example 1:
Benkyou shi-ta.
study do-past
'Someone has studied.'

Japanese auxiliaries combine with verbs and provide either aspectual or perspective information or information about honorification. In a verb-auxiliary construction, the information about subcategorization is a combination of the SUBCAT information of verb and auxiliary, depending on the type of auxiliary. The rule responsible for the information combination in these cases is the *head-specifier-rule*. We have three basic types of auxiliaries. The first type is aspect auxiliaries. These are treated as raising verbs, and include such elements as *iru* (roughly, progressive) and *aru* (roughly, perfective), as can be seen in example 2. The other two classes of auxiliaries provide information about perspective or the point of view from which a situation is being described. Both classes of auxiliaries add a *ni* (dative) marked argument to the argument structure of the whole predicate. The classes differ in how they relate their arguments to the arguments of the verb. One class (including *kureru* 'give'; see example 3) are treated as subject control verbs. The other class (including *morau* 'receive', see example 4) establishes a control relation between the *ni*-marked argument and the embedded subject.

Example 2:
Keeki wo tabe-te iru.
cake ACC eat progressive
'Someone is eating cake.'

Example 3:
Sensei wa watashi ni hon wo
teacher TOP I DAT book ACC
katte kure-ta.
buy give-past
'The teacher bought me a book.'

Example 4:
Watashi ga sensei ni hon wo
I NOM teacher DAT book ACC
katte morat-ta.
buy get-past
'The teacher bought me a book.'

1.4 Particles in a type hierarchy

The careful treatment of Japanese particles is essential, because they are the most frequently occurring words and have various central functions in the grammar. It is difficult, because one particle can fulfill more than one function and they can co-occur, but not arbitrarily. The Japanese grammar thus contains a type hierarchy of 44 types for particles. See Siegel (1999) for a more detailed description of relevant phenomena and solutions.

1.5 Numeral Expressions

Number names, such as *sen kyuu hyaku juu* '1910' constitute a notable exception to the general head-final pattern of Japanese phrases. We found Smith's (1999) head-medial analysis of English number names to be directly applicable to the Japanese system as well (Bender 2002). This analysis was easily incorporated into the grammar, despite the oddity of head positioning, because the type hierarchy of HPSG is well suited to express the partial generalizations that permeate natural language.

On the other hand, number names in Japanese contrast sharply with number names in English in that they are rarely used without a numeral classifier.

Example 5:
*Juu *(hiki no) neko ga ki-ta.*
`ten CL GEN cat NOM arrive-past`
'Ten cats arrived.'

The grammar provides for 'true' numeral classifiers like *hon*, *ko*, and *hiki*, as well as formatives like *en* 'yen' and *do* 'degree' which combine with number names just like numeral classifiers do, but never serve as numeral classifiers for other nouns. In addition, there are a few non-branching rules that allow bare number names to surface as numeral classifier phrases with specific semantic constraints.

1.6 Pragmatic information

Spoken language and email correspondence both encode references to the social relation of the dialogue partners. Utterances can express social distance between addressee and speaker and third persons. Honorifics can even express respect towards inanimates. Pragmatic information is treated in the `CONTEXT` layer of the complex signs. Honorific information is given in the `CONTEXT.BACKGROUND` and linked to addressee and speaker anchors.

The expression of empathy or in-group vs. out-group is quite prevalent in Japanese. One means of expressing empathy is the perspective auxiliaries discussed above. For example, two auxiliaries meaning roughly 'give' (*ageru* and *kureru*) contrast in where they place the empathy. In the case of *ageru*, it is with the giver. In the case of *kureru*, it is with the recipient. We model this within the sign by positing a feature `EMPATHY` within `CONTEXT` and linking it to the relevant arguments' indices.

2 Japanese MRS Semantics

In the multilingual context in which this grammar has been developed, a high premium is placed on parallel and consistent semantic representations between grammars for different languages. Ensuring this parallelism enables the reuse of the same downstream technology, no matter which language is used as input. Integrating MRS representations parallel to those used in the ERG into the Japanese grammar took approximately 3 months. Of course, semantic work is on-going, as every new construction treated needs to be given a suitable semantic representation. For the most part, semantic representations developed for English were straightforwardly applicable to Japanese. This section provides a brief overview of those cases where the Japanese constructions we encountered led to innovations in the semantic representations and/or the correspondence between syntactic and semantic structures. Due to space limitations, we discuss these analyses in general terms and omit technical details.

2.1 Nominalization and Verbal Nouns

Nominalization is of course attested in English and across languages. However, it is much more prevalent in Japanese than in English, primarily because of verbal nouns. As noted in Section 1.3 above, a verbal noun like *benkyou* 'study' can appear in syntactic contexts requiring nouns, or, in combination with a light verb, in contexts requiring verbs. One possible analysis would provide two separate lexical entries, one with nominal and one with verbal semantics. However, this would not only be redundant (missing the systematic relationship between these uses of verbal nouns) but would also contradict the intuition that even in its nominal use, the arguments of *benkyou* are still present.

Example 6:
Nihongo no benkyou wo hajimeru.
`Japanese GEN study ACC begin`
'Someone begins the study of Japanese.'

In order to capture this intuition, we opted for an analysis that essentially treats verbal nouns as underlyingly verbal. The nominal uses are produced by a lexical rule which nominalizes the verbal nouns. The semantic effect of this rule is to provide a nominal relation which introduces a variable which can in turn be bound by quantifiers. The nominal relation subordinates the original verbal relation supplied by the verbal noun. The rule is lexical as we have not yet found any cases where the verb's arguments are clearly filled by phrases in the syntax. If they do appear, it is with genitive marking (e.g., *nihongo no* in the example above). In order to reduce ambiguity, we leave the relationship between these genitive marked NPs and the nominalized verbal noun underspecified. There is nothing in the syntax to disambiguate these cases, and we find that they are better left to downstream processing, where there may be access to world knowledge.

2.2 Numeral Classifiers

As noted in Section1.5, the internal syntax of number names is surprisingly parallel between English and Japanese, but their external syntax differs dramatically. English number names can appear directly as modifiers of NPs and are treated semantically as adjectives in the ERG. Japanese number names can only modify nouns in combination with numeral classifiers. In addition, numeral classifier phrases can appear in NP positions (akin to partitives in English). Finally, some numeral-classifier-like elements do not serve the modifier function but can only head phrases that fill NP positions.

This constellation of facts required the following innovations: a representation of numbers that doesn't treat them as adjectives (in MRS terms, a feature structure without the ARG feature), a representation of the semantic contribution of numeral classifiers (a relation between numbers and the nouns they modify, this time with an ARG feature), and a set of rules for promoting numeral classifier phrases to NPs that contribute the appropriate nominal semantics (underspecified in the case of ordinary numeral classifiers or specific in the case of words like *en* 'yen').

2.3 Relative Clauses and Adjectives

The primary issue in the analysis of relative clauses and adjectives is the possibility of extreme ambiguity, due to several intersecting factors: Japanese has rampant pro-drop and does not have any relative pronouns. In addition, a head noun modified by a relative clause need not correspond to any gap in the relative clause, as shown by examples like the following (Matsumoto 1997):

Example 7:
atama ga yoku naru hon
`head NOM better become book`
'a book that makes one smarter'

Therefore, if we were to posit an attributive adjective + noun construction (distinct from the relative clause + noun possibility) we would have systematic ambiguities for NPs like *akai hon* ('red book'), ambiguities which could never be resolved based on information in the sentence. Instead, we have opted for a relative clause analysis of any adjective + noun combination in which the adjective could potentially be used predicatively. Furthermore, because of gapless relative clauses like the one cited above, we have opted for a non-extraction analysis of relative clauses.[2]

Nonetheless, the well-formedness constraints on MRS representations require that there be

[2] There is in fact some linguistic evidence for extraction in some relative clauses in Japanese (see e.g., Baldwin 2001). However, we saw no practical need to allow for this possibility in our grammar, and particularly not one that would justify the increase in ambiguity. There is also evidence that some adjectives are true attributives and cannot be used predicatively (Yamakido 2000). These are handled by a separate adjective + noun rule restricted to just these cases.

some relationship between the head noun and the relative clause. We picked the topic relation for this purpose (following Kuno 1973). The topic relation is introduced into the semantics by the relative clause rule. As with main clause topics (which we also give a non-extraction analysis), we rely on downstream anaphora resolution to refine the relationship.

2.4 Summary

For the most part, semantic representations and the syntax-semantic interface already worked out in the ERG were directly applicable to the Japanese grammar. In those cases where Japanese presented problems not yet encountered (or at least not yet tackled) in English, it was fairly straightforward to work out suitable MRS representations and means of building them up. Both of these points illustrate the cross-linguistic validity and practical utility of MRS representations.

3 Integration of a Morphological Analyzer

As Japanese written text does not have word segmentation, a preprocessing system is required. We integrated ChaSen (Asahara & Matsumoto 2000), a tool that provides word segmentation as well as POS tags and morphological information such as verbal inflection. As the lexical coverage of ChaSen is higher than that of the HPSG lexicon, default part-of-speech entries are inserted into the lexicon. These are triggered by the part-of-speech information given by ChaSen, if there is no existing entry in the lexicon. These specific default entries assign a type to the word that contains features typical to its part-of-speech. It is therefore possible to restrict the lexicon to those cases where the lexical information contains more than the typical information for a certain part-of-speech. This default mechanism is often used for different kinds of names and 'ordinary' nouns, but also for adverbs, interjections and verbal nouns (where we assume a default transitive valence pattern).[3]

The ChaSen lexicon is extended with a domain-specific lexicon, containing, among others, names in the domain of banking.

For verbs and adjectives, ChaSen gives information about stems and inflection that is used in a similar way. The inflection type is translated to an HPSG type. These types interact with the inflectional rules in the grammar such that the default entries are inflected just as 'known' words would be.

In addition to the preprocessing done by ChaSen, an additional (shallow) preprocessing tool recognizes numbers, date expressions, addresses, email addresses, URLs, telephone numbers and currency expressions. The output of the preprocessing tool replaces these expressions in the string with placeholders. The placeholders are parsed by the grammar using special placeholder lexical entries.

4 Robustness and Performance Issues

The grammar is aimed at working with real-world data, rather than at experimenting with linguistic examples. Therefore, robustness and performance issues play an important role. While grammar development is carried out in the LKB (Copestake 2002), processing (both in the application domain and for the purposes of running test suites) is done with the highly efficient PET parser (Callmeier 2000). Figures 1 and 2 show the performance of PET parsing of hand-made and real data, respectively.

Phenomenon	***items***	***etasks***	***filter***	***edges***	**first**	***total***	***tcpu***	***gc***	***space***
	#	***Ø***	***%***	***Ø***	**Ø (s)**	***Ø (s)***	***Ø (s)***	***Ø (s)***	***Ø (kb)***
Total	***742***	***946***	***95.7***	***303***	***0.06***	***0.11***	***0.11***	***0***	***833***

Fig.1 Performance parsing banking data, generated by [incr tsdb()]

Phenomenon	***items***	***etasks***	***filter***	***edges***	**first**	***total***	***tcpu***	***tgc***	***space***
	#	***Ø***	***%***	***Ø***	**Ø (s)**	***Ø (s)***	***Ø (s)***	***Ø (s)***	***Ø (kb)***
Total	***316***	***2020***	***96.5***	***616***	***0.23***	***0.26***	***0.26***	***0***	***1819***

Fig.2 Performance parsing document request data, generated by [incr tsdb()]

One characteristic of real-world data is the variety of punctuation marks that occur and the potential for ambiguity that they bring. In our grammar, certain punctuation marks are given lexical entries and processed by grammar rules. Take, for example, quotation marks. Ignoring them (as done in most development-oriented grammars and smaller grammars), leads to a significant loss of structural information:

[3] Kanayama et al. (2000) use a similar mechanism for most words. They report only 105 grammar-inherent lexical entries.

Example 8:

"Botan	*wo*	*osu"*	*to*	*it-ta*
button	ACC	push	COMPL	say-past

'Someone said: "push the button. "'

The formative *to* is actually ambiguous between a complementizer and a conjunction. Since the phrase before *to* is a complete sentence, this string is ambiguous if one ignores the quotation marks. With the quotation marks, however, only the complementizer *to* is possible. Given the high degree of ambiguity inherent in broad-coverage grammars, we have found it extremely useful to parse punctuation rather than ignore it.

The domains we have been working on (like many others) contain many date and number expressions. While a shallow tool recognizes general structures, the grammar contains rules and types to process these.

Phenomena occurring in semi-spontaneous language (email correspondence), such as interjections (e.g. *maa* 'well'), contracted verb forms (e.g. *tabe-chatta* < *tabete-shimatta* '(someone) ate it all up'), fragmentary sentences (e.g. *bangou: 1265* 'number: 1265') and NP fragments (e.g. *bangou*? 'number?') must be covered as well as the 'ordinary' complete sentences found in more carefully edited text. Our grammar includes types, lexical entries, and grammar rules for dealing with such phenomena.

Perhaps the most important performance issue for broad coverage grammars is ambiguity. At one point in the development of this grammar, the average number of readings doubled in two months of work. We currently have two strategies for addressing this problem: First, we include a mechanism into the grammar rules that chooses left-branching rules in cases of compounds, genitive modification and conjuncts, as we don't have enough lexical-semantic information represented to choose the right dependencies in these cases.[4] Secondly, we use a mechanism for hand-coding reading preferences among rules and lexical entries. Restrictions like *head-complement preferred to head-adjunct* are quite obvious. Others require domain-specific mechanisms that shall be subject of further work. Stochastic disambiguation methods being developed for the ERG by the Redwoods project at Stanford University (Oepen et al. 2002) should be applicable to this grammar as well.

5 Evaluation

The grammar currently covers 93.4% of constructed examples for the banking domain (747 sentences) and 78.2% of realistic email correspondence data (316 sentences), concerning requests for documents. During three months of work, the coverage in the banking domain increased 48.49%. The coverage of the document request data increased 51.43% in the following two weeks.

Phenomenon	***total items #***	***positive items #***	***word string %***	***lexical items Ø***	***parser analyses Ø***	***total results #***	***overall coverage %***
Total	***747***	***747***	***101***	***75.24***	***6.54***	***698***	***93.4***

Fig.3 Coverage of banking data, generated by [incr tsdb()]

Phenomenon	***total items #***	***positive items #***	***word string %***	***lexical items Ø***	***parser analyses Ø***	***total results #***	***overall coverage %***
Total	***316***	***316***	***1.00***	***83.90***	***39.91***	***247***	***78.2***

Fig.4 Coverage of document request data, generated by [incr tsdb()]

We applied the grammar to unseen data in one of the covered domains, namely the FAQ site of a Japanese bank. The coverage was 61%. 91.2% of the parses output were associated with all well-formed MRSs. That means that we could get correct MRSs in 55.61% of all sentences.

Conclusion

We described a broad coverage Japanese grammar, based on HPSG theory. It encodes syntactic, semantic, and pragmatic information. The grammar system is connected to a morphological analysis system and uses default entries for words unknown to the HPSG lexicon.

Some basic constructions of the Japanese grammar were described. As the grammar is aimed at working in applications with real-world data, performance and robustness issues are important.

The grammar is being developed in a multilingual context, where much value is

[4] Consider, for example, genitive modification: The semantic relationship between modifier and modifiee is dependent on their semantic properties: *toukyou no kaigi* - 'the meeting in Tokyo', *watashi no hon* - 'my book'. More lexical-semantic information is needed to choose the correct parse in more complex structures, such as in *watashi no toukyou no imooto* – 'My sister in Tokyo'.

placed on parallel and consistent semantic representations. The development of this grammar constitutes an important test of the cross-linguistic validity of the MRS formalism.

The evaluation shows that the grammar is at a stage where domain adaptation is possible in a reasonable amount of time. Thus, it is a powerful resource for linguistic applications for Japanese.

In future work, this grammar could be further adapted to another domain, such as the EDR newspaper corpus (including a headline grammar). As each new domain is approached, we anticipate that the adaptation will become easier as resources from earlier domains are reused. Initial evaluation of the grammar on new domains and the growth curve of grammar coverage should bear this out.

References

Asahara, Masayuki and Yuji Matsumoto (2000). Extended Models and Tools for High-performance Part-of-speech Tagger. In *Proceedings of the 18th International Conference on Computational Linguistics, Coling 2000*, 21-27. Saarbrücken, Germany.

Baldwin, Timothy (2001). *Making Lexical Sense of Japanese-English Machine Translation: A Disambiguation Extravaganza*. PhD thesis, Tokyo Institute of Technology.

Bender, Emily M. (2002). *Number Names in Japanese: A Head-Medial Construction in a Head-Final Language.* Paper presented at the 76th annual meeting of the LSA, San Francisco.

Callmeier, Ulrich (2000). PET — a platform for experimentation with efficient HPSG processing techniques. *Journal of Natural Language Engineering, Special Issue on Efficient Processing with HPSG: Methods, Systems, Evaluation*, pages 99-108.

Copestake, Ann (2002). *Implementing Typed Feature-Structure Grammars*. Stanford: CSLI.

Copestake, Ann, Alex Lascarides, and Dan Flickinger (2001). An Algebra for Semantic Construction in Constraint-based Grammars. *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001), Toulouse, France.*

Flickinger, Dan (2000). On Building a More Efficient Grammar by Exploiting Types. *Natural Language Engineering 6(1) (Special Issue on Efficient Processing with HPSG)*, pages 15-28.

Kanayama, Hiroshi, Kentaro Torisawa, Yutaka Mitsuishi and Jun'ichi Tsujii (2000). A Hybrid Japanese Parser with Hand-crafted Grammar and Statistics. In *Proceedings of the 18th International Conference on Computational Linguistics, Coling 2000*. Saarbrücken, Germany.

Kuno, Susumu (1973). *The Structure of the Japanese Language.* Cambridge, MA: The MIT Press.

Matsumoto, Yoshiko (1997). *Noun-Modifying Constructions in Japanese: A Frame Semantic Approach*. John Benjamins.

Oepen, Stephan and John Carroll (2000). Performance Profiling for Parser Engineering. *Journal of Natural Language Engineering, Special Issue on Efficient Processing with HPSG: Methods, Systems, Evaluation*, pages 81-97.

Oepen, Stephan, Kristina Toutanova, Stuart Shieber, Chris Manning, Dan Flickinger and Thorsten Brants (2002). The LinGO Redwoods Treebank. Motivation and Preliminary Applications. In *Proceedings of the 19th International Conference on Computational Linguistics, Coling 2002. Tapei, Taiwan. .*

Pollard, Carl and Ivan A. Sag (1994). *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Siegel, Melanie (1999). The Syntactic Processing of Particles in Japanese Spoken Language. In: Wang, Jhing-Fa and Wu, Chung-Hsien (eds.): *Proceedings of the 13th Pacific Asia Conference on Language, Information and Computation*, Taipei 1999.

Siegel, Melanie (2000). HPSG Analysis of Japanese. In: W. Wahlster (ed.): *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer Verlag.

Smith, Jeffrey D. (1999). English number names in HPSG. In Gert Webelhuth, Andreas Kathol, and Jean-Pierre Koenig (eds.), *Lexical and Constructional Aspects of Linguistic Explanation*. Stanford: CSLI. 145-160.

Yamakido, Hiroko (2000). Japanese attributive adjectives are not (all) relative clauses. In Roger Billerey and Brook Danielle Lillehaugen (eds.), *WCCFL 19: Proceedings of the 19th West Coast Conference on Formal Linguistics.* Somerville, MA: Cascadilla Press. 588-602.

Urdu and the Parallel Grammar Project

Miriam Butt
Cent. for Computational Linguistics
UMIST
PO Box 88
Manchester M60 1QD GB
mutt@csli.stanford.edu

Tracy Holloway King
Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
thking@parc.com

Abstract

We report on the role of the Urdu grammar in the Parallel Grammar (ParGram) project (Butt et al., 1999; Butt et al., 2002).[1] The ParGram project was designed to use a single grammar development platform and a unified methodology of grammar writing to develop large-scale grammars for typologically different languages. At the beginning of the project, three typologically similar European grammars were implemented. The addition of two Asian languages, Urdu and Japanese, has shown that the basic analysis decisions made for the European languages can be applied to typologically distinct languages. However, the Asian languages required the addition of a small number of new standard analyses to cover constructions and analysis techniques not found in the European languages. With these additional standards, the ParGram project can now be applied to other typologically distinct languages.

1 Introduction

In this paper, we report on the role of the Urdu grammar in the Parallel Grammar (ParGram) project (Butt et al., 1999; Butt et al., 2002). The ParGram project originally focused on three closely related European languages: English, French, and German. Once grammars for these languages were established, two Asian languages were added: Japanese and Urdu.[2] Both grammars have been successfully integrated into the project. Here we discuss the Urdu grammar and what special challenges it brought to the ParGram project. We are pleased to report that creating an Urdu grammar within the ParGram standards has been possible and has led to typologically useful extensions to the project.

