A Tree-structure Solution for the Development of ChineseNet

Liu Yang, Yu Jiangsheng, Yu Shiwen

Abstract In this paper, we would like to put forth the notion of tree-structure in the development of a WordNet-compatible concept dictionary. After getting the full- hyponymy information in WordNet successfully, we have further implemented a visual tree-structure control which enables the lexicographers to operate interactively on the view of the hyponymy tree, with correspondingly automatic modifications of the database in the background. The expressing of semantics in the development thus adopts a much more intuitionistic and efficient way. ICL (the Institute of Computational Linguistics) now has benefited a lot by employing this new solution for the development of CCD (the Chinese Concept Dictionary), our ChineseNet, here in Peking University.

Keywords Tree, Algorithm, Chinese, WordNet, CCD

1 Introduction

Nowadays NLP in Chinese is more focused on the processing of content information, such as Information Retrieval, Automatic Abstraction, Literature Classification and others, which needs a WordNet-compatible dictionary of Chinese concepts as the knowledge-base. On the other hand, the Chinese Concept Dictionary (CCD) is totally necessary for Word Sense Disambiguation (WSD) in the field of NLU and MT.

The Institute of Computational Linguistics (ICL), Peking University, with this point of view, has launched its ChineseNet project. The expectant CCD might be described as follows: it should carry the main relations already defined in WordNet with more or less updates to reflect the practice of Contemporary Chinese, and, it should be a bilingual one with the parallel Chinese-English concept pairs to be simultaneously presented within. Such a WordNet-compatible dictionary of Chinese concepts can largely meet our need of applications.

Thus comes the radical issue on how to build the dictionary, or, in other words, on what is the proper solution for it.

Answers may lie in the inherent structure of WordNet. As the relations that link synsets in the dictionary actually interweave into a huge lexical semantic net, say tens of thousands nodes, so what really counts in the development of such a WordNet-compatible dictionary is how to set up the relations properly and how to maintain the semantic consistencies in case of frequent occurrences of modifications. After analyses, we have come to believe that the difficulties of the development of the dictionary mainly result from these.

Roughly there can exist three kinds of potential choices for the solution, a linear-structured one, a tree-structured one or a net-structured one. Let’s begin with the two ends. Naturally, a linear structure can hardly do for the difficulties of carrying tremendous pieces of net-structured information during the period of development. To develop the dictionary directly on a large-scale, visual, net-structure control instead of merely the linear database may shed some light on this issue. However, due to the same problem of connatural complexity, of time and of storage, such a net-structure control has not appeared, and, perhaps, may never appear.

In this paper, we would like to put forth the notion of tree structure in the development of the dictionary.

Later on, we intend to give an illumination of the solution with emphasis on the basic ideas, though some of the algorithms may be inevitably touched on. The actual cases of the development practice in Peking University and the prospects of CCD will also be involved.

2 Getting the Full-hyponymy Information in WordNet

Obviously, the relation of hyponymy in WordNet is the most important relation among others.

Since WordNet means to describe a kind of syntagmatic relations in lexicon, hyponymy, we may say, serves as the main frame to grasp the other relations such as antonymy, meronymy, troponymy, etc. During the period of development, only when synsets (to act as the basic units which also highly rely on the ontology of full-hyponymy) and hyponymy (to act as the basic relation) are well defined and realized, can the other relations be appropriately added into the lexicon. Likewise, during the period of utilization, the full-hyponymy information is totally valuable for the higher applications and the browser users.

However, to extract the full hyponyms for a certain synset is by no means easy. As we have examined, the number of hyponyms for a synset ranges from 0 to 499 with a maximal hyponymy depth of 15 levels. This shows the shape of the potential full-hyponymy tree is quite unbalanced. Because of this, the ordinary searching algorithm can hardly do with the complexity of time and storage. If one inputs the word entity as an entry in WordNet 1.6 and try to search its full hyponyms, he will get nothing but a note of "Search too large. Narrow search and try again." provided that he does not narrow the searching by terminating it beforehand. Sure enough, if the entry is not entity but another word, say cat, the searching will probably do. The cases actually depend on the location of the entry word in the potential full-hyponymy tree in WordNet. The higher the level, the less possibility of success the searching will have.

