Language, Information and Computation(PACLIC 11), 1996, 129-136

A Logical Structure for the Construction of Machine Readable Dictionaries

Byung Jin Choi**, Jae Sung Lee*, Woon Jae Lee* and Key Sun Choi*

*Department of Computer Science, KAIST
**Center for Artificial Intelligence Research, KAIST

Abstract

During the last 10 years, there have been many efforts in some areas of Natural Language Processing to encode the normal text or documents into machine readable form. If we encode written data using a canonical form which can be recognized by a computer, we can extract needed information and process and utilize it for another purposes. From this point of view, we present an account of the encoding of a printed dictionary. The construction of a lexicon is very time-consuming and expensive work and the application of the lexicon is restricted. In this paper, we describe a logical structure for Korean printed dictionaries as a general lexical representation based on SDML, which can be transformed into another representation for different application requirements.

1. Introduction

With the rapid development of the computer technology, there are many efforts in some area of Natural Language Processing to convert normal text or documents into machine readable form. If we convert written data into a canonical form which can be recognized by computer, we can extract needed information and process and utilize it for another purpose.

In order to do this, we need to encode documents in a markup system. For the last 10 years, there have been many approaches to develop a mark-up system in many NLP projects. A Mark-up system is a set of instruction, with which a document or text can be stored in a formalized form. In other words, we can build complex data structures with a limited number of tags of this mark-up system.

In the development of mark-up systems, standardization problems have been an issue. At the beginning of the 1980s, some researchers tried to develop encoding system and at last SGML (Standardized Generalized Markup Language) and TEI (Text Encoding Initiative) are emerged. With the help of SGML or TEI many documents or texts are structured as DTD (Document Type Definition) of SGML or TEI (Bryan 1988, Ide et al. 1995, Sperberg-McQueen et al. 1994).

In this paper, we present an account of the encoding of printed dictionary. The construction of a lexicon for a limited application domain is both time-consuming and expensive. When we define and implement a lexicon only for some natural language applications, we cannot re-use this lexicon for another applications. To avoid this problem, we can encode the lexical information in a standard format. If we have the lexical information in a standard format, we can transform into the machine readable form or vice versa.

If a lexical information in one system is compatible with another application, we can reuse the existing standard lexicon and save our efforts to build a lexicon. Such a

reusability of existing lexical information is the main focus of this paper. Calzolari has distinguished two kinds of reusability (Kugler 1995). One is transforming already existing lexical resources into a different format, typically transforming printed dictionaries into a machine readable or machine tractable form (Amsler 1988, Alshawi 1989). The other is exploiting already existing lexical resources for different theories and -typically - for different applications (Briscoe et al. 1993, Hajicová & Rosen 1994). Such an idea is realised in this work.

In order to maintain the reusablity of lexical information, we have developed the Standard Dictionary Markup Language (SDML). With SDML we can define the logical structure of lexicon and store lexical information independent of specific applications, data structures and theories.

With the help of decoding program we can import the lexical information from printed dictionaries, text corpora, and some lexical databases into the standard dictionary. This information from the standard dictionary is then available for various natural language applications. In other countries there have been many such projects during the last 10 years. But we have no representative work on machine readable dictionary or electronic dictionary yet, except for the work of Kang (1996) about encoding Korean dictionaries. Kang adopted the TEI scheme for the encoding of Korean dictionary entries. We have tried an SDML-based encoding, which is adapted to dictionary entries. Because of its simplicity, it is easy to create a lexicon structure that can be applied across various possible dictionaries.

2. Standard Dictionary Markup Language (SDML)

SDML is used to define the various dictionary formats. Once the dictionaries and text formats are defined using SDML, they can be easily for the standardization and interchange of the information.

In a SDML definition we have three parts, header, front (pre-definition) and entry group. <Header> is used for dictionary information such as dictionary name, dictionary version etc. In <Front>, the attributes and values which are needed for internal structure are defined. The <entry group> comprises the lexical information of dictionaries. Entries are repeated, in which the headword (lexical entry) is described.

```
<sd>
    <sdHeader> [header information] </sdHeader>
    <group>
        <entry>
            <wname> [headword] </wname>
            <body> [standard dictionary elements] </body>
        </entry>
        <entry>
            <wname> [headword] </wname>
            <body> [standard dictionary elements] </body>
        </entry>
        [Repetition of  entry]
    </group>
</sd>
```

The elements which occur in <body> are regarded as standard dictionary format.

It can have only one field as a default value, or also have complex tree structures. A user can define the logical structure of lexicon in <body> and this definition is overriding and is overwritten the default definition. For the definition in SDML, it seems to be similar to SGML, we have some restrictions in using the rules.