The ParGram project uses the XLE parser and grammar development platform (Maxwell and Kaplan, 1993) to develop deep grammars for six languages. All of the grammars use the Lexical-Functional Grammar (LFG) formalism which produces c(onstituent)-structures (trees) and f(unctional)-structures (AVMs) as syntactic analyses.

LFG assumes a version of Chomsky's Universal Grammar hypothesis, namely that all languages are governed by similar underlying structures. Within LFG, f-structures encode a language universal level of analysis, allowing for cross-linguistic parallelism. The ParGram project aims to test the LFG formalism for its universality and coverage limitations and to see how far parallelism can be maintained across languages. Where possible, the analyses produced for similar constructions in each language are parallel. This parallelism requires a standard for linguistic analysis. In addition, the LFG theory itself limits the set of possible analyses, thus restricting the possible analyses to choose from. The standardization of the analyses has the computational advantage that the grammars can be used in similar applications, and it can simplify cross-language applications (Frank, 1999).

The conventions developed within the ParGram grammars are extensive. The ParGram project dictates not only the form of the features used in the grammars, but also the types of analyses that are chosen for constructions. In addition, the XLE platform necessarily restricts how the grammars can be written. In all cases, the Urdu grammar has successfully, and straightforwardly, incorporated the standards that were originally designed for the European languages. In addition, it has contributed to the formulation of new standards of analysis. Below we discuss several aspects of this: morphology, lexicon, and grammar development for the Urdu grammar within the ParGram project.

[1] We would like to thank Mary Dalrymple, Ron Kaplan, Hiroshi Masuichi, and Tomoko Ohkuma for their comments.

[2] Norwegian was also added at this time.

2 Morphology

The grammars in the ParGram project depend on finite-state morphologies as input (Beesley and Karttunen, 2002). Without this type of resource, it is difficult to build large-scale grammars, especially for languages with substantial morphology. For the original three languages, such morphologies were readily available. As they had been developed for information extraction applications instead of deep grammar applications, there were some minor problems, but the coverage of these morphologies is excellent. An efficient, broad-coverage morphology was also available for Japanese (Asahara and Matsumoto, 2000) and was integrated into the grammar. This has aided in the Japanese grammar rapidly achieving broad coverage. It has also helped control ambiguity because in the case of Japanese, the morphology determines the part of speech of each word in the string with very little ambiguity.

While some morphological analyzers already exist for Hindi,[3] e.g., as part of the tools developed at the Language Technologies Research Centre (LTRC), IIT Hyderabad (http://www.iiit.net/ltrc/index.html), they are not immediately compatible with the XLE grammar development platform, nor is it clear that the morphological analyses they produce conform to the standards and methods developed within the ParGram project. As such, part of the Urdu project is to build a finite-state morphology that will serve as a resource to the Urdu grammar and could be used in other applications.

The development of the Urdu morphology involves a two step process. The first step is to determine the morphological class of words and their subtypes in Urdu. Here we hope to use existing resources and lexicons. The morphological paradigms which yield the most efficient generalizations from an LFG perspective must be determined. Once the basic paradigms and morphological classes have been identified, the second step is to enter all words in the language with their class and subtype information. These steps are described below. Currently we are working on the first step; grant money is being sought for further development.

The finite-state morphologies used in the ParGram project associate surface forms of words with a canonical form (a lemma) and a series of morphological tags that provide grammatical information about that form. An example for English is shown in (1) and for Urdu in (2).

(1)	pushes:	push +Verb +Pres +3sg push +Noun +Pl
(2)	bOlA	bOl +Verb +Perf +Masc +Sg

(1) states the English surface form *pushes* can either be the third singular form of the verb *push* or the plural of the noun *push*. (2) states that the Urdu surface form *bOlA* is the perfect masculine singular form of the verb *bOl*.

The first step of writing a finite-state morphology for Urdu involves determining which tags are associated with which surface forms. As can be seen from the above examples, determining the part of speech (e.g., verb, noun, adjective) is not enough for writing deep grammars. For verbs, tense, aspect, and agreement features are needed. For nouns, number and gender information is needed, as well as information as to whether it is a common or proper noun. Furthermore, for a number of problematic morphological phenomena such as oblique inflection on nominal forms or default agreement on verbs, the most efficient method of analyzing this part of the morphology-syntax interface must be found (Butt and Kaplan, 2002).

After having determined the tag ontology, the patterns of how the surface forms map to the stem-tag sets must be determined. For example, in English the stem-tag set *dog +Noun +Pl* corresponds to the surface form *dogs* in which an *s* is added to the stem, while *box +Noun +Pl* corresponds to *boxes* in which an *es* is added. At this point in time, the basic tag set for Urdu has been established. However, the morphological paradigms that correspond to these tag combinations have not been fully explored.

Once the basic patterns are determined, the second stage of the process begins. This stage involves greatly increasing the coverage of the morphology by adding in all the stems in Urdu and marking them for which set of tags and surface forms they appear with. This is a very large task. However, by using frequency lists for the language and existing lexicons,[4] the most common words can be added first to obtain a major gain in coverage.

In addition, a guesser can be added to guess words that the morphology does not yet recognize (Chanod

[3] An on-line morphological analyzer is available at: http://ccat.sas.upenn.edu/plc/tamilweb/hindi.html

[4] A web search on `Hindi dictionary` results in several promising sites.

and Tapanainen, 1995). This guessing is based on the morphological form of the surface form. For example, if a form ending in *A* is encountered and not recognized, it could be considered a perfect masculine singular form, similar to *bOlA* in (2).

3 Lexicon

One advantage of the fact that the XLE system incorporates large finite-state morphologies is that the lexicons for the languages can then be relatively small. This is because lexicons are not needed for words whose syntactic lexical entry can be determined based on their morphological analysis. This is particularly true for nouns, adjectives, and adverbs.

Consider the case of nouns. The Urdu morphology provides the following analysis for the proper noun *nAdyA*.

(3) nAdyA +Noun +Name +Fem

The tags provide the information that it is a noun, in particular a type of proper noun (Name), and is feminine. The lexical entries for the tags can then provide the grammar with all of the features that it needs to construct the analysis of *nAdyA*; this resulting f-structure analysis is seen in Figures 2 and 4. Thus, *nAdyA* itself need not be in the lexicon of the grammar because it is already known to the morphological analyzer.

Items whose lexical entry cannot be predicted based on the morphological tags need explicit lexical entries. This is the case for items whose subcategorization frames are not predictable, primarily for verbs. Currently, the Urdu verb lexicon is hand constructed and only contains a few verbs, generally one for each subcategorization frame for use in grammar testing. To build a broad-coverage Urdu grammar, a more complete verb lexicon will be needed. To provide some idea of scale, the current English verb lexicon contains entries for 9,652 verbs; each of these has an average of 2.4 subcategorization frames; as such, there are 23,560 verb-subcategorization frame pairs. However, given that Urdu employs productive syntactic complex predicate formation for much of its verbal predication, the verb lexicon for Urdu will be smaller than its English counterpart. On the other hand, writing grammar rules for the productive combinatorial possibilities between adjectives and verbs (e.g., *sAf karnA* 'clean do'='clean'), nouns and verbs (e.g., *yAd karnA* 'memory do'='remember') and verbs and verbs (e.g., *kHA lEnA* 'eat take'='eat up') is anticipated to require significant effort.

There are a number of ways to obtain a broad-coverage verb lexicon. One is to extract the information from an electronic dictionary. This does not exist for Urdu, as far as we are aware. Another is to extract it from Urdu corpora. Again, these would have to be either collected or created as part of the grammar development project. A final way is to enter the information by hand, depending on native speaker knowledge and print dictionaries; this option is very labor intensive. Fortunately, work is being done on verb subcategorization frames in Hindi.[5] We plan to incorporate this information into the Urdu grammar verb lexicon.

4 Grammar

The current Urdu grammar is relatively small, comprising 25 rules (left-hand side categories) which compile into a collection of finite-state machines with 106 states and 169 arcs. The size of the other grammars in the ParGram project are shown in (4) for comparison.

(4)

Language	Rules	States	Arcs
German	444	4883	15870
English	310	4935	13268
French	132	1116	2674
Japanese	50	333	1193
Norwegian	46	255	798
Urdu	25	106	169

It is our intent to drastically expand the Urdu grammar to provide broad-coverage on standard (grammatical, written) texts. The current size of the Urdu grammar is not a reflection of the difficulty of the language, but rather of the time put into it. Like the Japanese and Norwegian grammars, it is less than two years in development, compared with seven years[6] for the English, French, and German grammars. However, unlike the Japanese and Norwegian grammars, there has been no full-time grammar writer on the Urdu grammar. Below we discuss the Urdu grammar analyses and how they fit into the ParGram project standardization requirements.

Even within a linguistic formalism, LFG for ParGram, there is often more than one way to ana-

[5] One significant effort is the Hindi Verb Project run by Prof. Alice Davison at the University of Iowa; further information is available via their web site.

[6] Much of the effort in the initial years went into developing the XLE platform and the ParGram standards. Due to these initial efforts, new grammars can be developed more quickly.

lyze a construction. Moreover, the same theoretical analysis may have different possible implementations in XLE. These solutions often differ in efficiency or conceptual simplicity. Whenever possible, the ParGram grammars choose the same analysis and the same technical solution for equivalent constructions. This was done, for example, with imperatives. Imperatives are assigned a null pronominal subject within the f-structure and a feature indicating that they are imperatives.

Parallelism, however, is not maintained at the cost of misrepresenting the language. Situations arise in which what seems to be the same construction in different languages cannot have the same analysis. An example of this is predicate adjectives (e.g., *It is red.*). In English, the copular verb is considered the syntactic head of the clause, with the pronoun being the subject and the predicate adjective being an XCOMP. However, in Japanese, the adjective is the main predicate, with the pronoun being the subject. As such, these constructions receive nonparallel analyses.

Urdu contains several syntactic constructions which find no direct correlate in the European languages of the ParGram project. Examples are correlative clauses (these are an old Indo-European feature which most modern European languages have lost), extensive use of complex predication, and rampant pro-drop. The ability to drop arguments is not correlated with agreement or case features in Urdu, as has been postulated for Italian, for example. Rather, pro-drop in Urdu correlates with discourse strategies: continuing topics and known background information tend to be dropped. Although the grammars do not encode discourse information, the Japanese grammar analyzes pro-drop effectively via technical tools made available by the grammar development platform XLE. The Urdu grammar therefore anticipates no problems with pro-drop phenomena.

In addition, many constructions which are stalwarts of English syntax do not exist in Asian languages. Raising constructions with *seem*, for example, find no clear correlate in Urdu: the construction is translated via a psych verb in combination with a *that*-clause. This type of non-correspondence between European and South Asian languages raises challenges of how to determine parallelism across analyses. A similar example is the use of expletives (e.g., *There is a unicorn in the garden.*) which do not exist in Urdu.

4.1 Existing Analysis Standards

While Urdu contains syntactic constructions which are not mirrored in the European languages, it shares many basic constructions, such as sentential complementation, control constructions, adjective-noun agreement, genitive specifiers, etc. The basic analysis of these constructions was determined in the initial stage of the ParGram project in writing the English, French, and German grammars. These analysis decisions have not been radically changed with the addition of two typologically distinct Asian languages, Urdu and Japanese.

The parallelism in the ParGram project is primarily across the f-structure analyses which encode predicate-argument structure and other features that are relevant to syntactic analysis, such as tense and number.[7] A sample analysis for the sentence in (5) is shown in Figures 1 and 2.

(5) nAdyA kA kuttA AyA
Nadya Gen.M.Sg dog.Nom come-Perf.M.Sg
'Nadya's dog came.'

The Urdu f-structure analysis of (5) is similar to that of its English equivalent. Both have a PRED for the verb which takes a SUBJ argument at the top level f-structure. This top level structure also has TNS-ASP features encoding tense and aspect information, as well as information about the type of sentence (STMT-TYPE) and verb (VTYPE); these same features are found in the English structure. The analysis of the subject is also the same, with the possessive being in the SPEC POSS and with features such as NTYPE, NUM, and PERS. The sentence in (5) involves an intransitive verb and a noun phrase with a possessive; these are both basic constructions whose analysis was determined before the Urdu grammar was written. Yet, despite the extensive differences between Urdu and the European languages—indeed, the agreement relations between the genitive and the head noun are complex in Urdu but not in English—there was no problem using the standard analysis for the Urdu construction.

4.2 New Analysis Standards

Analyses of new constructions have been added for constructions found in the new project languages.

[7] The c-structures are less parallel in that the languages differ significantly in their word orders. Japanese and Urdu are SOV while English is SVO. However, the standards for naming the nodes in the trees and the types of constituents formed in the trees, such as NPs, are similar.

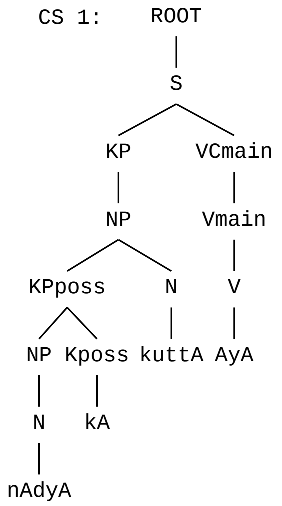
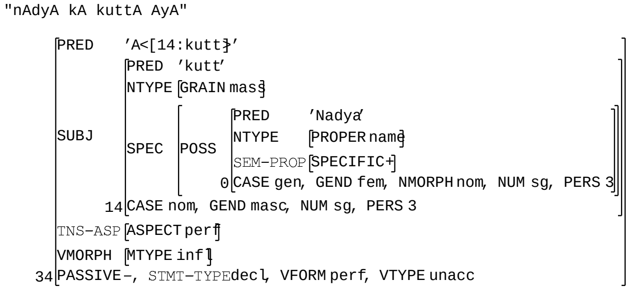


Figure 1: C-structure tree for (5)



```
"nAdyA kA kuttA AyA"

   PRED     'A<[14:kutt]>'
            PRED  'kutt'
            NTYPE [GRAIN mass]
                          PRED      'Nadya'
   SUBJ                   NTYPE     [PROPER name]
            SPEC   POSS
                          SEM-PROP  [SPECIFIC +]
                        0 CASE gen,  GEND fem,  NMORPH nom,  NUM sg,  PERS 3
          14 CASE nom,  GEND masc,  NUM sg,  PERS 3
   TNS-ASP  [ASPECT perf]
   VMORPH   [MTYPE infl]
34 PASSIVE -,  STMT-TYPE decl,  VFORM perf,  VTYPE unacc
```

Figure 2: F-structure AVM for (5)

These analyses have not only established new standards within the ParGram project, but have also guided the development of the XLE grammar development platform. Consider the analysis of case in Urdu. Although the features used in the analysis of case were sufficient for Urdu, there was a problem with implementing it. In Urdu, the case markers constrain the environments in which they occur (Butt and King, to appear). For example, the ergative marker *ne* only occurs on subjects. However, not all subjects are ergative. To the contrary, subjects can occur in the ergative, nominative, dative, genitive, and instrumental cases. Similarly, direct objects can be marked with (at least) an accusative or nominative, depending on the semantics of the clause. Minimal pairs such as in (6) for subjects and (7) for objects suggest a *constructive* (Nordlinger, 1998) approach to case.

(6) a. rAm kHÃs-A
Ram.Nom cough-Perf.M.Sg
'Ram coughed.'

b. rAm nE kHÃs-A
Ram=Erg cough-Perf.M.Sg
'Ram coughed (purposefully).'

(7) a. nAdyA nE gArI calAyI
Nadya=Erg car.Nom drive-Perf.F.Sg
hai
be.Pres.3.Sg
'Nadya has driven a car.'

b. nAdyA nE gArI kO calAyA
Nadya=Erg car=Acc drive-Perf.M.Sg
hai
be.Pres.3.Sg
'Nadya has driven the car.'

We therefore designed the lexical entries for the case markers so that they specify information about what grammatical relations they attach to and what semantic information is needed in the clausal analysis. The lexical entry for the ergative case, for example, states that it applies to a subject.

These statements require inside-out functional uncertainty (Kaplan, 1988) which had not been used in the other grammars. Inside-out functional uncertainty allows statements about the f-structure that contains an item. The lexical entry for *nE* is shown in (8).

(8) nE K @(CASE erg) line 1
(SUBJ ($) ^) line 2
@VOLITION line 3

In (8), the K refers to the part of speech (a case clitic). Line 1 calls a template that assigns the CASE feature the value erg; this is how case is done in the other languages. Line 2 provides the inside-out functional uncertainty statement; it states that the f-structure of the ergative noun phrase, referred to as ^, is inside a SUBJ. Finally, line 3 calls a template that assigns the volitionality features associated with ergative noun phrases. The analysis for (9) is shown in Figures 3 and 4.

(9) nAdyA nE yassin ko mArA
Nadya=Erg Yassin=Acc hit-Perf.M.Sg
'Nadya hit Yassin.'

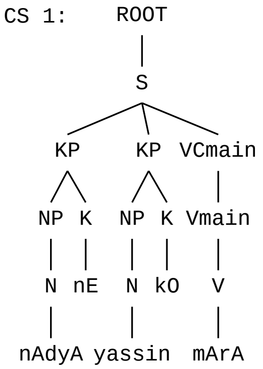
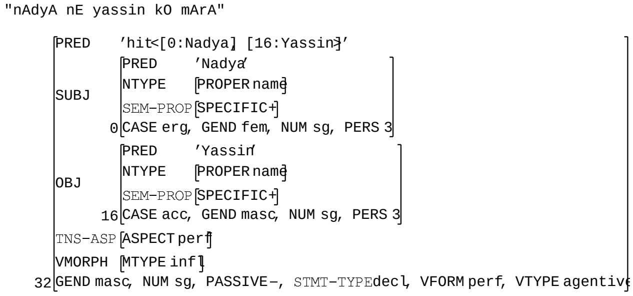


Figure 3: C-structure tree for (9)



```
"nAdyA nE yassin kO mArA"

   PRED    'hit<[0:Nadya] [16:Yassin]>'
           PRED     'Nadya'
           NTYPE    [PROPER name]
   SUBJ
           SEM-PROP [SPECIFIC +]
          0CASE erg, GEND fem, NUM sg, PERS 3
           PRED     'Yassin'
           NTYPE    [PROPER name]
   OBJ
           SEM-PROP [SPECIFIC +]
         16CASE acc, GEND masc, NUM sg, PERS 3
   TNS-ASP [ASPECT perf]
   VMORPH  [MTYPE infl]
 32GEND masc, NUM sg, PASSIVE -, STMT-TYPE decl, VFORM perf, VTYPE agentive
```

Figure 4: F-structure AVM for (9)

There are two intesting points about this analysis of case in Urdu. The first is that although the Urdu grammar processes case differently than the other grammars, the resulting f-structure in Figure 4 is similar to its counterparts in English, German, etc. English would have CASE nom on the subject instead of erg, but the remaining structure is the same: the only indication of case is the CASE feature. The second point is that Urdu tested the application of inside-out functional uncertainty to case both theoretically and computationally. In both respects, the use of inside-out functional uncertainty has proven a success: not only is it theoretically desirable for languages like Urdu, but it is also implementationally feasible, efficiently providing the desired output.

Another interesting example of how Urdu has extended the standards of the ParGram project comes from complex predicates. The English, French, and German grammars do not need a complex predicate analysis. However, as complex predicates form an essential and pervasive part of Urdu grammar, it is necessary to analyze them in the project. At first, we attempted to analyze complex predicates using the existing XLE tools. However, this proved to be impossible to do productively because XLE did not allow for the manipulation of PRED values outside of the lexicon. Given that complex predicates in Urdu are formed in the syntax and not the lexicon (Butt, 1995), this poses a significant problem. The syntactic nature of Urdu complex predicate formation is illustrated by (10), in which the two parts of the complex predicate *lık*^{h} 'write' and *diya* 'gave' can be separated.

(10) a. [anjum nE] [saddaf kO] [ciTTHI]
Anjum.F=Erg Saddaf.F=Dat note.F.Nom
[**likHnE** **dI**]
write-Inf.Obl give-Perf.F.Sg
'Anjum let Saddaf write a note.'

b. anjum nE **dI** saddaf kO [ciTTHI **likHnE**]

c. anjum nE [ciTTHI **likHnE**] saddaf kO **dI**

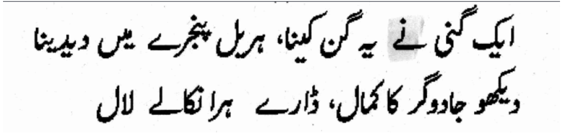
The manipulation of predicational structures in the lexicon via lexical rules (as is done for the English passive, for example), is therefore inadequate for complex predication. Based on the needs of the Urdu grammar, XLE has been modified to allow the analysis of complex predicates via the restriction operator (Kaplan and Wedekind, 1993) in conjunction with predicate composition in the syntax. These new tools are currently being tested by the implementation of the new complex predicates analysis.