Before we can go on, we need to introduce the item of position which plays a pivotal role in the tree-structure solution. A position means the location of a certain node in the tree and it serves to organize the tree. For example, a position by the value "005001002" is to be representing such a location of a node in a tree: at the 1st level, its ancestor being the 5th; at the 2nd level, its ancestor being the 1st; and at the 3rd level, its ancestor viz. itself now being the 2nd. In fact, such an encoding does take an appearance of a linear string while expressing the full information of a tree-structure. This special kind of encoding makes all the tree-structure algorithms feasible.

Now let us demonstrate the searching algorithm for getting the full-hyponymy information in WordNet. By and large, it involves a series of the two-way scanning process and the gathering process, with each round of the process series intending to get the information of nodes on one same level in the tree.

Suppose, in the I-th round of the process series, we have got the synsets L_{i1}, L_{i2}, … , L_{in} with the positions “X001”, “X002”, … , “X00n” respectively. This implies that on the I-th level in the tree, there are n nodes with the locations “X001”, “X002”, … , “X00n” respectively in the tree. Then we want to have the (I+1)-th round of the process series to get the information of the nodes on the (I+1)-th level in the tree. We have these synsets ordered by their offsets before we could do the two-way scanning process. In the scanning process, two pointers are set, one for the array of the above sorted synsets, the other for the corresponding DAT file to be compared with. During comparing, new positions on the (I+1)-th level are continuously generated according to the definition of position encoding. It is easy to prove that such a task can be done in an O(N) time and an $O(n+n^{'})$ storage, assuming N representing the number of records in the DAT file and $n^{'}$ representing the number of synsets on the (I+1)-th level in the tree. After the two-way scanning, $n^{'}$ synsets on the (I+1)-th level in the tree together with their respective positions can be got. In the gathering process, those synsets on the (I+1)-th level satisfying leaf-node condition is gathered while those sieved out are to be put into the next round of the process series.

These process series are to be carried on repetitively till no new positions are generated on one round, which just means that the full-hyponymy information in WordNet has already been completely achieved.

Also, it can be inferred that the number of the rounds equates to the depth of the full-hyponymy tree for a specific entry word, say 15 for entity, or 11 for food. Now we have got the full-hyponymy information in WordNet, and, if one wants to view the tree, these pieces of information are ready for the tree-structured control through an operation of creating tree.

By this special algorithm, the complexity of searching is greatly reduced. In our lab, a tapping of the top entry word entity on an ordinary PC means 100 or so seconds of waiting time with all the 45,148 synonyms generated. As for the ordinary entry word, say food with a total amount of 2,308 hyponyms, the algorithm is simply a real-time one.

3 The Tree-structure and the Operations on It

Following that, we have schemed a set of algorithms based on the existent Treeview Control in the Microsoft Visual Studio 6.0 and eventually implemented a new data-sensitive tree-structure control with 9 visualized operations on it.

Apropos of the design of the algorithms for the operations, it is crucial that two sorts of critical consistencies should be especially maintained. One is that of the structural information of the foreground tree and the other is that of the semantic information of the background database. As these algorithms are too intricate to be presented here, in an introductive paper, we would just list names of the 9 visualized operations below.

0. to create a tree from a file;
1. to new a brother node;
2. to new a child node;
3. to delete the current node (for one);
4. to delete the current node (for all);
5. to cut the current node;
6. to copy the current node;
7. to paste as brother nodes;
8. to paste as children nodes.

Among these operations, apart from that the No. 0 is to create a new tree from the external storage, the rest are all to edit the tree, with respectively the No. 1, 2 for addition, the No. 3, 4 for deletion, and the No. 5, 6, 7, 8 for batch movement. These operations have been carefully chosen to make them concise enough, capable enough and semantically meaningful enough.

It is easy to prove that any facultative-shaped tree can be attained by iterative practice of these operations.

4 The Development of ChineseNet in Peking University

By now, we have got the full-hyponymy information in WordNet successfully through the special searching algorithm, and, it is just these pieces of information that will serve as the data bases for our future use. Also, we have got a visual, data-sensitive, tree-structure control with the above well-defined operations on it. Then we will go on to organize the full-hyponymy information, plus some other information relevant to the synsets, into a hyponymy tree.

