BUILDING A WINDOWS-BASED BILINGUAL FUNCTIONAL SEMANTIC PROCESSOR

Jonathan J Webster

Department of Chinese, Translation and Linguistics
City Polytechnic of Hong Kong
CTWEBSTE@CPHKVX.CPHK.HK

0. SUMMARY

Under a strategic grant from the City Polytechnic of Hong Kong, work is proceeding toward implementation of a software tool, called the Functional Semantic Processor, which is designed to facilitate analysis of Chinese and English texts along the lines of Systemic Functional Grammar. The Functional Semantic Processor for Chinese and English texts has been developed in the MS Windows environment.

1. INTRODUCTION

Three orientations have formed the basis of the design of the Functional-Semantic Processor: (i) the focus of analysis is on discourse; (ii) the purpose is to discover those elements that contribute to the overall meaning conveyed by a text; (iii) the approach is modular, with each module corresponding to a structural representation from one of several dimensions of meaning. 'In order to provide insights into the meaning and effectiveness of a text', states Halliday (1985), 'a discourse grammar needs to be functional and semantic in its orientation, with the grammatical categories explained as the realisation of semantic patterns'(xvii). SFG provides this functional-semantic orientation.

2. SYSTEMIC FUNCTIONAL GRAMMAR (SFG)

Halliday's systemic-functional grammar (SFG) identifies the structures at clause level and above that contribute to the meaning of a text. An SFG analysis reveals 'how and why' a text means what it does. The clause, as constituent of the text, is the complex realisation of three functional-semantic components: textual, interpersonal and ideational. Each component contributes in its own way to the form of the clause. Corresponding to each component are various kinds of structural representation. In English, textual meaning is represented by theme-rheme and information structures; interpersonal meaning by mood-residue or exchange structure; and ideational or propositional meaning by transitivity structure. While Halliday identifies the clause as the most basic lexico-grammatical unit, nevertheless he also recognises and includes in his grammar other discourse units besides the clause. For example, Halliday's notion of information unit in English is similar to what Tsao (1979) calls topic-chains in Chinese where a topic extends its semantic domain over several sentences.

3. THE FUNCTIONAL SEMANTIC PROCESSOR

The Functional Semantic Processor is a tool to assist in the analysis of texts following M.A.K. Halliday's (1985) approach outlined in his *Introduction to Functional Grammar* (IFG) in terms of theme-rheme structure, information, mood-residue and transitivity. This particular work of Halliday's differs from his previous work in that the emphasis is on the structural rather than the systemic portion of a description of English.

The FSP includes a separate module for each kind of structural analysis. The modules are independent of one another, each has its own unique terminology and organisation. Transitivity structure, for example, includes elements not found in information, mood-residue or theme-rheme structures. Each module assembles a partial solution to the larger question of what is the meaning which the text as a whole conveys.

The approach is similar to that of the blackboard method of problem solving. Different knowledge sources participate in "assembling" a solution. Edward Feigenbaum (1988) calls it '"knowledge assembly" (vi). The four kinds of structural analysis that together comprise the FSP are each a knowledge source, a knowledge module. They are independent of one another, each has its own unique terminology and organisation. Whenever a clause is analysed, by whichever module, that analysis is saved to the black-board. Each module must then look to the blackboard, pulling relevant information from the blackboard to assist in completing its own analysis. The blackboard also plays an active role by alerting modules to any updates to its contents made by other modules.

At present the FSP does two things: (1) it assists the user in doing the analysis; (2) it facilitates the subsequent retrieval of information about the text by collecting all the clause analyses into a global database. In its present form the user must still do the analysis. The user, not the machine, identifies meaningful clause elements, and enters them in the appropriate data field. The computer records the user's interpretation in terms of a functional-semantic analysis. The FSP has up until now been only a tool - a processor, not a parser. As discussed below (Section 5), a parser is being added to the FSP to

perform lower-level syntactic analysis. The output from the parser will permit certain fields to be pre-set.