5 Script

One issue that has not been dealt with in the Urdu grammar is the different script systems used for Urdu and Hindi. As seen in the previous discussions and the Figures, transcription into Latin ASCII is currently used by the Urdu grammar. This is not a limitation of the XLE system: the Japanese grammar has successfully integrated Japanese Kana and Kanji into their grammar.

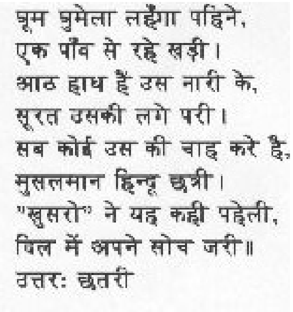
The approach taken by the Urdu grammar is different from that of the Japanese, largely because two scripts are involved. The Urdu grammar uses the ASCII transcription in the finite-state morphologies and the grammar. At a future date, a component will be built onto the grammar system that takes Urdu (Arabic) and Hindi (Devanagari) scripts and transcribes them for use in the grammar. This component will be written using finite-state technology and hence will be compatible with the finite-state morphology. The use of ASCII in the morphology allows the same basic morphology to be used for both Urdu and Hindi. Samples of the scripts are seen in (11) for Urdu and (12) for Hindi.

(11)



ایک گنی نے یہ گن کینا، ہریل پنجرے میں دیدینا
دیکھو جادوگر کا کمال، ڈارے ہرا نکالے لال

(12)



घूम घुमेला लहँगा पहिने,
एक पाँव से रहे खड़ी।
आठ हाथ हैं उस नारी के,
सूरत उसकी लगे परी।
सब कोई उस की चाह करे है,
मुसलमान हिन्दू छत्री।
"खुसरो" ने यह कही पहेली,
दिल में अपने सोच जरी॥
उत्तर: छतरी

6 Conclusion

The ParGram project was designed to use a single grammar development platform and a unified methodology of grammar writing to develop large-scale grammars for typologically different languages. At the beginning of the project, three typologically similar European grammars were used to test this idea. The addition of two Asian languages, has shown that the basic analysis decisions made for the European languages can be applied to typologically distinct languages. However, the Asian languages required the addition of a few new standard analyses to the project to cover constructions and analysis techniques not found in the European languages. With this new set of standards, the ParGram project can now be applied to other typologically distinct languages.

The parallelism between the grammars in the ParGram project can be exploited in applications using the grammars: the fewer the differences, the simpler a multi-lingual application can be. For example, a translation system that uses the f-structures as input and output can take advantage of the fact that similar constructions have the same analysis (Frank, 1999). The standardization also aids further grammar development efforts. Many of the basic decisions about analyses and formalism have already been made in the project. Thus, the grammar writer for a new language can use existing technology to bootstrap a grammar for the new language and can parse equivalent constructions in the existing languages to see how to analyze a construction. This allows the grammar writer to focus on more difficult constructions not yet encountered in the existing grammars.

References

Masayuki Asahara and Yuji Matsumoto. 2000. Extended models and tools for high-performance part-of-speech tagger. In *Proceedings of COLING*.

Kenneth Beesley and Lauri Karttunen. 2002. *Finite-State Morphology: Xerox Tools and Techniques.* Cambridge University Press. To Appear.

Miriam Butt and Ron Kaplan. 2002. The morphology syntax interface in LFG. Presented at LFG02, Athens, Greece; to appear in the proceedings (CSLI Publications).

Miriam Butt and Tracy Holloway King. to appear. The status of case. In Veneeta Dayal and Anoop Mahajan, editors, *Clause Structure in South Asian Languages.* Kluwer.

Miriam Butt, Tracy Holloway King, María-Eugenia Niño, and Frédérique Segond. 1999. *A Grammar Writer's Cookbook.* CSLI Publications.

Miriam Butt, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The parallel grammar project. In *Proceedings of COLING 2002.* Workshop on Grammar Engineering and Evaluation.

Miriam Butt. 1995. *The Structure of Complex Predicates in Urdu.* CSLI Publications.

Jean-Pierrre Chanod and Pasi Tapanainen. 1995. Creating a tagset, lexicon, and guesser for a French tagger. In *Proceedings of the ACL SIGDAT Workshop: From Texts To Tags. Issues in Multilingual Language Analysis*, pages 58–64.

Anette Frank. 1999. From parallel grammar development towards machine translation. In *Proceedings of MT Summit VII,* pages 134–142.

Ron Kaplan and Jürgen Wedekind. 1993. Restriction and correspondence-based translation. In *Proceedings of the Sixth European Conference of the Association for Computational Linguistics*, pages 193–202.

Ron Kaplan. 1988. Correspondences and their inverses. Presented at the Titisee Workshop on Unification Formalisms: Syntax, Semantics, and Implementation, Titisee, Germany.

John T. Maxwell, III and Ron Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Lingusitics*, 19:571–589.

Rachel Nordlinger. 1998. *Constructive Case: Evidence from Australian Languages.* CSLI Publications.

A Study in Urdu Corpus Construction

Dara Becker
Graduate Program in Software Engineering
University of St. Thomas
St. Paul, MN, 55105, U.S.A.
dmbecker@stthomas.edu

Kashif Riaz
Department of Computer Science
University of Minnesota-Twin Cities
Minneapolis, MN, 55455, U.S.A.
riaz@cs.umn.edu

Abstract

We are interested in contributing a small, publicly available Urdu corpus of written text to the natural language processing community. The Urdu text is stored in the Unicode character set, in its native Arabic script, and marked up according to the Corpus Encoding Standard (CES) XML Document Type Definition (DTD). All the tags and metadata are in English. To date, the corpus is made entirely of data from British Broadcasting Company's (BBC) Urdu Web site, although we plan to add data from other Urdu newspapers. Upon completion, the corpus will consist mostly of raw Urdu text marked up only to the paragraph level so it can be used as input for natural language processing (NLP) tasks. In addition, it will be hand-tagged for parts of speech so the data can be used to train and test NLP tools.

Introduction

We are interested in contributing a small, publicly available Urdu corpus of written text to the natural language processing community. In pursuit of natural language processing research in Urdu, we could not find a publicly available Urdu corpus with which to work, so we had to start our own to train and test machine learning algorithms.

The language engineering community seems anxious to move forward fast in research of South Asian languages, but cannot because corpora of South Asian languages are not ample. "There is a dearth of work on Indic languages. The need to focus on Indic languages was further strengthened by our major review (with over 80 research centres world wide responding) of the needs of the [language engineering] community. Indic languages are the ones that most researchers want to work with but cannot because lack of corpus resources" [1].

1 Urdu corpus

Our corpus is currently made up of newspaper articles and columns from the Urdu Internet site of the British Broadcasting Company (BBC Urdu). News story data is easy to gather because it is readily available on the Internet and already in electronic form, although Web sites in Urdu tend to be published in graphics (a point we will return to later).

It is important for the users of corpus data to know from where the data came. Software trained on a written text corpus will perform poorly on spoken data and vice versa.

Something to keep in mind when using this Urdu data is that vocabulary and the stylistics of news stories in Urdu are very different than in everyday speech. For example, in Urdu news stories, "militants" are described as "people who like violence" تشدد پسند عناصر . Such a phrase is hardly ever used in everyday speech. Headings of news stories have different stylistics. For example, a common way to associate a statement to the person who made the statement is to write the statement followed by a colon or dash and then the person's name. This trend has been observed in Urdu news stories published in Pakistan, India, the UK, and in the United States.

The first version of the Urdu corpus to be published will be relatively small (20,000–50,000 words), but we will regularly be adding to the corpus as time passes. It will publicly available at http://personal1.stthomas.edu/dmbecker/.

All the Urdu documents will appear in a minimally tagged format (i.e., only paragraph tags) and, in addition, will be hand-tagged for parts of speech.

2 XML

The natural choice these days for storing a corpus is in an XML format. An XML format provides needed standardization so that a user who is unfamiliar with the corpus data, but familiar with a given XML DTD, can interface with the corpus fairly efficiently. At its best, software that has been previously designed to handle a corpus marked up in a given XML structure can handle a new corpus marked up in the same structure. This is advantageous because someone does not have to comb through the new corpus trying to understand its design in order to redesign the software that interfaces with the corpus. The designer of a corpus is always familiar with his/her own design, so one advantage of using an XML language to mark up a corpus is to make the corpus readily available to other researchers.

We chose the Corpus Encoding Standard (CES) XML DTD to mark up our corpus [2]. The main enclosing tag in this DTD is <cesCorpus> which is broken into main parts, <cesHeader> and <cesDoc>.

The header <cesHeader> contains meta information about the corpus data such as, date created, creator's name and contact information, description of the source, categories of the content, the writing system of the language being stored, how hyphenation in the source text is handled, and much more information (Figure 1).

The document tag <cesDoc> is where the actual text of the language of interest is stored. Each document is itself marked up with metadata specific to each document, like topic and source information for every separate document in the corpus.

The language data inside the <cesDoc> tags can be marked up simply with a paragraph tag <p> (Figure 2) or they can be more elaborately marked up with tags of semantic value (e.g., date, number, measure, name, term, time, foreign word) and formatting value (e.g., figure, table, p, sp, div, caption) (Figure 3). Tags that indicate formatting features such as 'caption' are important because they can be used, for example, to automatically determine the topic of a story.

The actual implementation of tagging Urdu script at a detailed level presents a display problem for our XML editor of choice, XML Spy. Upon looking at Figure 3, which is an excerpt from XML Spy, one may think that the word order of the paragraph is out of order. At the display level, the word order is out of order—it is barely human-readable, but at the storage level, the text is perfectly tagged and will process correctly. In Figure 4, we show, in a human-readable format, the order in which the Urdu text and English tags are stored. If an XML editor were optimized to display a right-to-left language with left-to-right tags, this is how we imagine the text would look. More importantly though, this is the order in which XML Spy currently stores the Urdu corpus.

We began the corpus building process by storing Urdu documents at the paragraph level with no other tags peppering the data. However, we intend to hand tag the data for parts of speech so the data can be used to train and test natural language processing algorithms.

```
<cesHeader type="corpus" creator="Dara Becker" version="1.0" status="update"
    date.created="2/2/02" date.updated="4/17/02">
    <fileDesc>
        <titleStmt><h.title>Urdu Corpus</h.title></titleStmt>
        <editionStmt version="1.0a"/>
        <publicationStmt>
            <distributor>Dara Becker</distributor>
            <telephone></telephone>
            <eAddress type="email">dmbecker@stthomas.edu</eAddress>
            <eAddress type="www">http://personal1.stthomas.edu/DMBECKER/</eAddress>
            <availability status="free"/>
        </publicationStmt>
    </fileDesc>
</cesHeader>
```

Figure 1: An excerpt from the corpus header
(It is not well-formed because we deleted some required tags.)

```
<cesDoc version="1.0">
     <cesHeader type="text" creator="Dara Becker" version="1.0" status="new"
          date.created="2/18/02" date.updated="" lang="ur">
     </cesHeader>
     <text>
          <body>
               <p>
                    <title>غور پر کرنے بدر ایران کو یار حکمت<title>
               </p>
               <p>
امریکی خلاف کے مداخلت کی یونین سوویت سابق جو یار حکمت ہے۔ رہا جا کیا غور بھی پر کرنے بدر ایران انھیں کہ ہیں خبریں اور
بھی کی انتظامیہ کرزئی وہ اب اور ہیں جاتے جانے لئے کے خیالات مخالف اب تھے آئے سامنے میں مزاحمت والی چلے سے حمایت
کاروائیاں خلاف کے انتظامیہ افغان کو سرزمین کی ایران وہ کہ تھا لگایا الزام پر یار حکمت نے ایران ہفتے گذشتہ تھے۔ کررہے مخالفت
وہ تھا رہاکر فراہم حمایت جو کو دھڑوں مزاحم خلاف کے طالبان وہ کہ ہے کہنا کا ایران کہ جب ہیں کررہے استعمال لئے کے کرنے
کے امریکہ اقدام خلاف کے یار حکمت نے ایران کہ ہے خیال کا ذرائع بعض تاہم ہے۔ گئی دی کر بند بعد کے ہونے ختم کنٹرول کا طالبان
ہیں۔ کیے بعد کے اعتراضات
               </p>
          </body>
     </text>
</cesDoc>
```

Figure 2: An excerpt from a corpus document
(It is not well-formed because we deleted some required tags.)

```
<text>
          <body>
                    <p>
                              <title>غور پر کرنے بدر ایران کو یار حکمت</title>
                    </p>
                    <p>یار حکمت<name> ہے۔ رہا جا کیا غور بھی پر کرنے بدر ایران انھیں کہ ہیں خبریں اور
والی چلے سے حمایت امریکی خلاف کے مداخلت کی </name>یونین سوویت <name>سابق جو </name>□ر
مخالفت بھی کی انتظامیہ کرزئی وہ اب اور ہیں جاتے جانے لئے کے خیالات مخالف اب تھے آئے سامنے میں مزاحمت
<name>وہ کہ تھا لگایا الزام پر </name>یار حکمت<name> نے </name>ایران<name> ہفتے گذشتہ تھے۔ کررہے
<name>کہ جب ہیں کررہے استعمال لئے کے کرنے کاروائیاں خلاف کے انتظامیہ افغان کو سرزمین کی </name>ایران
تھا رہاکر فراہم حمایت جو کو دھڑوں مزاحم خلاف کے </name>طالبان<name> وہ کہ ہے کہنا کا </name>ایران
کہ ہے خیال کا ذرائع بعض تاہم ہے۔ گئی دی کر بند بعد کے ہونے ختم کنٹرول کا </name>طالبان<name> وہ
اعتراضات کے </name>امریکہ<name> اقدام خلاف کے </name>یار حکمت <name>نے </name>ایران<name>
ہیں۔ کیے بعد کے</p>
          </body>
</text>
```

Figure 3: An illustration of how detailed tagging rearranges the display of the text in XML Spy

اور خبریں ہیں کہ انھیں ایران بدر کرنے پر بھی غور کیا جا رہا ہے۔ <name> حکمت یار </name> جو سابق
<name> سوویت یونین </name> کی مداخلت کے خلاف امریکی حمایت سے چلے والی مزاحمت میں سامنے آئے تھے
اب مخالف خیالات کے لئے جانے جاتے ہیں اور اب وہ کرزئی انتظامیہ کی بھی مخالفت کررہے تھے۔ گذشتہ ہفتے <name> ایران
</name> نے <name> حکمت یار </name> پر الزام لگایا تھا کہ وہ

Figure 4: A human-readable rendition of what tagged Urdu would look like in an XML editor optimized to display a right-to-left language with left-to-right tags

3 Unicode

Another natural choice for storing data is to use the Unicode character set. The Unicode character set is another needed standard that we take advantage of in order to make our corpus data readily available to other researchers.

The only reason for choosing to initially store text from BBC Urdu, and not other news agencies, is that the BBC publishes in the Unicode character set. Other news sites that publish in Urdu have gotten in the habit of publishing in graphics, presumably to avoid the hassles of arranging compatible fonts and character sets in the publishing software, systems, and client browsers. We think too it could be that Urdu publishers prefer Nastaliq-style font. There are probably a host of wonderful Nastaliq-style fonts available that work on legacy character sets, and, perhaps, publishers prefer to keep using these fonts.

The choice to publish in graphics though makes it difficult for data harvesters to snag data from the Web. If one really wants the data that are published in graphic form, one has to rekey the text, scan it using optical character recognition technology, or contact the publisher for electronic copies of text, in which case one needs to be able to handle or convert from the character set in which the text was originally typed. In a previous project, we developed an application that can convert between 120 legacy character sets and can be customized to convert any other font or character set, so we should have minimal obstacles when it comes time to harvest non-Unicode data.

Storing Urdu data in the Unicode character set eliminates some problems—however, we have found other problems related to different approaches to mapping Unicode-based fonts to the Arabic subset of Unicode.

Unicode-based fonts seem to have been optimized for Arabic display, not for Urdu, so we have experienced difficulty displaying various forms of *heh*, *noon ghunna*, and *hamza*. We found the best Unicode-based font for properly displaying Urdu is Urdu Naskh Asiatype, available from the BBC Urdu Web site, at least among free fonts.

We compared this font (presumably optimized for Urdu) and Arial Unicode MS (presumably not optimized for Urdu) and found that the letter *heh* and its variations are mapped differently in these two Unicode-based fonts (Table 1).

Table 1: How fonts display variations of the letter *heh*

	Urdu Naskh Asiatype display	Arial Unicode MS display
06C1 ہ	FBA6 ﮦ or FEE9 ﻩ FEEA ﻪ FBA8 ﮨ FBA9 ﮩ	FBA8 ﮨ FBA9 ﮩ
06BE ھ	FBAA ﮪ or FEEB ﻫ FEEC ﻬ	FBAA ﮪ or FEEB ﻫ
0647 ه	not found in corpus	FBA6 ﮦ or FEE9 ﻩ FEEA ﻪ FEEC ﻬ

For this reason, the metadata of the Urdu text in the corpus will contain the name of the Unicode-based font in which the text is stored. Any text processor that uses the data will have to normalize the usages of *heh* and its variations. In order to view the Urdu text properly in its surface form the font in which the data was harvested will have to be applied.

Differences in font mappings are not much of a problem when handling English and other Roman-based orthographies, especially when using the Unicode character set, so special attention has to be paid to the different ways fonts display surface forms of Urdu letters.

4 Urdu input method

In order to add an Urdu document to our corpus that we only have in graphic form or hard copy, we spent significant time setting up our computer for Urdu Unicode input in order to be able to type into the corpus.

Using the Arabic support on our computer, Microsoft Windows 2000 5.0 Service Pack 2, we were easily able to install right-to-left script support. Since Windows 2000 uses the Unicode character set internally, we did not have to do anything special to get Unicode support for our efforts.

Devising a plan for inputting Urdu on the keyboard was the biggest challenge. We ended up using Tavultesoft Keyman software to map our own keyboard—it was very easy to use. Existing keyboard mappings for Arabic script-based languages, we found, are generally not phonetically mapped, meaning we would like Urdu letter *feh* to be mapped to the letter *f* on the keyboard and so forth. We did find one phonetically mapped keyboard that we liked for Persian [3], CRL Phonetic Layout, so we used that mapping as a basis for developing our own. It is not important that our keyboard mapping be standardized—it only need work for the one person typing our text.

Conclusion

In this paper, we presented the methodology we used to build an Urdu corpus. The process of corpora construction for South Asian languages, specifically Urdu, involves extra work because these languages are not written in a Roman-based script. The use of the Unicode character set and software that supports it makes building needed corpora in these languages possible and relatively easy. Once corpora in these languages become readily available, natural language processing work in these languages can move forward.

References

[1] P. Baker and A. McEnery, “Needs of Language-Engineering Communities; Corpus Building and Translation Resources,” MILLE working paper 7, Lancaster University, 1999.

[2] N. Ide and G. Priest-Dorman, Eds., “Corpus Encoding Standard,” [Online document], 2000 March 20, [cited 2002 Feb 28], Available: http://www.cs.vassar.edu/CES

[3] “Persian Keyboard Layouts,” Computing Research Laboratory, [Online document], [cited 2002 May 5], Available: http://crl.nmsu.edu/~mleisher/keyboards/persian.html

Automatic Word Spacing Using Hidden Markov Model for Refining Korean Text Corpora

Do-Gil Lee and **Sang-Zoo Lee** and **Hae-Chang Rim**
NLP Lab., Dept. of Computer Science and Engineering, Korea University
1, 5-ka, Anam-dong, Seongbuk-ku, Seoul 136-701, Korea

Heui-Seok Lim
Dept. of Information and Communications, Chonan University
115 AnSeo-dong, CheonAn 330-704, Korea

Abstract

This paper proposes a word spacing model using a hidden Markov model (HMM) for refining Korean raw text corpora. Previous statistical approaches for automatic word spacing have used models that make use of inaccurate probabilities because they do not consider the previous spacing state. We consider word spacing problem as a classification problem such as Part-of-Speech (POS) tagging and have experimented with various models considering extended context. Experimental result shows that the performance of the model becomes better as the more context considered. In case of the same number of parameters are used with other method, it is proved that our model is more effective by showing the better results.

1 Introduction

Automatic word spacing is a process to decide correct boundaries between words in a sentence containing spacing errors. In Korean, word spacing is very important to increase the readability and to communicate the accurate meaning of a text. For example, if a sentence "아버지가 방에 들어가셨다(Father entered the room)" is written as "아버지 가방에 들어가셨다(Father entered the bag)", then its meaning is changed a lot.