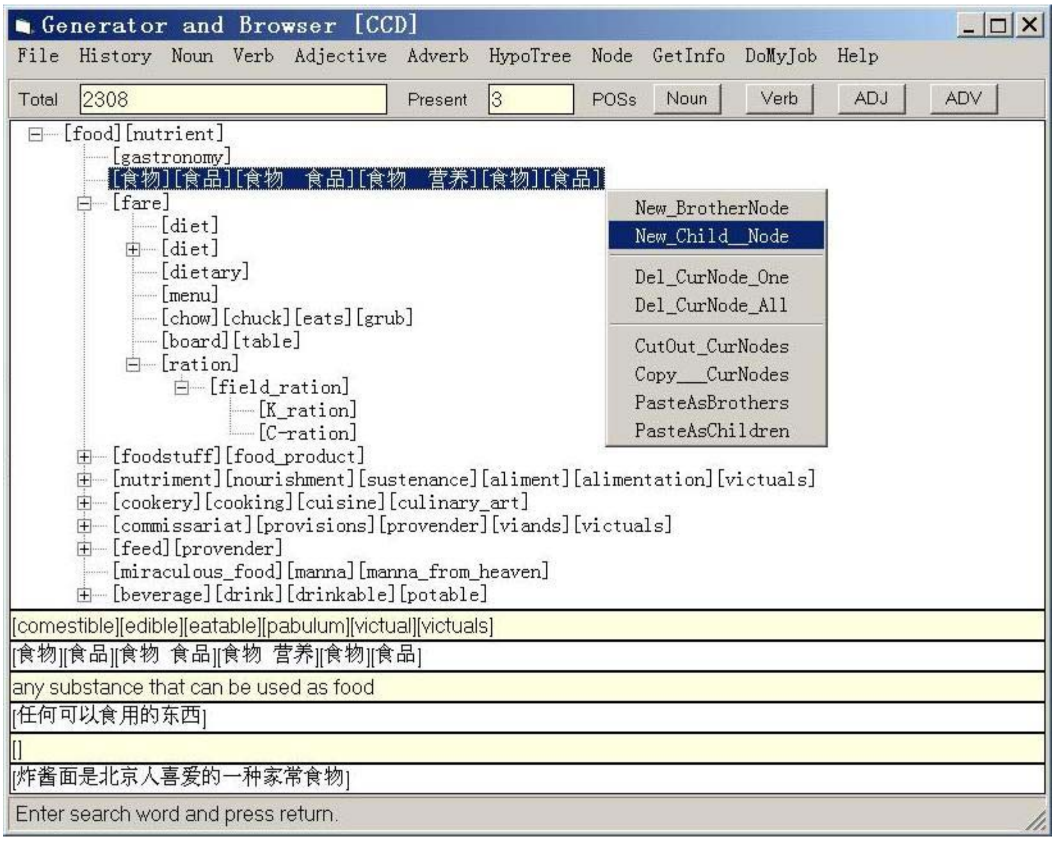
The lexicographers can now operate on the tree freely to express their intended lexical semantics, with correspondingly automatic modifications of the database in the background. It is of much significance that the lexicographers no longer need care for lots of details about the background database as they used to, for the foreground operations by themselves have already carried all the tree-structured lexical semantics and can deal with the net-structured lexical semantics with further work.

Such is the outline of the more common solution for the development of a WordNet-compatible dictionary. Generally speaking, it has provided an easy approach to the evolution of the dictionary.

However, when it comes to the development of ChineseNet, the cases can be a little more complicated. As we have mentioned in the introduction section, we wish that our CCD not only reflects the practice of Contemporary Chinese, but also should be a bilingual one. So, on the one hand, the development works by and large to the above-mentioned common solution with semantic operations to be done, on the other hand, going with each English synset information, the corresponding Chinese synset information is also to be presented.

To cope with the former problem, we have offered the 9 visualized operations. To cope with the latter problem, we would further add the datafields of the peer to peer Chinese items to the background database, and also add the Editbox Controls recording value of the Chinese items to the data-sensitive tree in the foreground.