Implementation of a bilingual version of the FSP will assist in the advancement of research into the similarities and differences between Chinese and English text structures. While Halliday maintains that the three components of meaning — ideational, interpersonal and textual — are universal to all languages, the structures corresponding to these components are, nevertheless, language specific. Thus this bilingual implementation of the FSP includes structures and fields relevant to Chinese and English. Certainly there is some overlap. Transitivity analysis, for example, facilitates comparison on a generally shared level of meaning. But there is also a great deal of difference between Chinese and English, particularly noticeable with respect to exchange or Mood-residue structure. McDonald (1992), in his outline of a FG for Chinese, does not recognise a separate Mood-Residue structure for Chinese.

The Functional-Semantic Processor (FSP) has been developed in the MS Windows 3.11 environment. Chinese is handled in the Windows environment using PC Express TwinBridge v.3.2. The FSP includes three linked components: (i) MS Word for Windows v.6.0 (WinWord) as text-editor; (ii) the Analyst's Interface (AI); and (iii) a database facility. From WinWord, the analyst can view the text, and select clauses for analysis. The AI, a Visual Basic application, is linked by means of OLE automation with WinWord. Working within the AI, the analyst can view, edit, and save the contents of fields on forms for each of the different kinds of structural analysis. Using VB's database capability, entries are saved to a database. The database entries may be retrieved from within the AI, or separately using MS Access. MS Access may also be used to generate reports including, for example, consistent patterns of usage occurring in the data.

4. THE ANALYST'S INTERFACE

The user opens a document in WinWord, highlights the clause to be analysed, presses a button in the toolbar labelled [FSP]. The text highlighted in WinWord now appears in the [Clause] textbox in the Analyst's Interface (AI). The document name and the bookmark for the selected text appear in the status line at the bottom of the AI window (Fig. 1). From the menu, the user chooses which analysis to perform on the selected clause. Once chosen, WinWord is minimized to an icon and the [FSP] button is deactivated. In this case, the user chooses to do transitivity analysis on the selected clause. Transitivity analysis is concerned with the semantic configuration of process, participant, and circumstance, i.e. the clause as a means of representing the processes of doing, thinking, being, speaking, etc.

The clause to be analysed appears in the [Clause] textbox. It has been selected from the Help file for MS Windows Write (Chinese and English versions).

在您的文件中，以上標題或下標題出現的文字，不會在您的文件中顯現出來。

The text that appears as a header or footer in your document does not appear in your document.

In this sentence, the process of the main clause is, in English *appear*, in Chinese 顯現. It is a material process, and the participant functioning in the role of actor is *The text that appears as a header or footer in your document*, or, in Chinese, 在您的文件中，以上標題或下標題出現的文字. We enter this information into the form identified by the tab at the bottom labelled Transitivity-[1] (Fig. 2). Using a mouse the user highlights the process, drags the selected text 顯現 to the [Process] textbox and drops it in. Likewise for the participant 在您的文件中，以上標題或下標題出現的文字, one drags the selected text from the [Clause] textbox to the [Participant] listbox. Once it is dropped into the listbox, a dialog box pops up to elicit from the user which role that participant is performing in the clause (Fig. 3).

This participant, which is also the subject, is a clause consisting of process, participant and circumstance:

在您的文件中 ***In your document*** **circumstance**	以上標題或下標題 ***as header or footer*** **participant**
出現 ***appears*** **process**	的文字 ***the text*** **participant**

It may be argued that this clause-as-subject, translated in English as *The text that appears as a header or footer in your document* is what Halliday identifies as a circumstantial identifying clause in which the process (note that in Chinese it is not the same predicate as occurs in the main clause) encodes a circumstance of manner - how the text appears in the document - as a relationship between the two participants (1985:120).

The user must open a second transitivity form, Transitivity - [2], to analyse the clause-as-subject (Fig. 4). There are now two transitivity forms open as indicated by the two tabs, but only one active. The tab of the active form is yellow in colour. To switch between forms, the user simply needs to click on the tab for that form.