There are many word spacing errors in documents on the Internet, which is the principal source of information. To deal with these documents properly, an automatic word spacing system is absolutely necessary. Besides, it plays an important role as a preprocessor of a morphological analyzer that is a fundamental tool for natural language processing applications, a postprocessor to restore line boundaries from an OCR, a postprocessor for continuous-syllable sentence from a speech recognition system, and one module for an orthographic error revision system.

In Korean, spacing unit is Eojeol. Each Eojeol consists of one or more words and a word consists of one or more morphemes. Figure 1 represents their relationships for a sentence "철수가 이야기책을 읽었다". According to the rules of Korean spelling, the main principle for word spacing is to split every word in a sentence. Because one morpheme may form a word and several morphemes too, there are confusing cases to distinguish among words. Even though postpositions belong to words, they should be concatenated with the preceding word. Besides, there are many conflicting (but can be permitted) cases with the principles. For example, spacing or concatenating individual nouns including a compound noun are both considered as right. As mentioned, word spacing is important for some reasons, but it is difficult for even man to space words correctly by spelling rules because of the characteristics of Korean and the inconsistent rules. Especially, it is much more confused in the case of having no influence on understanding the meaning of a sentence.

In this paper, we propose a word spacing model [1] using an HMM. HMM is a widely used statistical model to solve various NLP problems such as POS tagging(Charniak et al., 1993; Merialdo, 1994; Kim et al., 1998a; Lee, 1999). We regard the word spacing problem as a classification problem such as the POS tagging problem. When using an HMM for automatic word spacing task, raw texts can be used as training

[1] Strictly speaking, our model described here is an Eojeol spacing model rather than a word spacing model because spacing unit of Korean is Eojeol. But we in this paper do not distinguish between Eojeol and word for convenience. Therefore, we use the term "word" as word, spacing unit in English.

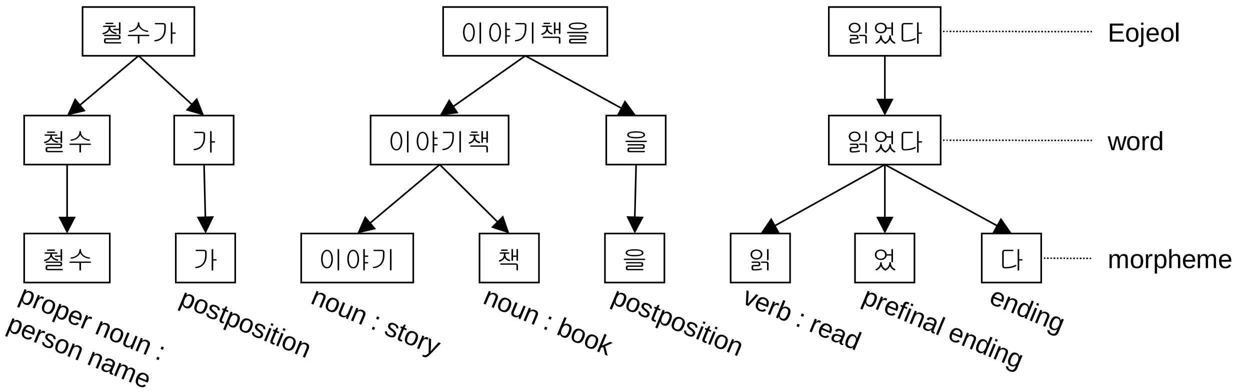


Figure 1: Constitution of the sentence "철수가 이야기책을 읽었다"

data. Therefore, we expect that HMM can be applied to the task effectively without bothering to construct training data.

2 Related Works

Previous approaches for automatic word spacing can be classified into two groups: rule based approach and statistical approach. The rule-based approach uses lexical information and heuristic rules(Choi, 1997; Kim et al., 1998b; Kang, 1998; Kang, 2000). Lexical information consists of postposition and Eomi[2] information, a list of spaced word examples, etc. Heuristic rules are composed of longest match or shortest match rule, morphological rules, and error patterns. This approach has disadvantage requiring higher computational complexity than the statistical approach. It also costs too much in constructing and maintaining lexical information. Most of rule-based systems use a morphological analyzer to recognize word boundaries. Another disadvantages of rule-based approach are resulted from using morphological analyzer. First, if ambiguous analyses are possible, frequent backtracking may be caused and many errors are propagated by an erroneous analysis. Second, results of automatic word spacing are highly dependent on the morphological analyzer; false word boundary recognition occurs if morphological analysis fails due to unknown words. In addition, if an erroneous word is successfully analyzed through overgeneration, the error cannot even be detected. Finally, if a word spacing system is used as a preprocessor of a morphological analyzer, the same morphological analyzing process should be repeated twice.

The statistical approach uses syllable statistics extracted from large amount of corpora to decide whether two adjacent syllables should be spaced or not(Shim, 1996; Shin and Park, 1997; Chung and Lee, 1999; Jeon and Park, 2000; Kang and Woo, 2001). In contrast to the rule-based approach, it does not require many costs to construct and to maintain statistics because they can be acquired automatically. It is more robust against unknown words than rule-based approach that uses a morphological analyzer.

A statistical method proposed in Kang and Woo (2001) has shown the best performance so far. In this method, word spacing probability $P(x_i, x_{i+1})$, between two adjacent syllables x_i and x_{i+1}, is in Equation 1. If the probability is greater than 0.375, a space is inserted between x_i and x_{i+1}.

$$\begin{aligned} P(x_i, x_{i+1}) = {} & 0.25 \times P_R(x_{i-1}, x_i) + \\ & 0.5 \times P_M(x_i, x_{i+1}) + \\ & 0.25 \times P_L(x_{i+1}, x_{i+2}) \end{aligned} \quad (1)$$

In Equation 1, P_R, P_M, and P_L denote the probability of a space being inserted in the right, middle, and left of the two syllables, respectively. They are calculated as follows:

$$P_R(x_{i-1}, x_i) = \frac{freq(x_{i-1}, x_i, SPACE)}{freq(x_{i-1}, x_i)}$$

$$P_M(x_i, x_{i+1}) = \frac{freq(x_i, SPACE, x_{i+1})}{freq(x_i, x_{i+1})}$$

[2] Eomi is a grammatical morpheme of Korean which is attached to verbal root

$$P_L(x_{i+1}, x_{i+2}) = \frac{freq(SPACE, x_{i+1}, x_{i+2})}{freq(x_{i+1}, x_{i+2})}$$

In the above equations, $freq(x)$ denotes a frequency of a string x from training data, and $SPACE$ denotes a white space.

Similar to this method, other statistical systems usually use the word spacing probability estimated from every syllable bigram[3] in the corpora. They calculate the probability by combining P_R, P_M, and P_L and compare it with a certain threshold. If the probability is higher than the threshold, then a space is inserted between two syllables.

It is reported that the performance is so sensitive to training data: it shows somewhat different performance according to similarity between input document and training data. And there is a crucial problem in the statistical method resulted from not considering the previous spacing state. For example, consider a sentence "공부할수있다" of which correctly word spaced sentence is "공부할 수 있다". According to Equation 1, the word spacing probability of "수" and "있" will be calculated as follows:

$$P(\text{수}, \text{있}) = 0.25 \times P_R(\text{할}, \text{수}) + 0.5 \times P_M(\text{수}, \text{있}) + 0.25 \times P_L(\text{있}, \text{다})$$

The probability $P_R(\text{할}, \text{수})$ as follows:

$$P_R(\text{할}, \text{수}) = \frac{freq(\text{할}, \text{수}, SPACE)}{freq(\text{할}, \text{수})}$$

But a space should have been inserted between "할" and "수" in the correct sentence, we should use $freq(SPACE, \text{수}, SPACE)$ instead of $freq(\text{할}, \text{수}, SPACE)$ in order to get the correct word spacing probability. This phenomenon comes from not considering the previous spacing state. To alleviate this problem, we can consider the previous spacing state that the system has decided before. But errors can be propagated from the previous false word spacing result. Eventually, to avoid such propagated errors, the system has to generate all possible interpretations from a given sentence and choose the best one. To choose the best state from all possible states, we use an HMM in this paper.

3 Word Spacing Model based on Hidden Markov Model

POS tagging is the most representative area for HMM. Before explaining our word spacing model using HMM, let's consider the POS tagging model using an HMM. POS tagging function $\Gamma(W)$ is to find the most likely sequence of POS tags $T = (t_1, t_2, \ldots, t_n)$ for a given sentence of words $W = (w_1, w_2, \ldots, w_n)$ and is defined in Equation 2:

$$\Gamma(W) \overset{def}{=} \operatorname*{argmax}_T P(T \mid W) \quad (2)$$

$$= \operatorname*{argmax}_T \frac{P(T)P(W \mid T)}{P(W)} \quad (3)$$

$$= \operatorname*{argmax}_T P(T)P(W \mid T) \quad (4)$$

$$= \operatorname*{argmax}_T P(T, W) \quad (5)$$

Using Bayes' rule, Equation 2 becomes Equation 3. Since $P(W)$ is a constant for T, Equation 3 is transformed into Equation 4.

The probability $P(T, W)$ is broken down into the following equations by using the chain rule:

$$P(T, W) = P(t_{1,n}, w_{1,n}) \quad (6)$$

$$= \prod_{i=1}^{n} \begin{pmatrix} P(t_i \mid t_{1,i-1}, w_{1,i-1}) \\ \times P(w_i \mid t_{1,i}, w_{1,i-1}) \end{pmatrix} \quad (7)$$

$$\approx \prod_{i=1}^{n} P(t_i \mid t_{i-K,i-1}) P(w_i \mid t_i) \quad (8)$$

Markov assumptions (conditional independence) used in Equation 8 are that the probability of a current tag t_i conditionally depends on only the previous K tags and that the probability of a current word w_i conditionally depends on only the current tag. In Equation 8, $P(t_i \mid t_{i-K,i-1})$ is called transition probability and $P(w_i \mid t_i)$ is called lexical probability. Models are classified in terms of K. The larger K is, the more context can be considered. Because of the data sparseness problem, bigram model (K is 1) and trigram model (K is 2) are used in general.

The word spacing problem can be considered similar to POS tagging. We define a word spacing task as a task to find the most likely sequence of word spacing tags $T = (t_1, t_2, \ldots, t_n)$ for a given sentence of syllables

[3] syllable bigram is defined to be any combination of two syllables with or without a space.

$S = (s_1, s_2, \ldots, s_n)$. Our word spacing model is defined as in Equation 9:

$$\operatorname*{argmax}_{T} P(T \mid S) \tag{9}$$

Word spacing tag is a tag to indicate whether the current syllable and the next one should be spaced or not. Tag, 1 means that a space should be put after the current syllable. Tag, 0 means that the current and the next syllable should not be spaced. For example, if we attach the word spacing tags to a sentence "공부 할 수 있다. (I can study)", then it is tagged as "공/0+부/0+할/1+수/1+있/0+다/0+./1".

Our proposed word spacing model is to find the tag sequence T for maximizing the probability $P(T, S)$.

$$\begin{aligned}
&P(T,\ S\ ) \\
&\quad = P(t_{1,n}, s_{1,n}) && (10)\\
&\quad = \Big(P(t_1) \times p(s_1 \mid t_1)\Big) \\
&\qquad \times \Big(P(t_2 \mid t_1, s_1) \times P(s_2 \mid t_{1,2}, s_1)\Big) \\
&\qquad \times \begin{pmatrix} P(t_3 \mid t_{1,2}, s_{1,2}) \\ \times P(s_3 \mid t_{1,3}, s_{1,2}) \end{pmatrix} \times \cdots \\
&\qquad \times \begin{pmatrix} P(t_n \mid t_{1,n-1}, s_{1,n-1}) \\ \times P(s_n \mid t_{1,n}, s_{1,n-1}) \end{pmatrix} && (11)\\
&\quad = \prod_{i=1}^{n} \begin{pmatrix} P(t_i \mid t_{1,i-1}, s_{1,i-1}) \\ \times P(s_i \mid t_{1,i}, s_{1,i-1}) \end{pmatrix} && (12)\\
&\quad \approx \prod_{i=1}^{n} \begin{pmatrix} P(t_i \mid t_{i-K,i-1}, s_{i-J,i-1}) \\ \times P(s_i \mid t_{i-L,i}, s_{i-I,i-1}) \end{pmatrix} && (13)
\end{aligned}$$

There are two Markov assumptions in Equation 13. One is that the probability of a current tag t_i conditionally depends on only the previous K (word spacing) tags and the previous J syllables. The other is that the probability of a current syllable s_i conditionally depends on only the previous L tags, the current tag t_i, and the previous I tags. This model is denoted by $\Lambda(T_{(K:J)}, S_{(L:I)})$. Similar to the POS tagging model, $P(t_i \mid t_{i-K,i-1}, s_{i-J,i-1})$ is called transition probability, and $P(s_i \mid t_{i-L,i}, s_{i-I,i-1})$ is called syllable probability in Equation 13. On the other hand, our word spacing model uses less strict Markov assumptions to consider a larger context. The larger the values of K, J, L, and I are, the more context can be considered. In order to avoid the data sparseness and excessively increasing parameters of a model, it is important to select proper values. In our current work, they are restricted as follows:

$$0 \leq K, J, L, I \leq 2$$

Thus, $3\times3\times3\times3 = 81$ models are possible. But we do not use the case of $(K, J) = (0, 0)$ in the trasition probabilities. As a result, we actually use 72 models. It has not yet been known that which model is the best. We can verify this only by means of experiments. Some possible models and their equations are listed in Table 1.

Probabilities can be estimated simply by the maximum likelihood estimator (MLE) from raw texts. The syllable probabilities and the transition probabilities of the model $\Lambda(T_{(1:2)}, S_{(1:2)})$ are estimated as follows:

$$\begin{aligned}
&P_{MLE}(t_i \mid t_{i-1}, s_{i-2,i-1}) \\
&\qquad = \frac{freq(s_{i-2}, t_{i-1}, s_{i-1}, t_i)}{freq(s_{i-2}, t_{i-1}, s_{i-1})} \\
&P_{MLE}(s_i \mid t_{i-1,i}, s_{i-2,i-1}) \\
&\qquad = \frac{freq(s_{i-2}, t_{i-1}, s_{i-1}, t_i, s_i)}{freq(s_{i-2}, t_{i-1}, s_{i-1}, t_i)}
\end{aligned}$$

To avoid zero probability, we just set very low value such as 0.00001 if an estimated probability is 0.

The probability that the model $\Lambda(T_{(1:1)}, S_{(0:1)})$ outputs "공/0+부/0+할/1+수/1+있/0+다/0+./1" from a sentence "공부할수있다." is calculated as follows:

$$\begin{aligned}
P(T, S) &= P(t_0 = 0 \mid s_{-1} = \$, t_{-1} = 1) \\
&\times\ P(s_0 = \text{공} \mid s_{-1} = \$, t_0 = 0) \\
&\times\ P(t_1 = 0 \mid s_0 = \text{공}, t_0 = 0) \\
&\times\ P(s_1 = \text{부} \mid s_0 = \text{공}, t_1 = 0) \\
&\times\ P(1 \mid \text{부}0) \cdot P(\text{할} \mid \text{부}1) \\
&\times\ P(1 \mid \text{할}1) \cdot P(\text{수} \mid \text{할}1) \\
&\times\ P(0 \mid \text{수}1) \cdot P(\text{있} \mid \text{수}0) \\
&\times\ P(0 \mid \text{있}0) \cdot P(\text{다} \mid \text{있}0) \\
&\times\ P(1 \mid \text{다}0) \cdot P(. \mid \text{다}1)
\end{aligned}$$

"\$" is a pseudo syllable which denotes the start of a sentence, and its tag is always 1.[4] The

[4] Because any two adjacent sentences should always be spaced.

Table 1: Some models and their equations

Model	Equation
$\Lambda(T_{(1:0)}, S_{(0:0)})$	$\prod_{i=1}^{n} P(t_i \mid t_{i-1}) \cdot P(s_i \mid t_i)$
$\Lambda(T_{(1:1)}, S_{(0:1)})$	$\prod_{i=1}^{n} P(t_i \mid t_{i-1}, s_{i-1}) \cdot P(s_i \mid t_i, s_{i-1})$
$\Lambda(T_{(1:1)}, S_{(1:1)})$	$\prod_{i=1}^{n} P(t_i \mid t_{i-1}, s_{i-1}) \cdot P(s_i \mid t_{i-1,i}, s_{i-1})$
$\Lambda(T_{(1:2)}, S_{(1:2)})$	$\prod_{i=1}^{n} P(t_i \mid t_{i-1}, s_{i-2,i-1}) \cdot P(s_i \mid t_{i-1,i}, s_{i-2,i-1})$
$\Lambda(T_{(2:2)}, S_{(2:2)})$	$\prod_{i=1}^{n} P(t_i \mid t_{i-2,i-1}, s_{i-2,i-1}) \cdot P(s_i \mid t_{i-2,i}, s_{i-2,i-1})$

most probable sequence of word spacing tags is efficiently computed by using the Viterbi algorithm.

4 Experimental Results

We used balanced 21st Century Sejong Project's raw corpus of 26 million word size. As the balanced corpus is used as training data, we expect that the performance would not be sensitive too much to a certain document genre. The ETRI POS tagged corpus of 288,269 word size was used for evaluation. We modified the corpus with no word boundary form for automatic word spacing evaluation.

We used three kinds of evaluation measures: syllable-unit accuracy (P_{syl}), word-unit recall (R_{word}), and word-unit precision (P_{word}). The word-unit recall is the rate of the number of correctly spaced words compared to the number of total words in a test document. The word-unit precision measures how accurate the system's results are. The reason why we do not divide the syllable-unit accuracy as recall and precision is that the number of syllables in a document and that of the system created are the same. Each measure is defined as follows:

$$P_{syl} = \frac{S_{correct}}{S_{total}} \times 100(\%)$$

$$R_{word} = \frac{W_{correct}}{W_{Dtotal}} \times 100(\%)$$

$$P_{word} = \frac{W_{correct}}{W_{Stotal}} \times 100(\%)$$

Where, $S_{correct}$ is the number of correctly spaced syllables, S_{total} is the total number of syllables in a document, $W_{correct}$ is the number of correctly spaced words, W_{Dtotal} is the total number of words in a document, and W_{Stotal} is the total number of words created by a system.

To investigate every model, we calculated the two accuracies for different K, J, L, and I. Accuracies for each model are listed in Table 2.

According to the experimental results, we are sure that models considering more contexts show better results. The model $\Lambda(T_{(2:2)}, S_{(1:2)})$ is the best for all measures.

Note that some models show the better accuracies than the model $\Lambda(T_{(2:2)}, S_{(2:2)})$, which uses the largest context. It seems that this is caused by sparseness of data. After evaluating the method of Kang and Woo (2001) for our training and test data, it shows 93.06% syllable-unit accuracy, 76.71% word-unit recall, and 67.80% word-unit precision. Compared with these results, our model shows much better performance. If I is two in $\Lambda(S_{(K:J)}, T_{(L:I)})$, syllable trigrams are used. Although I is less than two (such as the model $\Lambda(T_{(2:1)}, S_{(1:1)}$, which uses syllable bigrams), our model is better than Kang and Woo (2001)'s. This fact tells us that our model is also more effective even when used the same number of parameters of the model.