1) In the content model, we use the occurrence indicators, '*', '+' and '?', and as connectors ',' and '|'. The connector '&' is redundant for defining the dictionary format.
2) For reasons of compatibility, predefined tags should be used. (in the upgrade version, the more standard tags will be defined.)

The default definition for <body> is following:

```
<!ELEMENT body      - -       (#PCDATA)* >
```

In order to build a new structure for <body>, we only need to write the definition for needed elements. As an example we can change the structure of <body> with the overwriting definition of <pos> and <def>.

```
<!DOCTYPE     sd      SYSTEM "sdml.dtd"   [
<!ELEMENT     body    - o       (pos, def) >
<!ELEMENT     pos     - o       (#PCDATA) >
<!ELEMENT     def     - o       (#PCDATA) >
]>
<sd>
        <!-- the instance of the above dictionary format -->
                :
</sd>
```

3. Representation of lexical entries in SDML

The first thing that we take into account is how to encode lexical information in order to extract relevant information efficiently. Each dictionary has a different structure (macro structure) and each dictionary entry varies extremely in its structure (micro structure) both within and among dictionaries. (Ide et al. 1995)

Therefore, it is very difficult to find a general structural description which can be applied across all possible dictionaries.

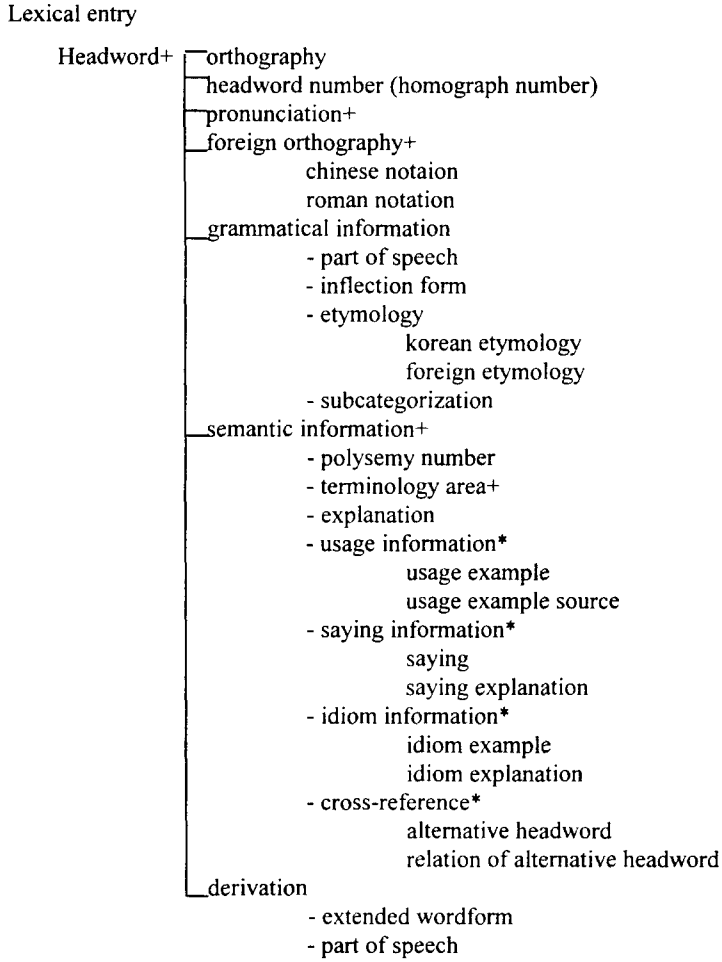
In dictionary compiling, the macro structure is associated with the order of lexical entries. How to find a word in dictionary (e.g. according to alphabet or concept) is the problem of constructing macro structure. How the information of headword is organized is the problem of micro structure design. In Korean dictionaries, the macro structure may vary a lot, therefore only the micro structure of lexical entries is described.