Thus a tool for the development of the bilingual dictionary CCD has come out as below. The interface view is showing the full-hyponymy tree for the entry word food, which is one of the 25 initial semantic units of nouns in WordNet with the category value of 13.



Total 2308 Present 3

```
⊟ [food][nutrient]
     [gastronomy]
     [食物][食品][食物  食品][食物  营养][食物][食品]
  ⊟ [fare]
        [diet]
     ⊞ [diet]
        [dietary]
        [menu]
        [chow][chuck][eats][grub]
        [board][table]
     ⊟ [ration]
        ⊟ [field_ration]
              [K_ration]
              [C-ration]
  ⊞ [foodstuff][food_product]
  ⊞ [nutriment][nourishment][sustenance][aliment][alimentation][victuals]
  ⊞ [cookery][cooking][cuisine][culinary_art]
  ⊞ [commissariat][provisions][provender][viands][victuals]
  ⊞ [feed][provender]
     [miraculous_food][manna][manna_from_heaven]
  ⊞ [beverage][drink][drinkable][potable]
```

[comestible][edible][eatable][pabulum][victual][victuals]

[食物][食品][食物 食品][食物 营养][食物][食品]

any substance that can be used as food

[任何可以食用的东西]

[]

[炸酱面是北京人喜爱的一种家常食物]

Enter search word and press return.

5 Conclusions and Future Work

Peking University has launched the ChineseNet project since September, 2000, and by now we have fulfiled 10,000 or so Chinese-English concept pairs. Due to the nice features of visualization and interaction of the tree-structure solution, we assuredly have benefited a lot by employing it for the development work. What is more, as the byproducts of these methods and experiences, we even have found some faults of semantic expressing with WordNet 1.6, such as many occurrences of nodes with multi-father in the same category, improper locations of relational pointers in DAT files and others.

In the long run, ICL wants to come to a total amount of 60,000 bilingual concept pairs which might largely meet our need of applications.

Acknowledgement

This work is a component of researches into Chinese Information Extraction funded by

No. 69483003, National Foundation of Natural Science and Project 985 in Peking University.

We are expressly grateful to Prof. Wang Fengxin and Prof. Lu Chuan for unforgettable discussion and support. Thanks also to all the fellows who have participated in and collaborated on the work, among which we would like to mention Mr. Zhang Huarui, Ms. Song Chunyan, Mr. Liu Dong, Ms. Wang Wei, and others.

References

Beckwith, R., Miller, G. A. and Tengi, R. 1993. *Design and Implementation of the WordNet Lexical Database and Searching Software*.

Cook, G. and Barbara, S. 1995. *Principles & Practice in Applied Linguistics*. Oxford: Oxford University Press.

Cruse, D. Alan. 1986. *Lexical Semantics*. Cambridge and New York: Cambridge University Press.

Fellbaum, C. 1993. *English Verbs as a Semantic Net*.

Fellbaum, C., Gross, D. and Miller, K. 1993. *Adjectives in WordNet*.

Fellbaum, C. 1999. *WordNet: An Electronic Lexical Database*. Cambridge, Mass.: MIT Press.

Huang, C.R. et al. 2001. Linguistic Tests for Chinese Lexical Semantic Relations: Methodology and Implications. Report in the Second Workshop on Chinese Lexical Semantics, Beijing, 2001.

Keil, F. C. 1979. *Semantic and Conceptual Development: An Ontological Perspective*. Cambridge, Mass.: Harvard University Press.

Lyons, John. 1977. *Semantics*, 2 vols. London and New York: Cambridge University Press.

Miller, G. A. 1993. *Noun in WordNet: A Lexical Inheritance System.*

Miller, G. A. et al. 1993. *Introduction to WordNet: An On-line Lexical Database.*

Touretzky, D. S. 1986. *The Mathematics of Inheritance Systems*. Los Altos, Calif.: Morgan Kaufmann

Vossen, P. (ed). 1998. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks.* Dordrecht: Kluwer.

Liu Yang
Email: liuyang@pku.edu.cn
Yu Jiangsheng
Email: yujs@pku.edu.cn
Yu Shiwen
Email: yusw@pku.edu.cn
Institute of Computational Linguistics, Dept. of CS
Peking University
Beijing 100871, P. R. China