There are two questions that we want to know about the word spacing models: First, how much training data is required to get the best performance of a given model. Second, which model best fits a given training corpus. To answer these questions, we compare the performance of various models according to the size of training corpus in Figure 2. The left plot shows the syllable-unit precision and the right plot shows the word-unit precision. In the figure, "HMM" denotes the proposed model, and its number decides the model's type. "Kang" denotes Kang and Woo (2001)'s model. "HMM2110" uses syllable unigrams, "HMM2111" and "Kang" use syllable bigrams, and "HMM2212" uses syllable trigrams. The models used here are the models that show the best accuracies among the models that use same

Table 2: Experimental results according to $(K,\ J,\ L,\ I)$

Model	P_{syl}	R_{word}	P_{word}	Model	P_{syl}	R_{word}	P_{word}	Model	P_{syl}	R_{word}	P_{word}
(0,1,0,0)	84.26	41.28	44.06	(0,1,0,1)	88.93	55.38	57.10	(0,1,0,2)	88.45	53.83	55.88
(0,1,1,0)	89.44	56.91	61.34	(0,1,1,1)	95.58	79.31	82.58	(0,1,1,2)	95.74	79.76	83.68
(0,1,2,0)	84.44	42.15	47.02	(0,1,2,1)	92.86	70.26	71.63	(0,1,2,2)	94.97	76.90	79.45
(0,2,0,0)	85.48	45.65	47.52	(0,2,0,1)	88.93	56.24	57.21	(0,2,0,2)	89.59	58.23	59.88
(0,2,1,0)	90.22	59.12	63.74	(0,2,1,1)	95.60	79.26	82.94	(0,2,1,2)	95.92	80.41	84.56
(0,2,2,0)	86.46	47.62	52.15	(0,2,2,1)	93.44	72.06	73.90	(0,2,2,2)	95.22	77.84	80.59
(1,0,0,0)	85.75	47.05	48.96	(1,0,0,1)	90.24	60.73	62.20	(1,0,0,2)	89.74	58.68	61.09
(1,0,1,0)	89.28	59.80	59.98	(1,0,1,1)	95.64	81.17	81.81	(1,0,1,2)	95.90	81.50	83.56
(1,0,2,0)	82.85	45.10	45.38	(1,0,2,1)	93.30	73.04	73.39	(1,0,2,2)	94.94	77.52	78.88
(1,1,0,0)	85.83	49.95	50.43	(1,1,0,1)	90.96	63.18	64.89	(1,1,0,2)	90.21	62.99	62.58
(1,1,1,0)	89.85	61.47	62.80	(1,1,1,1)	96.15	82.88	84.10	(1,1,1,2)	96.17	82.67	84.86
(1,1,2,0)	84.21	49.44	49.29	(1,1,2,1)	94.07	75.54	76.87	(1,1,2,2)	95.62	80.32	82.13
(1,2,0,0)	87.21	54.25	54.85	(1,2,0,1)	90.83	63.34	64.59	(1,2,0,2)	91.54	66.39	67.00
(1,2,1,0)	90.74	64.14	65.63	(1,2,1,1)	96.07	82.44	84.09	(1,2,1,2)	96.39	83.51	85.91
(1,2,2,0)	86.96	55.50	55.95	(1,2,2,1)	94.67	77.53	79.28	(1,2,2,2)	95.90	81.39	83.42
(2,0,0,0)	86.18	50.25	51.42	(2,0,0,1)	90.44	61.97	63.61	(2,0,0,2)	89.77	61.52	62.17
(2,0,1,0)	89.49	61.07	61.32	(2,0,1,1)	95.83	82.11	82.73	(2,0,1,2)	95.91	82.09	83.39
(2,0,2,0)	83.37	46.52	47.15	(2,0,2,1)	93.55	73.91	74.63	(2,0,2,2)	95.03	78.36	78.96
(2,1,0,0)	86.51	52.60	53.46	(2,1,0,1)	91.10	64.81	65.85	(2,1,0,2)	90.69	65.11	65.10
(2,1,1,0)	90.34	64.04	64.90	(2,1,1,1)	96.29	83.73	84.74	(2,1,1,2)	96.28	83.43	85.21
(2,1,2,0)	85.07	52.32	52.63	(2,1,2,1)	94.31	76.69	77.82	(2,1,2,2)	95.91	81.51	83.45
(2,2,0,0)	88.58	58.94	59.84	(2,2,0,1)	91.78	67.07	68.32	(2,2,0,2)	92.44	69.88	70.54
(2,2,1,0)	91.65	67.82	69.14	(2,2,1,1)	96.26	83.46	84.88	(2,2,1,2)	**96.69**	**84.93**	**86.82**
(2,2,2,0)	88.97	61.20	62.28	(2,2,2,1)	95.01	78.99	80.60	(2,2,2,2)	96.04	82.05	83.96

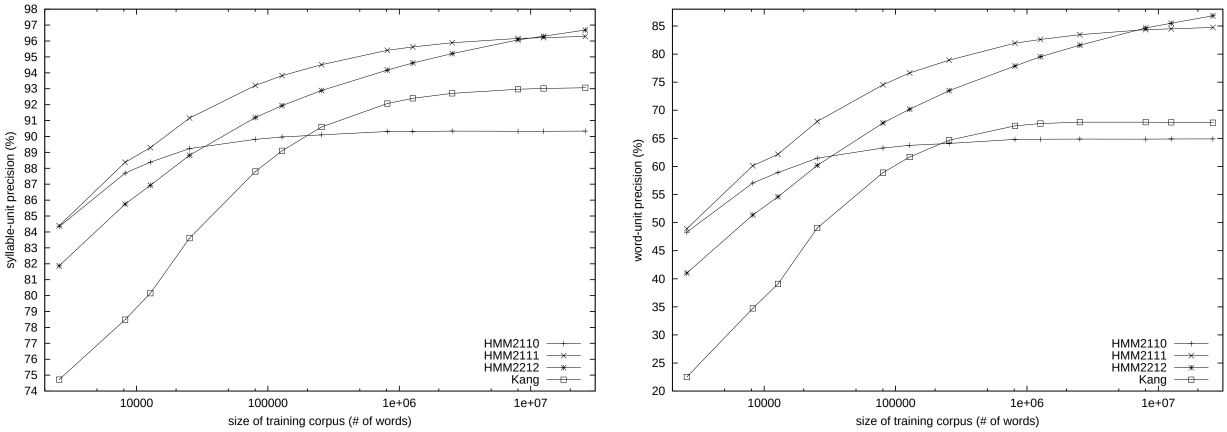


Figure 2: Accuracies according to the size of training corpus

syllable ngrams.

We can observe the changes of the accuracies according to the size of the training data. "HMM2110" using syllable unigrams converges quickly on small training data. "HMM2111" and "Kang" using syllable bigrams converge on much more training data. Note that "HMM2212" does not converge in these plots. Therefore, there is a possibility of improvement of this model's performance on more large training data. "HMM2212" shows lower performance than other models on small training data. The reason is that the data sparseness problem occurs.

5 Conclusion

Recently, text resources available from the Internet have been rapidly increased. However, there are many word spacing errors in those resources, which cannot be used before correcting errors. Therefore, the need for automatic word spacing system to refine text corpora has been raised. In this paper, we have proposed an automatic word spacing model using an HMM. Our method is a statistical approach and does not require complex processes and costs in constructing and maintaining lexical information as in the rule-based approach. The proposed model can effectively solve the word spacing problem by using only syllable statistics automatically extracted from raw corpora. According to the experimental results, our model shows higher performance than the previous method even when using the same number of parameters. We used just MLE to estimate probability, but the more a model extends the context; the more the data sparseness problem may arise.

In future work, we plan to adopt a smoothing technique to increase the performance. Further research on an effective evaluation method for conflicting cases is also necessary.

References

E. Charniak, C. Hendrickson, N. Jacobson, and M. Perkowitz. 1993. Equations for part-of-speech tagging. In *National Conference on Artificial Intelligence*, pages 784–789.

J.-H. Choi. 1997. Automatic Korean spacing words correction system with bidirectional longest match strategy. In *Proceedings of the 9th Conference on Hangul and Korean Information Processing*, pages 145–151.

Y.-M. Chung and J.-Y. Lee. 1999. Automatic word-segmentation at line-breaks for Korean text processing. In *Proceedings of the 6th Conference of Korea Society for Information Mangement*, pages 21–24.

N.-Y. Jeon and H.-R. Park. 2000. Automatic word-spacing of syllable bi-gram information for Korean OCR postprocessing. In *Proceedings of the 12th Conference on Hangul and Korean Information Processing*, pages 95–100.

S.-S. Kang and C.-W. Woo. 2001. Automatic segmentation of words using syllable bigram statistics. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium*, pages 729–732.

S.-S. Kang. 1998. Automatic word-segmentation for Hangul sentences. In *Proceedings of the 10th Conference on Hangul and Korean Information Processing*, pages 137–142.

S.-S. Kang. 2000. Eojeol-block bidirectional algorithm for automatic word spacing of Hangul sentences. *Journal of the Korea Information Science Society*, 27(4):441–447.

J.-D. Kim, H.-S. Lim, S.-Z. Lee, and H.-C. Rim. 1998a. Twoply hidden markov model: A Korean pos tagging model based on morpheme-unit with word-unit context. *Computer Processing of Oriental Languages*, 11(3):277–290.

K.-S. Kim, H.-J. Lee, and S.-J. Lee. 1998b. Three-stage spacing system for Korean in sentence with no word boundaries. *Journal of the Korea Information Science Society*, 25(12):1838–1844.

S.-Z. Lee. 1999. New statistical models for automatic part-of-speech tagging. Ph.D. thesis, Korea University.

B. Merialdo. 1994. Tagging english text with a probabilistic model. *Computational Linguistics*, 20(2):155–172.

Kwangseob Shim. 1996. Automated word-segmentation for Korean using mutual information of syllables. *Journal of the Korea Information Science Society*, 23(9):991–1000.

J.-H. Shin and H.-R. Park. 1997. A statistical model for Korean text segmentation using syllable-level bigrams. In *Proceedings of the 9th Conference on Hangul and Korean Information Processing*, pages 255–260.

Constructing of a Large-Scale Chinese-English Parallel Corpus

Le Sun, Song Xue, Weimin Qu, Xiaofeng Wang,Yufang Sun

Chinese Information Processing Center
Institute of Software, Chinese Academy of Sciences
Beijing 100080, P. R. China
lesun, bradxue, qwm, wxf, yfsun@sonata.iscas.ac.cn

Abstract

This paper describes the constructing of a large-scale (above 500,000 pair sentences) Chinese-English parallel corpus. The current status of Chinese corpora is overviewed with the emphasis on parallel corpus. The XML coding principles for Chinese–English parallel corpus are discussed. The sentence alignment algorithm used in this project is described with a computer-aided checking processing. Finally, we show the design of the concordance of the parallel corpus and the prospect to further development.

Introduction

With the development of the corpus linguistics, more and more language resources have been established and used in language engineering research and applications. As we all know, there are different kinds of corpora for different kinds applications. For example, the Chinese Part-Of-Speech annotation corpus used to train program for Chinese word segmentation and POS tag, the Chinese tree bank used to Chinese syntax study, and so on.

In this paper the constructing of a large-scale Chinese-English parallel corpus, which is totally above 500,000 pair sentences and the first year task is 100,000 pair sentences, is described. The applications of the large-scale Chinese-English parallel corpus put emphasis on the sentence template extracting for EBMT (Example-Based Machine Translation) and translation model training for SBMT (Statistical-Based Machine Translation). The latent applications may include the bilingual lexicon extraction, special term or phase extraction, bilingual teaching, Chinese-English contrastive study, etc.

Numerous corpus data gathering efforts exit all of the world. The rapid multiplication of such efforts has made it critical to create a set of standards for encoding corpora. CES (Corpus Encoding Standard), which is conformant to the TEI Guideline for Electronic Text Encoding and Interchange of the Text Encoding Initiative (TEI 2002), has been adopted by many corpus-based work. The XML Corpus Encoding Standard (XCES) is a part of the Guideline developed by the Expert Advisory Group on Language Engineering Standards (Ide, N., Bonhomme, P., Romary, L. 2000). The coding of our Chinese-English Parallel Corpus is in broad agreement with the TEI Guideline for electronic texts.

In the following section, we first present a brief review of the current status of Chinese corpora with the emphasis on parallel corpus. Then the XML coding principles for Chinese–English parallel corpus are discussed in detail. Following this is the sentence alignment algorithm used in this project with a computer-aided checking processing. Finally, we show the design of the concordance of the parallel corpus and the prospect to further development.

1 Chinese Corpus Project Overview

The Chinese Corpus constructing work started in 1920’s, See Zhiwei Feng (2001). The machine-readable corpora established in 1980’s are listed as following:

- Chinese Modern Literature Corpus (1979), 5.27 Million Chinese Characters, WuHan University;
- Modern Chinese Corpus (1983), 20 Million Chinese Characters, Beijing University of Aeronautics and Astronautics;
- Middle School Chinese Book Corpus (1983), 1.06 Million Chinese Characters, Beijing Normal University;
- Modern Chinese Word Frequency Corpus (1983), 1.82 million Chinese characters, Beijing Language & Culture University.

The first national large-scale Chinese corpus project is proposed in 1991 by State Language Commission in China. The Chinese texts used in this corpus are selected carefully under the condition of times, genre, and field. Now the corpus is about 20 million Chinese characters.

From 1992, there are several large-scale Chinese corpus constructed by different institutes. The most noticeable in them is the Chinese POS annotation corpus accomplished by Institute of Computational Linguistics, Peking University, with the cooperation with Fujitsu Company. The content of this corpus is people's daily, one of the most popular newspapers in China. The Chinese texts are segmented and added POS tag with high precision. The total Chinese Characters are about 27 million.

There are several Chinese corpora in Tsinghua University also. The corpus, which is used for Chinese segmentation study, includes 100 million Chinese characters. The Hua Yu corpus (2 million Chinese characters) is a POS tagged field-balance corpus. And the 10 percent of this corpus has been used for constructing Chinese tree bank.

These are also other valuable Chinese corpora established in ShanXi University, Harbin technical University, ShangHai Normal University, City University of Hong Kong, Taiwan Academia Sinica, University of Pennsylvania and so on. Please refer to Zhiwei Feng (2001) for detail.

In October 2001, a national corpus project, that is, national 863 project about Chinese Information Processing Platform, is launched. It's a cooperation project between five institutes in China, including Institute of Software, Chinese Academy of Sciences, Institute of Computational Linguistics, Peking University, Tsinghua University, Nanjing University and Institute of Language, State Language Commission. The content of corpora and intended scale in this project are showed in table 1 in detail. The large-scale Chinese-English parallel corpus described in this paper is one of the scheming corpora in this project.

The multilingual corpus is important for computational linguistics research and contrastive linguistics study. So there are many multilingual corpus have been established or being developed in many institutes in China mainland. The table 2 shows the Chinese-English parallel corpus had been constructed in Mainland China. There are also some bilingual corpora about other language pair, such as Chinese-Japanese, Chinese-German, etc.

Sub-Project Name	*Responsible Institute*	*First-Year Scale*	*Scheming Scale*
Chinese Balance Corpus	State Language Commission	70 MCC	150 MCC
Chinese-English Parallel Corpus	IOS, Chinese Academy of Science	100 TS	500 TS
Chinese POS Annotation Corpus	ICL, Peking University	7 MCC	30 MCC
Chinese Tree Bank	Tsinghua University	15 TS	60 TS
Chinese Concept Dictionary	ICL, Peking University	20 TC	60 TC
Chinese Semantic Knowledge Base	Tsinghua University	8 TW	24TW

Table 1 The 863 Chinese corpus project

MCC: Million Chinese Character TS: Thousand Sentence
TC: Thousand Concept TW: Thousand Word

Institute	Corpus Describing	Scale
ICL, Peking University	Sentence & Phrase Alignment	5 TS
Harbin Institute of Technology	Sentence, Phrase, Word Alignment	Above 5 TS
State Language Commission	Computer Science and Plato	Unknown
Beijing Foreign Studies University	Literary, Science and Civilization in China	Unknown
Northeastern University	Sentence & Phrase Alignment	Unknown
IOS, Chinese Academy of Science	Sentence Alignment	8 TS

Table 2 The Chinese-English parallel corpus in Mainland

It has been noticed by many scholars that we should build a principle for sharing language resource in research work and to avoid the waste in time and effort in repeated construction.

2 Resource Collection

Unlike single linguistic resource, the parallel resource for special language pair is limited no matter what language pair is. Although the Chinese and English both are most popular language in the world, we still encounter much difficult in obtaining parallel corpus resource from Internet for following reasons:

- There are seldom web pages in China provide the same content in English pages and in Chinese pages;
- The English news in web are translated freely other than literally with many content omission;
- Some bilingual texts are restricted and used only to member.

After two years efforts, there are totally about 16,000KB untagged Chinese-English parallel texts in hand. The genres of the resource we collected are showed in table 3.

Chinese Genre	About Percent
News	10%
Literature	30%
Government Report	25%
Sciences & Technology	35%

Table 3 The genre in parallel texts

3 Coding

3.1 General Principles

The coding of the parallel corpus is in broad agreement with the TEI Guideline for electronic texts. The eXtendible Make-up Language (XML) is used for the text coding. Textual features are marked by tags enclosed within angle brackets. For example, a title is marked by start tag <title> and an end tag </title >. Every element has some attributes to identifier of the element.

The document type definition (DTD) for the texts in the corpus may differs in some respects from the TEI model. The general principle for coding are based on following consideration:

- Comply with TEI guide lines on the whole;
- Define the tag with clear meaning used by most people in china;
- Only used the attributes which can be easily and automatically get from source texts, except the alignment link, which is the key attribute in this corpus and several steps are used to keep high precise (See section 4 for detail);
- Try to keep all the interim resource in hand in case information loses, such as, the title tag in HTML files.

The overall structure of a Chinese-English Parallel corpus is shown by this example:

```
<article id=”UH001”>
<Header type =”Unix Handbook”>
</Header>
<text>
</text>
</article>
```

There are two main parts in a text: a header and the main text. Every text has an unique identifier that is, article id, in this case UH001 (indicating text 001 of the Unix Handbook)

3.2 The header

Each text is described by a header, which has four parts in accordance with the TEI guidelines: a file description, an encoding description, a profile description, and a revision description. The file description gives bibliographical information on the source text. The elements include title, author, www address (If the text is obtain from Internet), etc. The encoding description in our corpus is very brief, only the project name and the DTD file name are listed.

The country or region use the language is indicated in the profile description. The description under <language> used in our corpus is in terms of labels like: Mainland Chinese (MaC), Hong Kong Chinese (HKC), Taiwan Chinese (TwC), Singapore Chinese (SiC), American English (AmE), British English (BrE), Canadian English (CaE), etc.

Another tag used in the profile description is <textclass>. According to the parallel resource in hand, the texts are grouped into 4 genres (as show in table 3), such as, News , Literature, Science & Technical，Government Report.

A series of changes are listed in the revise description and specified the change, the date of the change, the person responsible for the change, and the nature of the change.

3.3 Text Units and Alignment Unit

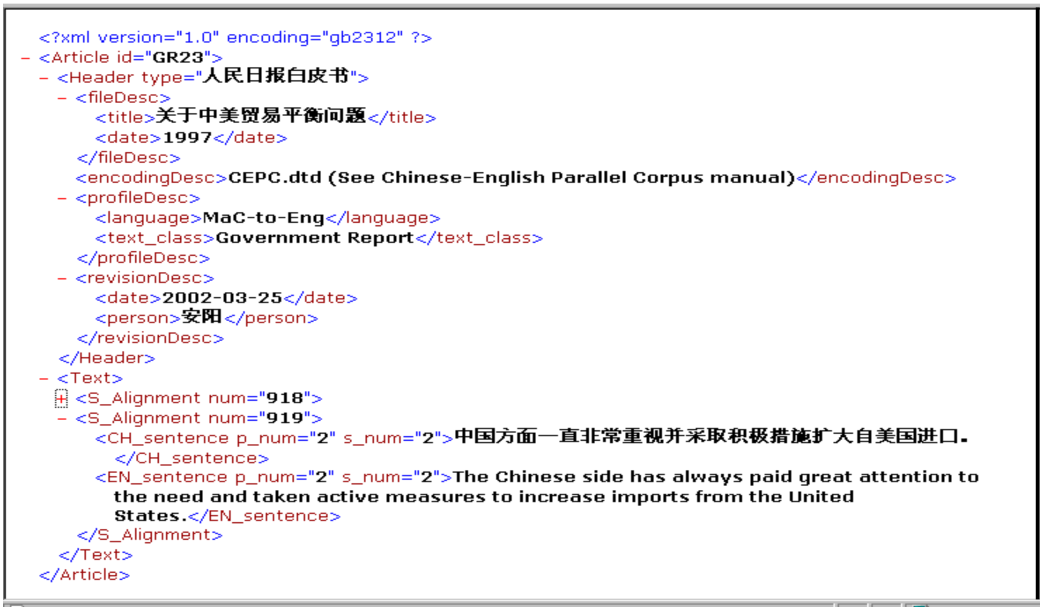
The corpus texts are segmented according to the natural units, such as: chapter, paragraph, sentence (S-unit), and word. The English words are simply marked by spacing as in ordinary written text. The Chinese words are not indicated by space in order to avoid the segment error.

An ID is given to every paragraph to indicate the relative position in whole chapter. The sentence is called S-unit, the same as Johansson, Ebeling and Oksefjell (1999) to underline that they are not necessarily sentences in a grammatical sense.

The sentence alignment type between Chinese S-unit and English S-unit maybe 1:1, 2:1, 3:1, 1:2, 1:3,2:2, 3:2, 2:3. Links between parallel texts are showed by attributes of S-Alignment. One of the Chinese alignment unit (it may beyond one S-unit) are linked with the correspondence English alignment unit.