There are many kinds of Korean dictionaries (Samsung, Minjung, Kumsung, Donga etc.). We have compared these dictionaries, analysed each lexical entry and specified the atomic elements which appear in a dictionary entry. Many of them are

different in their internal structure but there are also relevant common elements. The common lexical information is following:

> headword (orthography), homograph number, pronunciation, part of speech, inflection class, etymology, terminology classification, usage information, definition, cross-reference, related headword (antonym, synonym, hyponym etc.), example, source, saying, idiom, derivation

This information is hierarchically organized. As a top level there is a information of word form which comprises pronunciation, orthography, and foreign orthography. At the same level as the lexical entries, we have grammatical information, semantic information, and derivational morphological information. All these are subclassified and contain specified information in their respective subhierarchies. This internal structure can be displayed as a tree:



This graphic representation is encoded in SDML in the following way:

DTD definition	Explanation
<!DOCTYPE WDDIC [	Dictionary
<!ELEMENT WDDIC - - (le)+ >	
<!ELEMENT le - - (hw , hwn , (fn | pr)+ , (gi , si+)+ ,wfext*) >	lexical entry
<!ELEMENT hw - - (#PCDATA) >	headword
<!ELEMENT hwn - O (#PCDATA) >	homograph number
<!ELEMENT fn - - (ch | rom) >	foreign notation
<!ELEMENT ch - O (#PCDATA) >	chinese notation
<!ELEMENT rom - O (#PCDATA) >	roman notation
<!ELEMENT pr - O (#PCDATA) >	pronunciation
<!ELEMENT gi - - (pos , fl? , etym?, subc?) >	grammatical information
<!ELEMENT pos - O (#PCDATA) >	part of speech
<!ELEMENT fl - O (#PCDATA) >	inflection form
<!ELEMENT etym - - (ketm | fetm) >	etymology
<!ELEMENT ketm - O (#PCDATA) >	korean etymology
<!ELEMENT fetm - O (#PCDATA) >	foreign etymology
<!ELEMENT subc - O(#PCDATA) >	subcategorization
<!ELEMENT si - - (num , term+ , expl , usage*, saying* , idiom* , althw*) >	semantic information
<!ELEMENT num - O (#PCDATA) >	polysemy number
<!ELEMENT term - O (#PCDATA) >	terminology area
<!ELEMENT expl - O (#PCDATA) >	explanation
<!ELEMENT usage - - (us_ex , us_source?) >	usage information
<!ELEMENT us_ex - O (#PCDATA) >	usage example
<!ELEMENT us_source - O (#PCDATA) >	usage example source
<!ELEMENT saying - - (sng , sng_ex?) >	saying information
<!ELEMENT sng - O (#PCDATA) >	saying example
<!ELEMENT sng_ex - O (#PCDATA) >	saying explanation
<!ELEMENT idiom - - (idm , idm_ex) >	idiom information
<!ELEMENT idm - O (#PCDATA) >	idiom example
<!ELEMENT idm_ex - O (#PCDATA) >	idiom explanation
<!ELEMENT alt_hw - - (hw , alt_hw_rel) >	alternative headword (cross-reference)
<!ELEMENT alt_hw_rel - O (#PCDATA) >	relation of alternative headword
<!ELEMENT wf_ext - - (hw , pos)+ >	extended wordform: derivation
] >	

4. Conclusion

In this paper, we tried to describe the structure of a Korean standard dictionary, which is first encoded in a standard format and can then be reused for various applications. On the basis of SDML, we could define a DTD for the logical structure of standard dictionary. If a standard dictionary can be organized this way, we can extract many useful linguistic information and utilize many kind of dictionaries such as synonym dictionary, antonym dictionary, idiom dictionary, example dictionary, and dictionaries for morphological analysis etc. Furthermore we can process these

dictionaries and use them for NLP systems, e.g. machine translation, automatic language analysis, and information retrieval etc. In this way we can save time and effort for building other lexicons for specific purpose. We have implemented this standard dictionary in TDMS (Text and Database Management System). In the future, we will concentrate on the implementation of structured information retrieval for extracting arbitrary information from the lexical database. Furthermore, the integration of non-monotonic logic is desirable for the lexical organization.

Acknowledgements

The research described here was undertaken as a part of the project 'Korea Information Base System' by the support of Ministry of Science & Technology and Ministry of Culture & Sports in Korea and the project 'Multimedia Hangeul Engineering' supported by Samsung Co..

References

Alshawi, Hiyan. 1989. Processing Dictionary Definitions with Phrasal Pattern Hierarchies. In B. Boguraev and E. Briscoe, eds., *Computational Lexicography for Natural Language Processing*, 153-170. London and New York: Longman.

Amsler, Robert A., and W. Tompa. 1988. An SGML-based Standard for English Monolingual Dictionaries. In *Proceedings of the 4th Annual Conference of the UW Centre for the New Oxford English Dictionary*, 61-80. Ontario: Waterloo.

Boguraev, Bran. 1991. Special Issue on Computational Lexicons. *International Journal of Computational Lexicography 4.*

Boguraev, Bran. 1994. Machine-readable dictionaries and computational linguistics research. In A. Zampolli, N. Calzolari, and M. Palmer, eds., *Linguistica Computationale, Vol. IX.X Current Issues in Computational Linguistics: in Honor of Don Walker*, 119-154. Dordrecht: Kluwer Academic Pub.