3.4 Sample Text

A sample text of our Chinese-English parallel corpus is showed in figure 1.



```
   <?xml version="1.0" encoding="gb2312" ?>
 - <Article id="GR23">
   - <Header type="人民日报白皮书">
     - <fileDesc>
         <title>关于中美贸易平衡问题</title>
         <date>1997</date>
       </fileDesc>
       <encodingDesc>CEPC.dtd (See Chinese-English Parallel Corpus manual)</encodingDesc>
     - <profileDesc>
         <language>MaC-to-Eng</language>
         <text_class>Government Report</text_class>
       </profileDesc>
     - <revisionDesc>
         <date>2002-03-25</date>
         <person>安阳</person>
       </revisionDesc>
     </Header>
   - <Text>
     + <S_Alignment num="918">
     - <S_Alignment num="919">
         <CH_sentence p_num="2" s_num="2">中国方面一直非常重视并采取积极措施扩大自美国进口。
           </CH_sentence>
         <EN_sentence p_num="2" s_num="2">The Chinese side has always paid great attention to
           the need and taken active measures to increase imports from the United
           States.</EN_sentence>
       </S_Alignment>
     </Text>
   </Article>
```

Figure1 Sample Text

4 Sentence Alignment

4.1 Algorithm Overview

The key attribute in this corpus is alignment link, which connect the one or more Chinese sentence with one or more correspond English sentence. In order to keep high precise in sentence alignment, several steps are used with the human and computer cooperation.

The first step to extract structural information for parallel corpus is paragraph alignment and sentence alignment, that is noting which paragraph and sentence in one language correspond to which paragraph and sentence in another language.

This problem has been studied by many researchers and a number of quite encouraging results have been reported. However, almost all bilingual corpora used in research are clear (nearly without sentence omission or insertion) and literal translation bilingual texts. The performance tends to deteriorate significantly when these approaches are applied to noisy complex corpora (with sentence omission or insertion, less literal translation).

There are basically three kinds of approaches on sentence alignment: the length-based approach (Gale & Church 1991 and Brown et al. 1991), the lexical approach (key & Roscheisen 1993), and the combination of them (Chen 1993, Wu 1994 and Langlais 1998, etc.).

The first published algorithms for aligning sentences in parallel texts are length-based approach proposed by Gale & Church (1991) and Brow et al (1991). Based on the observation that short sentences tend to be translated as short sentences and long sentences as long sentences, they calculate the most likely sentence correspondences as a function of the relative length of the candidates. The basic approach of Brow et al. is similar to Gale and Church, but works by comparing sentence length in words rather than characters. While the idea is simple, the models can still be quite effective when used to clear and literal translated corpora. Once the algorithm had accidentally mis-aligned a pair sentence, it tends to be unable to correct itself and get back on track before the end of the paragraph. Use alone, length-based alignment algorithms are therefore neither very robust nor reliable.

Kay & Roscheisen (1993) use a partial alignment of lexical items induce a maximum likelihood at sentence level. The method is reliable but time consuming.

Chen (1993) combines the length-based approach and lexicon-based approach together. A translation model is used to estimate the cost of a certain alignment, and the best alignment is found by using dynamic programming as the length-based method. The method is robust, fast enough to be practical and more accurate than previous methods.

The first sentence alignment model used to align English-Chinese bilingual texts is proposed by Wu (1994). For lack of cognates in English-Chinese, he used lexical cues to add the robust of his model.

All of these works are test on nearly clear and literal translation bilingual corpora.

There are seldom papers related to paragraph alignment. It's believed by most of the researchers that the paragraph alignment is an easier task than sentence alignment. Gale & Church (1991) suggest that the same length-based algorithm can be used to align paragraph also.

4.2 The Alignment Steps

Sentence alignment algorithm of our system can be outlined as follows:

Step 1: Align sentence by the improved length-based algorithm.(Desicibed in Sun etc. 1999)

Step 2: A lexicon checking process is added to judge all the alignment results in step 1. A score is given to every alignment pair (A Chinese word segmentation system is used in this process to find Chinese word).

Step 3: The alignments whose score above a threshold C_1 are judged as correct alignment. Remove these correct alignments from bilingual texts temporally.

Step 4: The rest parts are aligned again by length based approach.

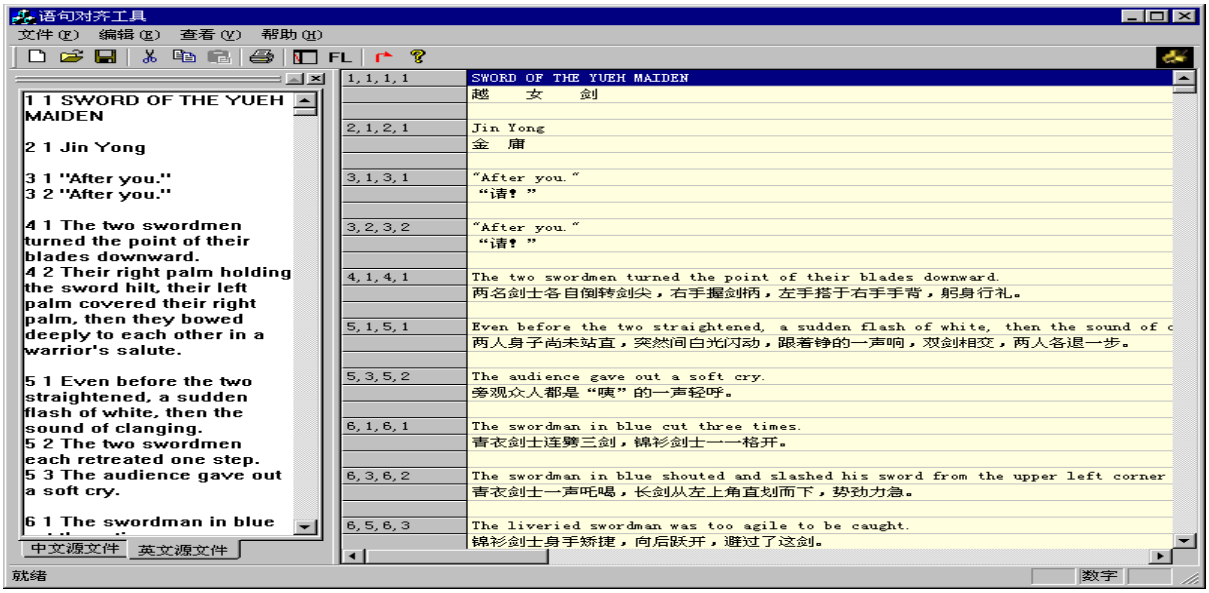
Step 5: Repeat step 2, the score of every lignment is showed as a reference to human checking.

4.3 Computer-Aided Checking

It's obviously difficult to increase greatly the accuracy and robust of sentence alignment only by length based approach. So a lexicon checking process is added to our system. The alignment results obtained by length based approach are checked by an English-Chinese lexicon. A score S_A is given to every alignment sentence pair. The score S_A is calculated by following idea, that is, the twice number of correctly matched English words and Chinese words to the sum of number of English and Chinese words. In figure 2, the interface for human checking is showed in order to processes the noise Chinese-English parallel resource.

4.4 Experiment Results

We tested our alignment algorithm with part of a computer handbook (Sco Unix handbook). There are about 4681 English sentences and 4430 Chinese sentences in this computer handbook after filter noisy figures and tables. The detail experiment result of automatic sentence alignment is show in table 4. The total precision is about 95%.



1 1 SWORD OF THE YUEH MAIDEN

2 1 Jin Yong

3 1 "After you."
3 2 "After you."

4 1 The two swordmen turned the point of their blades downward.
4 2 Their right palm holding the sword hilt, their left palm covered their right palm, then they bowed deeply to each other in a warrior's salute.

5 1 Even before the two straightened, a sudden flash of white, then the sound of clanging.
5 2 The two swordmen each retreated one step.
5 3 The audience gave out a soft cry.

6 1 The swordman in blue

1, 1, 1, 1	SWORD OF THE YUEH MAIDEN 越　女　剑
2, 1, 2, 1	Jin Yong 金　庸
3, 1, 3, 1	"After you." "请！"
3, 2, 3, 2	"After you." "请！"
4, 1, 4, 1	The two swordmen turned the point of their blades downward. 两名剑士各自倒转剑尖，右手握剑柄，左手搭于右手手背，躬身行礼。
5, 1, 5, 1	Even before the two straightened, a sudden flash of white, then the sound of c 两人身子尚未站直，突然间白光闪动，跟着铮的一声响，双剑相交，两人各退一步。
5, 3, 5, 2	The audience gave out a soft cry. 旁观众人都是"咦"的一声轻呼。
6, 1, 6, 1	The swordman in blue cut three times. 青衣剑士连劈三剑，锦衫剑士一一格开。
6, 3, 6, 2	The swordman in blue shouted and slashed his sword from the upper left corner 青衣剑士一声吒喝，长剑从左上角直划而下，势劲力急。
6, 5, 6, 3	The liveried swordman was too agile to be caught. 锦衫剑士身手矫捷，向后跃开，避过了这剑。

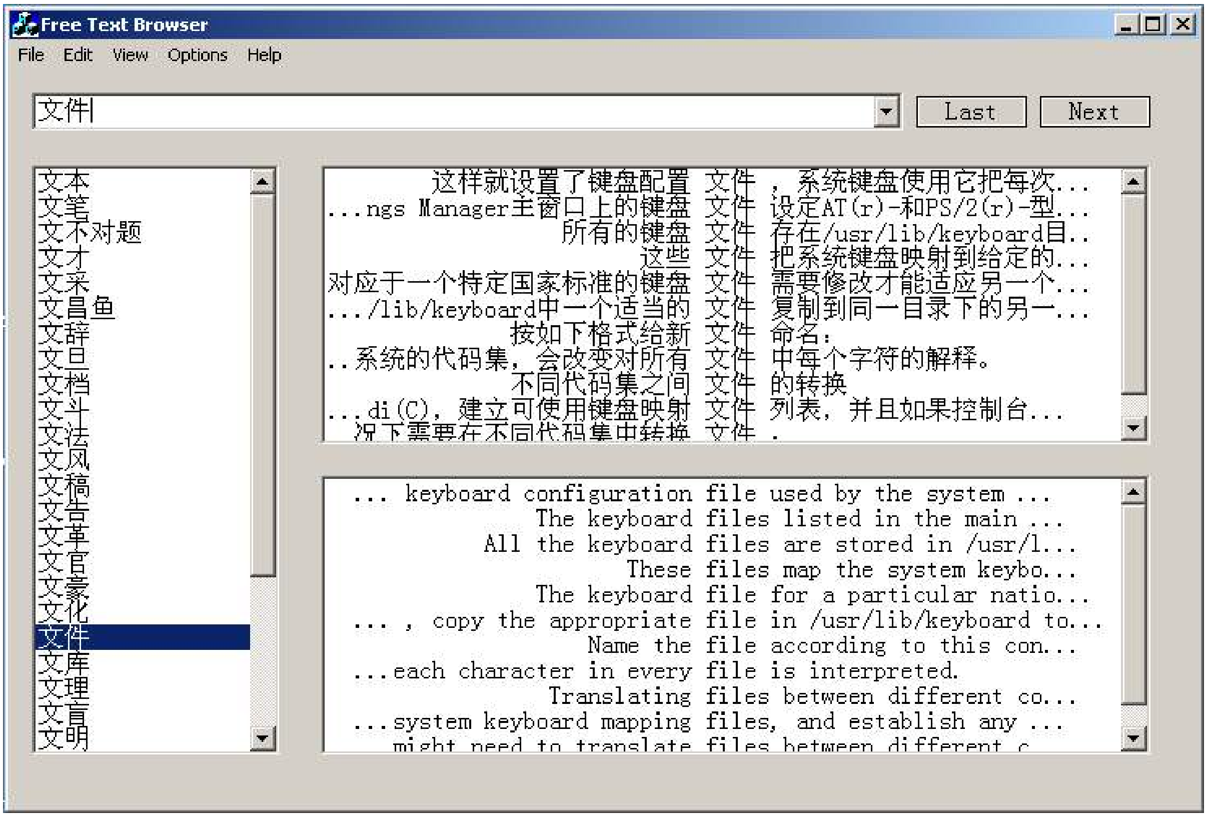
Figure 2 Interface for Human Checking

Class of Alignment	No. of Aligned Sentence Pair	No. of Correct Sentence Pair	No. of Error Sentence Pair	Precision
1:1	2992	2957	35	98.83%
1:2	238	211	27	88.66%
2:1	414	352	62	85.02%
2:2	113	97	16	85.84%
1:3	35	24	11	68.57%
3:1	75	49	26	65.33%
2:3	13	6	7	46.15%
3:2	22	16	6	72.72%
3:3	6	3	3	50.00%
0:1	3	2	1	66.67%
1:0	7	4	3	75.00%
Total	3918	3721	197	94.97%

Table 4 The detail experiment result of automatic sentence alignment

5 Bilingual Concordance Design

We also designed a bilingual concordance tool used for discovering facts during the translation between Chinese and English. Besides a listing of the keywords with the contexts in which they appear, the correspondence translation sentence also be presented in this tool. The options may include bilingual concordances, sorting in a variety of orders, and producing basic text statistics. The intended interface is showed in figure 3.



```
文本
文笔
文不对题
文才
文采
文昌鱼
文辞
文旦
文档
文斗
文法
文风
文稿
文告
文革
文官
文豪
文化
文件
文库
文理
文盲
文明
```

```
          这样就设置了键盘配置 文件 ，系统键盘使用它把每次...
...ngs Manager主窗口上的键盘 文件 设定AT(r)-和PS/2(r)-型...
                  所有的键盘 文件 存在/usr/lib/keyboard目..
                        这些 文件 把系统键盘映射到给定的...
对应于一个特定国家标准的键盘 文件 需要修改才能适应另一个...
.../lib/keyboard中一个适当的 文件 复制到同一目录下的另一...
              按如下格式给新 文件 命名：
..系统的代码集，会改变对所有 文件 中每个字符的解释。
              不同代码集之间 文件 的转换
...di(C)，建立可使用键盘映射 文件 列表，并且如果控制台...
  况下需要在不同代码集中转换 文件 .
```

```
  ... keyboard configuration file used by the system ...
               The keyboard files listed in the main ...
          All the keyboard files are stored in /usr/l...
                     These files map the system keybo...
               The keyboard file for a particular natio...
  ... , copy the appropriate file in /usr/lib/keyboard to...
                     Name the file according to this con...
  ...each character in every file is interpreted.
                 Translating files between different co...
  ...system keyboard mapping files, and establish any ...
     might need to translate files between different c
```

Figure 3 The Interface for bilingual Concordance

6 Conclusion & Further Prospects

In this paper, we introduce the developing project, that is, the constructing of a large-scale (above 500,000 pair sentences) Chinese-English parallel corpus. The current status of Chinese corpora is overviewed with the emphasis on parallel corpus. The XML coding principles for Chinese–English parallel corpus are discussed. The sentence alignment algorithm used in this project is described with a computer-aided checking processing in order to processes the noise Chinese-English parallel resource.. We show the design of the bilingual concordance for the parallel corpus, also.

As a beginning project, there is still much room for further development. The parallel resource is relative rare, so the new ways, such as, data exchange with other researcher institute and translation company, should be launched to obtain more parallel resource which can be used to research society. The coding principle should be adjusted in real work. A coding rule in more detail should form in near future. We also intend to add the option for recommendation the correspondence translation word for input keywords in concordance tool.

Acknowledgements

This work is supported by China 863 project (Grant No. 2001AA114040) and the National Science Fund of China under contact 69983009. Our thanks go to all the project members from

five institutes for discussion and the anonymous reviewers for kind suggestions. .

References

Catherine N. Bal (1997), Tutorial: Concordances and Corpora, http://www.georgetown.edu/cball /corpora/tutorial.html

D. Wu, (1994) *Aligning a Parallel English-Chinese Corpus Statistically with Lexical Criteria*, In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL'94), pp.80-87

I. D. Melamed. (1996) *Automatic Detection of Omissions in Transaltions,* In Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, Denmark

ISLE, International Standards for Language Engineering http://www.ilc.pi.cnr.it/EAGLES96/isle/ISLE _Home_Page.htm

J.S. Chang and M. H. Chen (1997) *An alignment method for noisy parallel corpora based on image processing techniques,* In Proceedings of the 35th Meeting of the Association for Computational Linguistics, Madrid, pp. 297-304

Kay M., and Roscheisen M. (1993). *Text-Translation Alignment*, Computational Linguistics, 19/1,pp.121-142

Le Sun , Lin Du, Yufang Sun, Jin Youbin (1999) *Sentence Alignment of English-Chinese Complex Bilingual Corpora.* Proceeding of the workshop MAL'99, 135-139

N. Ide, L. Romary (2001). A Common Framework for Syntactic Annotation *Proceedings of ACL'2001,* Toulouse, 298-305

N. Ide, L. Romary, (2000) XML Support for Annotated Language Resources. *Proceedings of the Workshop on Web-based Language Documentation and Description,* Philadelphia, 148-153.

N. Ide, P. Bonhomme,, L. Romary (2000). XCES: An XML-based Standard for Linguistic Corpora.. *Proceedings of the Second Language Resources and Evaluation Conference* (LREC), Athens, Greece, 825-30.

P. F. Brown, J. C. Lai, and R. L. Mercer (1991) *Aligning Sentences in Parallel Corpora*, In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL'91), pp.169-176.

P. Fung, and K. W. Church (1994) *K-vec: A New Approach for Aligning Parallel Texts*, In Proceedings of the 15th International Conference on Computational Linguistics (COLING'94), Tokyo, Japan, pp. 1096-1102,

Ph. Langlais, M. Simard, J. Veronis, S.Armstong, P. Bonhomme, F. Debili, P. Isabelle, E. Souissi, and P. Theron. (1998) *Arcade: A cooperative research project on parallel text alignment evaluation*. In First International Conference on Language Resources and Evaluation, Granada, Spain.

S. F. Chen, (1993) *Aligning Sentences in Bilingual Corpora Using Lexical Information*. In Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics, pp. 9-16

Shiwen Yu, Xuefeng Zhu, Hui Wang, Yunyun Zhang (1998), *The Grammatical Knowledge-base of Contemporary Chinese: A complete Specification.* Tsinghua University Publishers

Stig Johansson, Jarle Ebeling, Signe Oksefjell (1999), English-Norwegian Parallel Corpus:Manual, http://www.hf.uio.no/iba/prosjekt/

TEI (2002) The XML Version of the TEI Guidelines http://www.hcu.ox.ac.uk/TEI/Guidelines/

W. A. Gale, and K. W. Church (1991) *A Program for Aligning Sentences in Bilingual Corpora*, In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL'91), pp. 177-184

Zhiwei Feng (2001), *The History and Current status of Chinese Corpus Research*, International Conference on Chinese Computing ICCC2001, pp. 1-15 (In Chinese)

AnnCorra : Building Tree-banks in Indian Languages

Akshar Bharati
Rajeev Sangal
Vineet Chaitanya
Amba Kulkarni
Dipti Misra Sharma
International Institute of Information Technology
Hyderabad, india
{sangal, vc, amba, dipti}@iiit.net

K.V. Ramakrishnamacharyulu
Rashtirya Sanskrit Vidyapeetha, Tirupati, India
kvrk@sansknet.org

ABSTRACT

This paper describes a dependency based tagging scheme for creating tree banks for Indian languages. The scheme has been so designed that it is comprehensive, easy to use with linear notation and economical in typing effort. It is based on Paninian grammatical model.

1.BACKGROUND

The name AnnCorra, shortened for "Annotated Corpora", is for an electronic lexical resource of annotated corpora. The purpose behind this effort is to fill the lacuna in such resources for Indian languages. It will be an important resource for the development of Indian language parsers, machine learning of grammars, lakshancharts (discrimination nets for sense disambiguation) and a host of other such tools.

2. AIMS AND OBJECTIVE

The aim of the project is to :

- develop a generalised linear syntacto- semantic tag scheme for all Indian languages
- annotate training corpus for all Indian languages
- develop parallel tree-banks for all Indian languages

To fulfill the above aim - a marathon task - a collaborative model has been concieved. Any collaborative model implies involvement of several people with varying levels of expertise. This case, becomes further complicated as the tag scheme to be designed has to be equally efficient for all the Indian languages. These languages, though quite similar, are not identical in their syntactic structures. Thus the tag scheme demands the following properties :-

- comprehensive enough to capture various sysntactic relations across languages.
- simple enough for anyone, with some background in linguistics, to use.
- economical in typing effort (the corpus has to be manually annotated).

3. AN ILLUSTRATION

The task can be better understood with the help of an illustration. Look at the following sentence from Hindi

0:: **rAma ne moHana ko**
'Rama' 'ErgPostP' 'Mohan' 'PostP'

nIlI kitAba dI
'blue' 'book' 'gave'

'Rama gave the blue book to Mohan.'

Tree-1 is a representation of the above verb, argument relationship within the various constituents of sentence 0 -

```
                 dI
         ------------------------
         |         |            |
      k1 |       k4|          k2|
         |         |            |
  rAma_ne  moHana_ko   kitAba
                                |
                                |nmod
                                |
                              nIlI
```

Tree-1

Since the input for tagging is a text corpus and the marking has to be done manually, the tagging scheme is linearly designed. Therefore, Sentence 0 will be marked as follows -

rAma_ne/k1 moHana_ko/k4 [nIlI
'Ram postp' Mohan postp' 'blue '

kitAba]/k2 dI::v
'book' 'gave'

The markings here represent

- **'di'** ('give') is the verb node
- **'rAma_ne'** is the 'karta' or 'agent' (k1) of the verb 'dI',
- **'moHana_ko'** is 'sampraadana' or 'beneficiary' (k4) of verb 'dI' ('give')
- '**[nIlI kitAba]**' – (blue book) a noun phrase - is the 'karma' or 'object' (k2) of the verb.

The elements joined by an underscore represent one unit. Postpositions which are separated by white space in the written texts are actually the inflections of the preceding noun or verb units. Therefore, they are conjoined.

The modifier-modified elements are paranthesised within square brackets. Tags showing the name of the ARC (or branch) are marked by '/' immediately after the constituent they relate to. '/' is followed by the appropriate tagname.

Thus '/' specifies a relationship of a word or constituent with another word or constituent. In this case it is the relationship of verb 'dI' with the other elements in the sentence.

Tags denoting a type of node are marked by '::'. '::v' indicates that 'dI' is a verbal node.

The idea here is to mark only the specific grammatical information. Certain DEFAULT CONVENTIONS are left unmarked. For example, the adjective 'nIlI' ('blue') of 'kitAba' ('book) has been left unmarked in the above example since normally noun modifiers precede the noun they modify (adjectives precede nouns). Such DEFAULT CONVENTIONS save unnecessary typing effort.

4. GRAMMATICAL MODEL

It was quite natural to use Paninian grammatical model for sentence analysis (hence the tagnames) because :-

1) Paninian grammatical model is based on the analysis of an Indian language (Sanskrit) it can deal better with the type of constructions Indian languages have.

2) The model not only offers a mechanism for SYNTACTIC analysis but also incorporates the SEMANTIC information (nowadays called dependency analysis). Thus making the relationships more transparent. (For details refer Bharati (1995).)

Following tags (most of which are based on Paninian grammatical model) have been used in the above example.

k1 : kartaa (subject or agent)
k2 : karma (object)
k4 : sampradaana (beneficiary)
v : kriyaa (verb)

Obviously the task is not an easy one. Standardization of these tags will take some time. Issues, while deciding the tags, are many. Some examples are illustrated below to show the kind of structures which the linear tagging scheme will have to deal with.

4.1. Multiple Verb Sentences

To mark the nouns-verb relations with the above tags in single verb sentences is a simple task. However, consider the following sentence with two verbs :-

1: **rAma ne khAnA khAkara**
am' 'postp' 'food' 'having_eaten'

pAnI pIyA
'water' 'drank'

`Ram drank water after eating the food.`

Sentence 1 has more than two verbs - one non-finite (khAkara) and one finite (piyA). The finite verb is the main verb. Noun 'khAnA' is the object of verb 'khAkara', whereas noun 'pAnI' is the object of verb 'piyA'. 'k2' is the tag for object relation in our tagging scheme. Co-indexing becomes the obvious solution for such multiple relations. Since there are two verbs the tagging scheme allows them to be named as 'i' and 'j' (using notation 'i' and 'j'). By default 'i' refers to the main verb and any successive verb by other characters ('j' in the present case):

rAma_ne khAnA khAkara::vkr:j
'Ram_postp' 'food' 'having_eaten:j'

pAnI piyA::v:i
'water' 'drank:i'

This provides the facility to mark every noun verb relationship.

rAma_ne/k1>i khAnA/k2>j khAkara::vkr:j pAnI/k2>i piyA::v:i

Fortunately, there is no need to mark it so "heavily". A number of notations can be left out, and the DEFAULT rules tell us how to interpret such "abbreviated' annotation. Thus, for the above sentence, the following annotation is sufficient and is completely equivalent to the above :

rAma_ne/k1 khAnA/k2 khAkara::vkr:j pAnI/k2 piyA::v

Even though there are two verbs, there is no need to name the verbs and refer to them. Two default rules help us achieve such brevity (without any ambiguity) :

(1) karta or k1 kaaraka always attaches to the last verb in a sentence (Thus 'rAma_ne/k1' attaches to the verb at the end).
(2) all other kaarakas except k1, attach to the nearest verb on the right. Thus 'khAnA/k2' attaches to 'khAkara' and 'pAnI/k2' attaches to 'piyA', their respective nearest verbs on the right.

4.2. Compound Units

Sometimes two words combine together to form a unit which has its own demands and modifiers, not derivable from its parts. For example, a noun and verb join together to operate as a single unit, namely as a verb. In the sentence 'rAma (Rama) ne (postp) snAna(bath) kiyA (did)', 'snAna' and 'kiyA' together stand for a verb 'snAna+kiyA' (bathed). Such verbal compounds are like any other verb having their own kaarakas.This sentence would be marked as follows :

rAma_ne/k1 snAna::v+ kiyA::v-
'Ram_postp' 'bath+' 'did-'

`Ram took a bath`

A 'v+' or a 'v-' indicates that the word 'snAna' or 'kiyA' are parts of a whole (a verb in this case). Taken together they function as a single verb unit. Such a device which may appear to be more powerful was needed to mark the 'single unitness' of parts which may appear separately in a sentence. Thus, the above notation allows even distant words to be treated as a single compound. Such occurrences are fairly common in all Indian languages as illustrated in the following example from Hindi :

snAna::v+ to mEMne/k1
'bath' 'emph' 'I_erg'

subaHa_HI kara_liyA_thA::v-
'morning_emph' 'had_done'

I had bathed (taken a bath) in the morning itself.

'+'and ' - ' help in marking this relation explicitly. (a more detail description of the notation in 5.1)

4.3. Embedded Sentence

Tags are also designed to mark the relations within a complex sentence. Consider the example below where a complete sentence (having verb 'piyA' (drank)) is a kaaraka of the main verb 'kaHA' (said).

moHana ne kaHA ki {rAma
'Mohan' 'postp' 'said' 'that ' {'Rama'

ne pAnI khAnA khAkara
'postp' 'water' 'food' 'having eaten'

piyA}.
'drank}

(Mohan said that Ram drank water after having eaten the food)

The embedded sentence can be first marked as follows -

--------- {rAma_ne/k1 pAnI/k2>j khAnA/k2 khAkara::vkr piyA::v:j}::s.

The whole embedded sentence is the 'karma' (object) or k2 of 'piyA' (drank): The relation of the embedded sentence relation as the object of the main verb is co-indexed in the following way :-

moHana_ne kaHA::v:i ki
'Mohan_postp' 'said' 'that'

rAma_ne/k1 pAnI/k2>j khAnA/k2
'Rama_postp' 'water' 'food'

khAkara::vkr piyA::v:j::s/k2>i
'having_eaten' 'drank'

Thus the device of naming the elements and co-indexing them with their respective arguments can be used most effectively.

5. TAGGING SCHEME

The tagging scheme contains : notations, defaults, and tagsets.

5.1. NOTATION

Certain special symbols such as double colon,underscore, paranthesis etc. are introduced first. Two sets of tags have been provided (to mark the crucial ARC and node information). However, apart from these symbols and tags, some special notation is required to explicitly mark certain disjointed, scattered and missing elements in a sentence. Following notation is adopted for marking these elements :-

5.1. 1. X+ ... X- : disjointed elements

As shown above (4.2), when a single lexical unit composed of more than one elements is separated by other intervening lexical units, its 'oneness' is expressed by using '+' on the first element in the linear order and '-' on the second element. '+' indicates to look ahead for the other part till you find an element with '-'. '-' suggests, 'an element marked '+' is left behind, to which it should get itself attached'.

Example - Verb **'snAna_karanA'** (to bathe) in Hindi can occur disjointedly

snAna to mEMne kiyA_thA
'bath' 'emph' 'I' 'did'

para phira gaMdA Ho_gayA
'but' 'again' 'dirty' 'became'

`Bathe I did , but got dirty again.'

'snAna_karanA' is one verb unit in Hindi. But its two components 'snAna' and 'karanA' can occur separately. Notation 'X+....X-' can capture the 'oneness' of these two elements. So **'snAna.karanA'** (‘bathe’) in the above sentence would be marked as follows :

snAna::v+ to mEMne
'bath' 'emph' 'I'

kiyA_thA::v- para phira gaMdA
'did' but' 'again' 'dirty'

Ho_gayA
'became'

Another example of 'scattered elements' is 'agara .... to' construction of Hindi.

agara tuma kaHate to mEM
'if' 'you' 'said' 'then' 'I'

A_ jAtA
'would_have_come'

`Had you asked I would have come'

‘agara' and 'to' together give the 'conditionality' sense. Though they never occur linearly together they have a 'oneness' of meaning. Their dependency on each other can also be expressed through 'X+....X-' notation.

agara::yo+ tuma kaHto::yo- mEM A_ jAtA (tag 'yo' is for conjuncts)

5.1.2. >i ....:i : explicitly marked dependency (:i is the head)

(a) Example -- The sentence 1a below has the dependency structure given in T-2

1a. **phala rAma ne**
'fruit' 'Rama' 'Ergpostp'

naHA_ kara khAyA
'having_bathed' 'ate'

' Rama ate the fruit after taking a bath'

```
                 khAyA
                   |
       |----------|-----------------|
   k1 |       naHA_kara::vkr    |k2
       |                          |
     rAma_ne                 phala
```

T.2

Default (5.2.5) states that all kaarakas attach themselves to the nearest available verb on the right. In (1a) above, the nearest verb available to 'phala' (fruit) is 'naHA_kara'. However, 'phala' (fruit) is not the 'k2' of 'naHA_kara'. It is the 'k2' of the main verb 'khA'. Therefore, an explicit marking is required to show this relationship. The notation '>i...:i' makes this explicit. Therefore,

phala/k2>i rAma_ne naHA_kara khAyA::v:i

Where 'khAyA' is the 'head', thus marked ':i' and 'phala' is the dependent element, thus marked '>i'. An element marked '>i' always looks for another element marked ':i'.

(b) Another example of such attachments which need to be marked explicitly is given below -

2a. **rAma, moHana Ora shyAma**
'Rama', 'Mohan' 'and' 'Shyama'
Ae
'came'

```
                  Ora
                   |
        |----------|---------------|
        |          |               |
      rAma     moHana    shyAma
```

T-3

To show their attachment to 'Ora' (and) the three elements 'rAma','moHana', 'shyAma' have to be marked (as in 2b.) the following way in our linear tagging scheme.

rAma>i, moHana>i Ora::yo:i shyAma>i

The justification to treat 'Ora' as the head and show the 'wholeness' of all the elements joined by '>i' to ':i' is made explicit by the following examples-

rAma, Ora Haz, moHana Ora
'Rama' 'and''yeah', 'Mohana' 'and'

shyAma Ae_ the
'Shyama' 'had_come'

In this case there is an intervening element 'Ora HAz' (‘and_yeah) between 'rAma' and 'moHana' etc. So paranthesis alone will not resolve the issue of grouping the constituents of a whole. (By paranthesising, elements which are not part of the whole will also be included.) To avoid this the 'Ora' (and) has to be treated as a head.

5.1.3. 0 : explicit marking of an ellipted element (missing elements). Example -

rAma bAjZAra gayA, moHana
'Rama' 'market' 'went' 'Mohana'

ghara Ora Hari skUla
'home' 'and' 'Hari' 'school'

‘Rama went to the market, Mohana home and Hari to the school.’

The sentence above has two ellipted elements. The second and third occurrence of the verb 'gayA'(‘went’). To draw a complete tree the information of the missing elements is crucial here. Arguments 'moHana', 'ghara', 'Hari', and 'skUla' are left without a head, and their dependency cannot be shown unless we mark the 'ellipted' element.

rAma bAjZAra gayA, moHana
'Rama' 'market' 'went', 'Mohana'

ghara 0 Ora Hari skUla 0
'home' 'and' 'Hari' 'school'

In cases where this information can be retrieved from some other source (DEFAULT) it need not be marked. In the above case it need not be marked. However, there may be cases where marking of the missing element is crucial to show various relationships. In such cases it has to be marked. Look at the following example -

eka Ora sajjana
'one' 'more' 'gentleman'

kaHate_HEM bacce baDZe
'says' 'children' 'big'

Ho_gaye_ HEM kisI
'become' 'nobody'

kI bAta naHIM mAnate
'gen' 'saying' 'not' 'agree'

' One more gentleman says that the kids have grown older and do not listen to anybody.'

The above sentence does not have any explicit 'yojaka(conjunct)', between two sentences,

a) **bacce baDZe Ho gaye HEM** and
`kids have grown older'

b) **kisI kI bAta naHIM mAnate**
`do not listen to anybody'

Both these sentences together form the 'vAkyakarma(sentential object)' of the verb 'kaHate HEM' (‘say’).

So the analysis would be -

[eka Ora sajjana]/k1 kaHate_HEM::v:i
‘one’ ‘more’ ‘gentleman’ ‘says’
{{bacce/k1ud baDZe/k1vid
‘children’ ‘big’
Ho_gaye_HEM::v}::s {kisI_kI/6
‘become’ ‘nobody’_s’

bAta]/k2 naHIM::neg
‘words’ ‘not’
mAnate::v}::s}/k2>I
‘listen’

It appears to be a neatly tagged sentence. However, some crucial information is missing from this analysis. In the sentence the relationship between the two sentences within the larger sentential object is not expressed. The problem now is how to do it. Use of '>i...:i' notation can help express this. However, it needs the ':i' information and since there is no explicit 'yojaka' (conjunct) element between the two sentences it will not be possible to mark it. The information of the presence of a 'yojaka' (conjunct) which is the head of a co-ordinate structure is CRUCIAL here. Without its presence its dependency tree cannot be drawn. The notation '0' can be of help in such situations. '0' can be marked in the appropriate place. This will allow the tagging of the dependent elements. Therefore, the revised tagging would be -

[eka Ora sajjana]/k1 kaHate_HEM::v:i

{{bacce/k1ud baDZe/k1vid

ho_gaye_HEM::v}::s>j 0::yo:j

{kisI_kI/6 bAta]/k2 naHIM::neg

mAnate::v}::s>j}/k2>i

Here the information of missing conjunct has been marked by a '0'.

5.2. DEFAULTS

Apart from tagsets and special notations the scheme also relies on certain defaults. Defaults have been specified to save typing by the human annotator. For example, no sentence has to be marked ba a sentence tag till it is crucial for the dependency analysis. For example :

rAma ne yaHa socA ki
'Rama' 'postp' 'this' 'thought' 'that'
moHana AegA
'Mohana' 'would_come'

`Rama thought that Mohana would come'

This is a complex sentence where the subordinate sentence is the object complement of the verb 'socA'(‘thought’) . To indicate the relation of the subordinate clause with the main verb, it has to marked.

Similarly, within the square paranthesis, right most element is the Head. So there is no need to mark it. Postpositions's attachment to the previous noun is also covered by the default rule. There are other defaults which take care of modifier-modified relationships. In short, the general rules have been accounted for by defaults and only the specific relations have to be marked. Elements preceding the head within paranthesis are to be accepted as modifiers of the head. However, In case the number of elements within paranthesis is more than two (Head plus two) and one or more of them do not modify the head then it should be marked.

Example - **[HalkI nIlI kitAba**],
'light' 'blue' 'book'

Here, 'halkI'(‘light’) can qualify both 'nIlI'(‘blue’) and 'kitAba'(‘book’). In case it is modifying 'kitAba'(‘book’), say, in terms of light weight, then it should be left unmarked. But if it modifies 'nIlI'(‘blue’), in terms of light shade, then it SHOULD be marked by adding '>' on the right of the modifying element.

'halkI' [HalkI> nIlI kitAba].
‘light’ [‘light’> ‘blue’ ‘book’]

Let us look at another case where the dependency has to be explicitly marked. Participle form 'tA_HuA', in Hindi, can modify either a noun or a verb. For example take the Hindi sentence -

mEMne/k1 dODZate_Hue::vkr
'I_erg' 'running'

ghoDZe_ko/k2 dekhA::v
'horse' 'saw'

This ambiguous sentence may mean either the following :-
a**) mEMne dODZate_Hue::vkr>i ghoDZe_ko:i/k2 dekhA ;**

'I saw the horse while the horse was running'
Or

b) mEMne dODZate_Hue::vkr>i ghoDZe_ko/k2 dekhA::v:I

'While I was running I saw the horse'

There is no need to mark ':i' in sentence (a). However (b) will need explicit marking.

5.3.TAGSETS

The tagsets used here have been divided into two categories -

1) TAGSET-1 - Tags which express relationships are marked by a preceding '/' . For example kaarakas are grammatical relationships, thus they are marked '/k1', '/k2', '/k3' etc.

2) TAGSET-2 - Tags expressing nodes are marked by a preceding '::' verbs etc. are nodes, so they will be marked '::v',

Certain conventions regarding the naming of the tags are ;

k = kaaraka, -- all the kaaraka tags will begin with k-,
Therefore, k1, k2, k3 etc.
n = noun
v = verb -- eg. v, vkr etc.

6. CONCLUSIONS

A tagging scheme has been designed to annotate corpora for various Indian languages. The objective has been to use uniform tags for all the Indian languages thereby evolving a standard which can be followed for various syntactic analysis for machine processing. The scheme is yet to be yet implemented on corpora from various languages. Some trial workshops have been conducted to see its applicability in other Indian languages. However, once the actual task of tagging begins one may come across cases which are not covered by the present scheme. The idea is to provide a basic scheme which can later be improved and revised.

7. REFERENCES

Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, "Natural Language Processing: A Paninian Perspective", Prentice-Hall of India,
New Delhi, 1995.

Bharati, Akshar, Dipti M Sharma, Vineet Chaitanya, Amba P Kulkarni,
Rajeev Sangal, Durgesh D Rao, LERIL : Collaborative Effort for Creating
Lexical Resources, In Proc. of Workshop on Language Resources in
Asian Languages, together with 6th NLP Pacific Rim Symposium, Tokyo.

OLACMS: Comparisons and Applications in Chinese and Formosan Languages

Ru-Yng Chang
Institute of Linguistics, Academia Sinica
130 Sec.2 Academy Rd.
Nankang, Taipei, Taiwan, 115
ruyng@gate.sinica.edu.tw

Chu-Ren Huang
Institute of Linguistics, Academia Sinica
130 Sec.2 Academy Rd
Nankang, Taipei, Taiwan, 115
churen@gate.sinica.edu.tw

Abstract

OLACMS (stands for Open Language Archives Community Metadata Set) is a standard for describe language resources. This paper provides suggestion to OLACMS 0.4 version by comparing it with other standards and applying it to Chinese and Formosan languages.

1 Introduction[1]

The Open Language Archives Community (OLAC, http://www.language-archives.org) is an international partnership of institutions and individuals who are creating a worldwide virtual library of language resources by: (1) developing consensus on best current practices for the digital archiving of language resources; (2) developing a network of interoperating repositories and services for housing and accessing such resources.

Three primary standards are the foundational basis of the OLAC infrastructure that serve to bridge the multiple gaps which now lie in between language resources and users: (1)OLACMS: the OLAC Metadata Set (Qualified DC, Dublin Core), (2) OLAC MHP: refinements to the OAI (Open Archives Initiative, http://www.openarchives.org) protocol, and (3) OLAC Process: a procedure for identifying Best Common Practice Recommendations.

It is crucial to note that the OLAC standards are not standards for the language resources community alone. They are based on two broadly accepted standards in the digital archives community. First, the Dublin Core Metadata Initiative (DCMI) is an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models. There are fifteen Doblin Core Metadata Elements (DCMS) and their qualifiers. OLACMS extends the DC minimally to anwer the needs of the language archives community (Bird, Simons, and Huang 2001).

Second, the Open Archives Initiative (OAI) was launched in October 1999 to provide a common framework across electronic preprint archives, and it has since been broadened to include digital repositories of scholarly materials regardless of their type. The OAI infrastructure requires compliance with two standards: the OAI Shared Metadata Set (i.e. DCMS), and the OAI Metadata Harvesting Protocol (MHP). The OAI MHP allows software services to query a repository using HTTP requests, also an important feature of the recently hyped Semantic Web (http://www.w3.org/2001/sw/). Using the OAI infrastructure, the community's archives can be federated and become a virtual meta-archive collecting all available information. The federeated structure allows end-users to query multiple archives simultaneously. Currently, the Linguistic Data Consortium has harvested the catalogs of over 20 participating archives on behalf of OLAC, and created a search interface which permits queries over all 30,000+ records. A single search typically returns records from multiple archives. The prototype can be accessed via the OLAC website.