Boguraev, Bran., and T. Briscoe (eds.). 1989. *Computational Lexicography for Natural Language Processing*. London and New York: Longman.

Briscoe, Ted, V. de Paiva, and A. Copestake (eds.). 1993. *Inheritance, Defaults, and the Lexicon.* Cambridge: Cambridge University Press.

Bryan, Martin. 1988. *SGML: An Author's Guide to the Standard Generalized Markup Language*. Addison-Wesley

Copestake, Ann. 1990. An Approach to Building the hierarchical Element of a Lexical Knowledge Base from a Machine Readable Dictionary. In *Proceedings of the First International Workshop on Inheritance in Natural Language Processing*, 19-29. Tilburg.

EDR. 1993. *EDR Electronic Dictionary Technical Guide.* Japan Electronic Dictionary Research Institute.

Hajicová, Eva, and R. Alexander. 1994. Machine Readable Dictionary as a Source of

Grammatical Information. In A. Zampolli, N. Calzolari, and M. Palmer, eds., *Linguistica Computationale, Vol. IX.X Current Issues in Computational Linguistics: in Honor of Don Walker*, 191-199. Dordrecht: Kluwer Academic Publishers.

Ide, Nancy, and J. Véronis. 1995. Encoding Dictionaries. In: Nancy Ide & Jean Veronis , eds., *Text Encoding Initiative*, 167-180. Dordrecht: Kluwer Academic Pub.

Kang, Beommo. 1996. Using the TEI Scheme in Compiling a Korean Dictionary. In *ALLC-ACH'96* (6), 162-165. Bergen.

Kugler, Marianne, K. Ahmad, and G. Thurmair (eds.). 1995. *Translator's Workbench: Tools and Terminology for Translation and Text Processing*. Berlin: Springer-Verlag.

Sperberg-McQueen, C.M. and L. Burnard (eds.). 1994. *Guidelines for Electronic Text Encoding and Interchange (TEI P3)*. Chicago and Oxford.

Sperberg-McQueen, C.M. and N. Ide. 1995. The TEI: History, Goals, and2 Future. In Nancy Ide & Jean Veronis , eds., *Text Encoding Initiative*, 5-16. Dordrecht: Kluwer Academic Pub.

Zampolli, A., Calzolari, N., and M. Palmer 1994. *Linguistica Computationale, Vol. IX.X Current Issues in Computational Linguistics: in Honor of Don Walker*. Dordrecht: Kluwer Academic Publishers.

Appendix: Example of a Korean lexical entry and its representation in SDML

가난 【명】【하|형】 (←간난) 살림살이가 넉넉하지 못함. 빈곤. ¶ ~한 집 / ~에 쪼들리다. / ~에서 벗어나다. **[가난 구제는 나라도 못한다]**. 가난한 사람을 구제하기는 끝이 없어 개인은 물론 나라의 힘으로도 어렵다.
[가난이 원수] 가난하기 때문에 고통을 받게되니 가난이 원수같이 느껴진다.
[가난한 집 제사 돌아오듯] 치르기 힘든 일이 자주 닥침을 비유하는 말.
가난(이) 들다. 【구】 ① 가난하게 되다. ② 쓸만한 것이 드물어 구하기 어렵다.

```
<entry>
<wname> 가난 </wname>
<hwn> 1
<pr>   가난 </pr>
<gi>
  <pos> 명사 </pos>
</gi>
<si>
   <num> 1
   <expl> 살림살이가 넉넉하지 못함. 빈곤 </expl>
   <usage>
     <us_ex>   가난한 집
     <us_ex>   가난에 쪼들리다
     <us_ex>   가난에서 벗어나다
   </usage>
   <saying>
     <sng>        가난 구제는 나라도 못한다.
     <sng_expl>  가난한 사람을 구제하기는 끝이 없어 개인은 물론 나라의 힘으로도
                  어렵다.
   </saying>
   <saying>
     <sng> 가난한 집 제사 돌아오듯
     <sng_expl> 치르기 힘든 일이 자주 닥침을 비유하는 말
   </saying>
   <idiom>
     <idm> 가난(이) 들다
     <idm_expl> ① 가난하게 되다. ② 쓸만한 것이 드물어 구하기 어렵다.
   </idiom>
   <alt_hw>
     <hw> 간난
     <alt_hw_rel> 유의어
   </alt_hw>
</si>
<wfext>
   <hw> 가난하다 </hw>
   <pos> 형용사  </pos>
</wfext>
</entry>
```