[1] We are indebted to Steven Bird and reviewers of the 3rd Workshop on Asian Language Resources and International Standardization for their valuable comments and corrections. Colleagues of the Language Archives project at Academia Sinica provided data and suggestions. Any remaining errors are ours.

In this paper, we trace the version changes of OLACMS, especially in comparison with other (often related) international standards. We will then concentrate on the application of OLACMS to Chinese language resources. In conclusion, we will make some suggestions for OLACMS to account for the characteristics of Chinese language archives.

2 Mapping with other international standards

2.1.Mapping with IMDI

ISLE Meta Data Initiative (IMDI) is a cousin of OLACMS. IMDI proposes a metadata set for natural language processing under the broader International Standards for Language Engineering (ISLE) project. ISLE is co-sponsored by the European Commission of the EU and National Science Foundation of the USA. It aims to develop a set of internationally accepted standards for natural language processing base on the result of the earlier European standard building project (EAGLES, http://www.ilc.pi.cnr.it/EAGLES96/home.html). On one hand, IMDI is an elaboration of OLACMS since it deals specifically with recording sessions. They can also be considered a complimenting each other since they are both devised under the aegis of ISLE.

IMDI Metadata Elements for Session Descriptions, Version 2.5 was completed in June 2001. The elements evolved from the previous EAGLES metadata set described in Wittenburg et al. (2000). Both metadata sets share the aim to improve the accessibility/availability of Language Resources (LR) on the Internet. To achieve this goal, they created a browsable and searchable universe of meta-descriptions similar to those devised by other communities on the Internet.

The focus on Session Description was motivated in Broeder et al. (2000). They observed that individual linguistic resource usually exists in clusters of related resources. For instance, a field video recording of an informant who describes a picture sequence involves several resources. By his definition, an (linguistic) event that called a session is the top element and there results a number of related linguistic resources: Video tape, Photographs, Digitised video file, Digitised photographs, Digitisations of the images used as stimuli, One electronic transcription file, One or more electronic analysis files, Field notes and experiment descriptions (in electronic form). However, since not all linguistic resources come to existence directly through sessions, hence not all linguistic resources can be described by IMDI.[2]

In principle, IMDI metadata can be mapped to OLAC metadata, just as OLAC metadata can be mapped to DC. IMDI Team (August 2001) mapped IMDI Session Descriptions with OLAC 0.3 Version. IMDI Team also use existing description formalisms used by institutions that deal with “published corpora” such as [ELRA] and [LDC]. The set of metadata elements that describe “published corpora” are called “catalogue” metadata elements. The IMDI Team (Gibbon, et al. 2001) launched IMDI Metadata Elements for Catalogue Descriptions, Version 2.1. It also includes Metadata Elements for Lexicon Descriptions.

OLACMS has been updated since December 2001. Hence we did an updated comparison and present the result in this section. Note that since IMDI is an elaboration of OLACMS, we concentrate on the IMDI elements that are not specified in OLACMS and are likely to find wider application. Please note that the section contains our own recommendations inspired by the IMDI/OLAC comparison. We try to add our motivation even for the items that are directly adopted from IMDI. In terms of OLAC scheme, these suggested revision/addition can be assigned the status of attributes (for use by sub-communities), and can be incorporated into the OLACMS later if the community find such addition necessary.

2.1.1. Controlled Vocabulary

Controlled vocabulary defines the basic concepts of the metadata set and any addition to the controlled vocabulary should be motivated by

[2] It is possible to conceive language resources such as lexica and grammars as created through a very large set of (non-planned and non-documented) sessions. But this consideration is beyond the scope of this paper and will not be pursued further here.

the essentiality of the concept.

- **Controlled Vocabulary for *Logical Structure* of linguistic resources**: Language resources come in different forms and various units. A critical piece of information in cataloguing language resource is a description of the composition of the resources. For instance, any English lexicon can be conventionally and naturally viewed as composed of 26 sections defined by shared initial alphabet. Having an element of Logical Structure: alphabetically ordered would give us vital information of how to manipulate the resource. Other vocabularies such as 'sequential chapter', 'dialogue turns', or 'sequential phonemes' would also offer crucial information. In addiiton, if sequential database is indeed the future of language resources, the description of the sequencing logic will play an essential role.
- Add Annotator to [OLAC–Role]. By annotator, we do not refer to the natural person or an automatic program who puts the tags on. By annotator we refer to the institution that implemented the annotation. This information is crucial since this annotator 1) has at least partial IP right on the resource; 2) often set/defines the tagset standard adopted (e.g. Brown, LOB, Penn TreeBank). In other words, annotator can differentiate a new version of resource or even identify totally new resource.
- Add values of archiving Quality to the refine controlled vocabulary of Format.

2.1.2. Elements

One existing elements may need further refining with existing mechanisms.

- Refining the element Project: Many language resources are developed under or partially supported by a project grant. For now, a project can be the value of Creator or Contributor. But just like all other individual creators and contributors, a project needs to be described in fuller details. We need to use attributes to describe the Founder, PI's, Host Institutes, etc. of a project. An umbrella project, such as EAGLES, ISLE, or at a even more complex level, ESPRIT, requires elaboration of contributors and funding timelines themselves.

2.1.3. Updating and Revising the Attributes

- Add sub type to the Space attribute : Coverage of the language resources often calls for geographical information. Hence we need to define the subtypes that include Continent, Country, Administrative division, longitude, latitude, address, etc.
- Add subtype for non-standard Identifier : There are many sets of identifers are defined locally and do not follow URL. In this case, we can add the name of the identifier system (or cataloguer) under schme. For instance, each libary often has its own set of call numbers. Other well-known identifiers arre LCC Catalog No (<Identifier sceeme="LCC"> LCC Catalog No</Identifier>). This could also apply to well-established identifiers such as ISSN and ISBN.
- Although OLAC:Format does not stipulate any refine attributes, however, it is already stipulated in DC:Format. The DC format refine has two control vocabulary entries: Medium specifies the material that the cataloguer uses; while extent records size and duration of the archive. We suggest that OLAC can simply adopt these two refine attributes.

2.2.Mapping with Linguistic Documentation Archives

In addition to IMDI Metadata, Gary Holton (2000) also proposes a system of metadata for the description of language documentation resources following OLACMS. While the system described here should be sufficient for any linguistic resource, it is motivated by the specific ongoing need to describe linguistic documentation materials contained in the Alaska Native Language Center (ANLC) Archive. Particular attention is paid to description of first-hand documentation materials such as field notes, grammatical notes, and phonological descriptions, many of which currently exist only in written form. Existing resources are in the process of being digitized, and new digital resources continue to be acquired. The ANLC collection presently contains more than ten thousand items. While much of the material consists of original manuscripts of archival quality, the collection also includes published materials and materials existing in other archival collections, duplicated in whole or in part. The ANLC Archive thus combines both archival and library functions.

The unique need described in Holton (2000) is that he wants the Metadata set to be applied simultaneously to non-digital archives, such as manuscript, reel-to-reel cassettes, CD recordings etc. This can be done by adopting the DC:Format refine attribute of Medium. In order to descibe the archives more felicitously, we also need to add speaker, interviewer Holder, and Guardian to the value of controlled vocabulary of refine of Creator and Contributor. However, there does not seem to be any straightforward way to transfer Target Dialect.

2.3.Summary

Based on the two comparison of different metadata sets, we found that the DC qualifier can be applied effectively to solve the bridging and conversion problems between different DC-based extension metadata sets. This should be exactly what OLACMS design has in mind. The attributes that were not stipulated in OLACMS 0.4, if found in DC and motivated by actual need to describe language resources, can be easily adopted. One way to ensure the versatility is to keep all DC attribute in OLACMS, even though some of the attributes may be dormant and not actively applied now. Another issue worth noting is that any cataloguer may add sub-elements to achive more comprehensive description. However, such addition should, follow the extension and adaptability of the DC.

3 Use Controlled Vocabulary for Temporal and Geographic Location

Constable, and Simons (2000) listed all the causes for language changes, which basically involve the change in the temporal-spatial location of the poeple. Since China used a different calendar system until late in early 20th century, all inherent temporal description of inherited Chinese archives do not conform to the current DC standard. In order to identify Western and Chinese chronology, we may stipulate that the primary types of the scheme element to be Western (W_Calendar) or Chinese (C_Calendar). We may also add other chronological methods, such as lunar or solar calendar. The sub_type of Chinese calendar will then include time, dynasty name, state name, emperor's reign, and the reign name of the emperor. Take the Academia Sinica Ancient Chinese Corpus for example. Its coverage is Early Mandarin Chinese, and will marked as such in the metadata: <Coverage scheme="C_calendar/phase">EarlyMandarin </Coverage>. The users will be able to refer to a historical linguistic calendar and find that the time equals to the dynasties of Yuan, Ming, and Ching. And will be able to convert the time to western calendar using the conversion table of [Sinica Calendar].

When Coverage has a spatial refinement, a location can have different names because of the unit used in cataloguing, as well as because of temporal or regional and linguistic variaions. Hence, the spatial value of Coverage must be defined by a scheme. A scheme must stipulate temporal reference as unit of catalogue. For instance, the Sinica Corpus covers the language of the Republica of China in Taiwan. Its metadata will have the following value <Coverage refine= "spatial" scheme= "ROC/Taiwan">. As mentioned above [Sinica Calendar] offers conversion table for the past 2000 years between Chinese and Western calendars. As for the units for cataloguing of spatial location, OLAC 0.4 Version adopts [TGN](Getty Thesaurus of Geographical Terms). And many other digital archives follow Alexandria Digital Library Feature Type Thesaurus [ADL]. The ADL type thesaurus have been adopted by the digital archives project in Taiwan and translated into Chinese by Academia Sinica Metadata Architecture and Application Team [Sinica MAAT].

4 Applying OLACMS to Language Archives in Taiwan

Each text in Academia Sinica Balanced Corpus of Modern Chinese (Sinica Corpus) is marked up with five textual parameters: Mode, Genre, Style, Topic and Medium. These are important textual information that needs to be catalogued in metadata. The following shows how we transfer and represent these (legacy) textual information to OLACMS:

4.1.Mode and Genre

Table 1 The relation between Mode and Genre of Sinica Corpus(Ckip Technology Report 93-05)

Mode	Genre
Written	Reportages Commentary Advertisement Letters Announcement Fiction Prose Biography & Diary Poem Manual
written-to-be-spoken	Script Speech
Spoken	Conversation
spoken-to-be-written	analects Speech Meeting Minute

We add a refine attribute under Type. Mode is added in the controlled vocabulary as Primary type, and Genre is added as sub type. For instance, a recorded and transcribed speech is catalogued as <Type code="Sound" refine="spoken-to-be-written/Speech"/>.

4.2.Style

There are four styles that are differentiated in Sinica Corpus: Narrative, Argumentative, Expository, and Descriptive. We add a new refine attirbute under Descriptio, with Style as a controlled vocabulary. For instance, a diary will be catalogued as: <Description refine="Style"> Narration </Description>.

4.3.Medium

Sinica Corpus specifies the media of the language reources as: Newspaper, General Magazine, Academic Journal, Textbook, Reference Book, Thesis, General Book, Audio/Visual Medium, Conversation/Interview. We may also add other audio-video media such as CD,V8…etc. As mentioned above, this can be easily described with DC: Format refine attribute of Medium.

4.4.Topic

The Topic parameter of Sinica Corpus has the same content as the element Subject. This can simply be transferred through a table.

Table 2 Topic of Sinica Corpus(Ckip Technology Report 93-05)

Primary	Sub
Philosophy	Thoughts | Psychology | Religion |
Natural Science	Mathematics | Astronomy | Physics | Chemical | Mineral | Creature | Agriculture | Archeology | Geography | Environmental Protection | Earch Science | Engineering |
Social Sciences	Economy | Finance | Business & Management | Marketing | Politics | Political Party | Political Activities | National Policy | International Relations | Domestic Affairs | Military |Judicature | Education | Transportation | Culture | History | Race | Language | MassMedia | Public Welfare | Welfare | Personnel Matters | Statistical Survey | Crime | Calamity | Sociological Facts |
Arts	Music | Dance | Sculp | Painting | Photography | Drama | Artistry | Historical Relics | Architecture | General Arts |
General /Leisure	Travels | Sport | Foods | Medical Treatment | Hygine | Clothes | Movie and popular arts | People | Information | Cunsume | Family |
Literature	Literary Theory | Criticism | Other literary work | Indigenous Literature | Childern's Literature | Martial Arts Literature | Romance |

An example for the adoptation follows: for a Sinica Corpus text with a Topic of Arts and a sub-topic of Music, it will be catalgued as follows: <Subject>Arts/Music</Subject>.

4.5.Additional Controlled Vocabulary

- Proofreader: Since both manually and automatically digitized materials must be proofread to ensure quality, we suggest that [OLAC-Role] be enriched by a new value: Proofreader. For inherited texts with no IP restrictions, this may be the critical information piece of information to identify who is the rightful owner/creator of the electronic version.
- There are many Medium values old (procelain, rubbing, bamboo engraving, silk scroll, etc.) and new (DVD, MO, ZIP...etc). Hence the controlled vocabulary of attributes such as Medium and SourceCode often has quick and drastic changes. In order to maintain versatility and comprehensive coverage, this set of controlled vocabulary must be open and allows each participant to register, subject to the approval by OLAC.

5 Language Identification

Constable and Simons (2000) noted that a computer, unlike human beings, cannot automatically identify the language of a text that it is reading yet. Hence metadata must play a central role in identifying the language that each resource uses. For instance, Malay and English uses the same 26 letters. And Archaic Chinese 2000 years ago and Modern Mandarin can be expressed by pretty much the same set of Chinese characters. These are all different languages and need to be identified before a language resource can be used. SIL (Summer Institute of Linguistics, in its white-paper identified five major issues for language identification: Change, Categorization, Inadequate definition, Scale, and Documentation. SIL has produced an online searchable database: Ethnologue that provides a comprehensive system of language identification covering more than 6,800 languages. This is adopted by OLAC as an obvious improvement over the very small set covered in DC.

Bird et al. (2001), however, pointed out some problems of coverage if the Enthlogue system is adapted without further means of enrichment. The three broad categories of problem are: over-splitting, over-chunking and omission. Over-splitting occurs when a language variety is treated as a distinct language. For example, Nataoran is given its own language code (AIS) even though the scholars at Academia Sinica consider it to be a dialect of Amis (ALV). Over-chunking occurs when two distinct languages are treated as dialects of a single language (there does not appear to be an example of this in the Ethnologue's treatment of Formosan languages). Omission occurs when a language is not listed. For example, two extinct languages, Luilang and Quaquat, are not listed in the Ethnologue. Another kind of omission problem occurs when the language is actually listed, but the name by which the archivist knows it is not listed, whether as a primary name or an alternate name. In such a case the archivist cannot make the match to assign the proper code. For instance, the language listed as Taroko (TRV) in the Ethnologue is known as Seediq by Academia Sinica; several of the alternate names listed by the Ethnologue are similar, but none matches exactly.

The above problems may prove to be a stumbling block for archives that attempt to integrate linguistic resources with GIS (Geographic Information System), such as the [Formosan Language Archive] at Academia Sinica. A GIS-based language atlas will most likely be very concerned with fine-grained changes and variations among languages and dialects within a geographic area. In other words, these kind of archives may either discover yet unrecorded language or sub-language differentiations or need even finer classification in Ethnologue or any language identification system. Hence the solution proposed in Bird et al. (2001) of allowing local language classification systems to register must be implemented under OLAC.

6 Conclusion

We looked at a couple of OLAC derived metadatasets, as well as applied OLAC version 0.4. to three different language archives in Taiwan. We proposed some suggestions for enriching of OLACMS based on the study. There are two general directions to bear in mind. First, as the number and complexity of language resources becomes higher and higher, the need to have a uniform standard or to easy access to the owner of each resource becomes even greater. Therefore, we envision that the element of Creator, Contributor etc. needs further elaboration, which may include practical information such as email addresses etc. Second, as the language archives get richer, the need to note language variation grows even bigger. Simple language identification of allotting a resource a unique language code is not enough. There will be great need to infer linguistic relations from these codes. Since it is impossible to build a complete reportiore of resources for all languages, it is very often that a resources from the closest related language must be borrowed. The representation of linguistic relations will be the next challenge of language identification.

References

I. Bibliography

Bird, S. 2000. ISLE: International Standards in Language Engineering Spoken Language Group, http://www.ldc.upenn.edu/sb/isle.html

Bird, S., G. Simons, and C.-R. Huang 2001. The Open Language Archives Community and Asian Language Resources, 6th Natural Language Processing Pacific Rim Symposium Post-Conference Workshop, Tokyo, Japan.

Broeder, D., P. Suihkonen, and P. Wittenburg. 2000. Developing a Standard for Meta-Descriptions of Multimedia Language Resources, Web-Based Language Documentation and Description workshop, Philadelphia, USA.

CKIP. 1993. An Introduction to Sinica Corpus. CKIP Technology Report 93-05. IIS, Academia Sinica.

Constable, P. and G. Simons. 2000. Language identification and IT: Addressing problems of linguistic diversity on a global scale, SIL Electronic Working Papers 2000-001.http://www.sil.org/silewp/2000/001/

EAGLES/ISLE. ISLE Meta Data Initiative, http://www.mpi.nl/world/ISLE/

Gibbon, D., Peters, W., and Wittenburg, P., 2001. Metadata Elements for Lexicon Descriptions, Version 1.0, MPI Nijmegen, http://www.mpi.nl/ISLE/documents/draft/ISLE_Lexicon_1.0.pdf

Holton, G. 2000. Metadata for Linguistic Documentation Archives, Web-Based Language Documentation and Description workshop, Philadelphia, USA.

IMDI Team. 2001. IMDI Metadata Elements for Session Descriptions, Version 2.5, MPI Nijmegen, http://www.mpi.nl/ISLE/documents/draft/ISLE_MetaData_2.5.pdf.

IMDI Team. 2001. Mapping IMDI Session Descriptions with OLAC, Version 1.04, MPI Nijmegen. http://www.mpi.nl/ISLE/documents/draft/IMDI%20to%20OLAC%20Mapping%201.04.pdf

IMDI Team. 2001. IMDI Metadata Elements for Catalogue Descriptions, Version 2.1, MPI Nijmegen, http://www.mpi.nl/ISLE/documents/draft/IMDI_Catalogue_2.1.pdf

Palmer, M. 2000. ISLE: International Standards for Language Engineering: A European/US joint project, http://www.cis.upenn.edu/~mpalmer/isle.kickoff.ppt

Wittenburg, P., D. Broeder, and B. Sloman. 2000. EAGLES/ISLE: A Proposal for a Meta Description Standard for Language Resources, White Paper. LREC 2000 Workshop, Athens.

II. Websites

[OLAC] Open Language Archives Community, http://www.language-archives.org

[OLACMS] OLAC Metadata Set, http://www.language-archives.org/OLAC/olacms-20011022.html

[DCMI] Dublin Core Metadata Initiative, http://dublincore.org/

[DCMS] Dublin Core Element Set, Version 1.1 - Reference Description, http://dublincore.org/documents/dces/.

[DC-Q] Dublin Core Qualifiers. http://dublincore.org/documents/2000/07/11/dcmes-qualifiers/

[ISLE] International Standards for Language Engineering, http://www.ilc.pi.cnr.it/EAGLES96/isle/ISLE_Home_Page.htm

[ELRA] European Language Resources Association, http://www.icp.grenet.fr/ELRA/

[LDC] Linguistic Data Consortium, http://morph.ldc.upenn.edu/

[Sinica Calendar] Western Calendar and Chinese Calendar Conversion Table of Academia Sinica Computing Centre.
http://www.sinica.edu.tw/~tdbproj/sinocal/luso.html.

[Academia Sinica Ancient Chinese Corpus] Academia Sinica Tagged Corpus of Early Mandarin Chinese, http://www.sinica.edu.tw/Early_Mandarin/

[TGN] Getty Thesaurus of Geographical Terms, http://www.getty.edu/research/tools/vocabulary/tgn/index.html

[ADL] Alexandria Digital Library Feature Type, http://alexandria.sdc.ucsb.edu/gazetteer/gaz_content_standard.html

[Sinica MAAT] Metadata Architecture and Application Team, http://www.sinica.edu.tw/~metadata/standard/place/ADL-element.htm

[Sinica Corpus] Academia Sinica Balanced Corpus of ModernChinese, http://www.sinica.edu.tw/SinicaCorpus/

[Ethnologue] http://www.ethnologue.com

[Formosan Language Archive] Academia Sinica Formosan Language Archive, http://www.ling.sinica.edu.tw/Formosan/