Returning to the main clause, the circumstance of where the text appearing as header or footer does not appear is

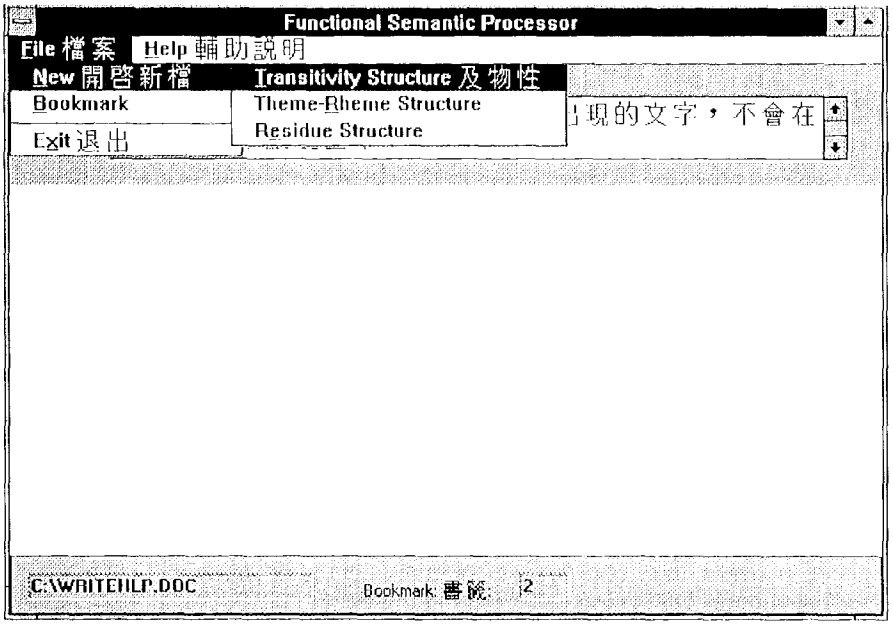

Figure 1

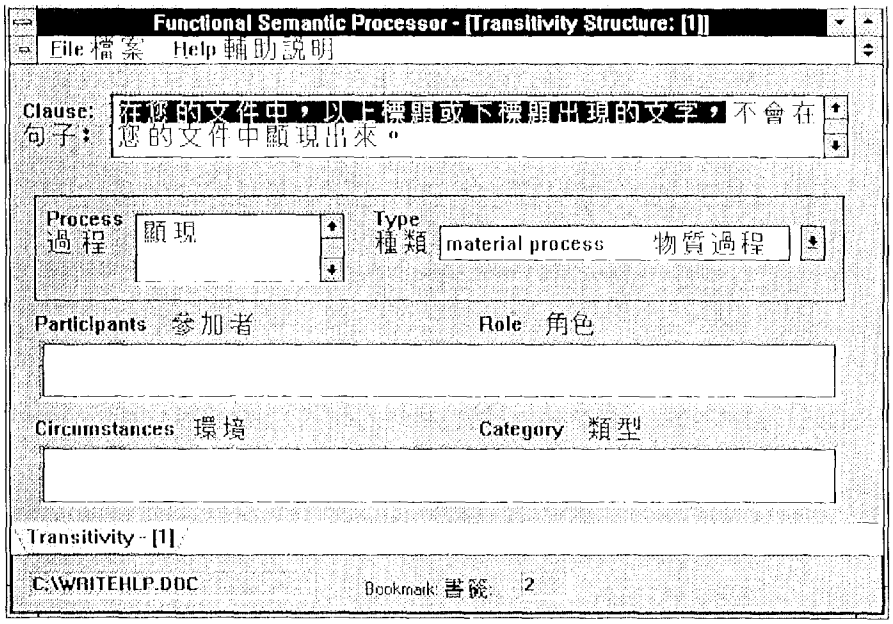

Figure 2

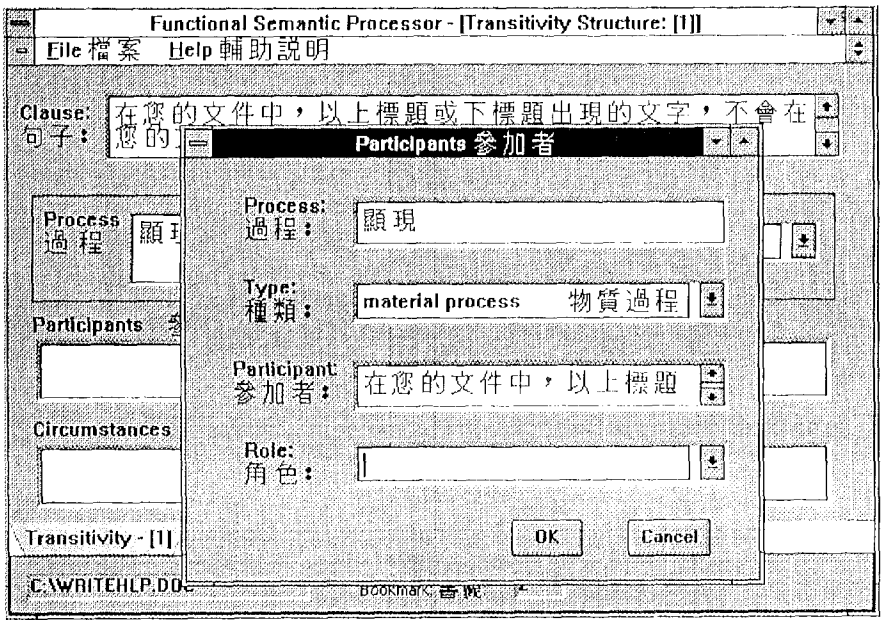

Figure 3

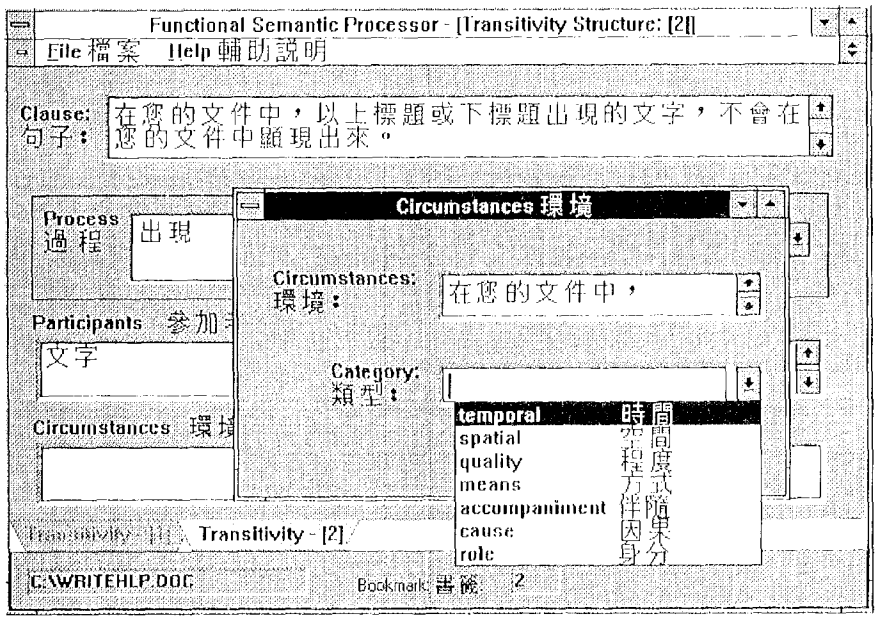

Figure 4

in your document, 在您的文件中. Notice that as in the English sentence, this prepositional phrase occurs twice, but that the first occurrence in the Chinese sentence at the beginning of the sentence belongs to the clause-as-subject, not the main clause.

A circumstance, once highlighted in |Clause|, is dragged and dropped into the |Circumstances| listbox. A dialog box appears asking the user to identify the type of circumstance, whether temporal, spatial, etc. The circumstance and its type then appear in the |Circumstances| list box.

Once an analysis is completed, it is saved to the database along with the name of the document in which the clause appears, the bookmark identity of the clause, and the type and number of analysis, e.g. Transitivity - |1|. Bookmarks, a feature of WinWord for tagging pieces of text, are automatically assigned to the highlighted text when the user presses the |FSP| button in the toolbar. If I highlight a previously analysed clause in a WinWord document, and activate the FSP, all analyses related to that clause will be retrieved and displayed in the FSP. I may then switch between the analyses by clicking with the mouse on the tabs.

As illustrated above, transitivity analysis is concerned with the semantic configuration of process, participant(s), and circumstance(s). Other considerations, such as whether a participant is the grammatical subject or object, whether or not a participant is mentioned, or where the elements appear in the clause, are dealt with by other structures.

5. AGENDA FOR FURTHER DEVELOPMENT

Beyond improving the design and implementation of a bilingual FSP for English and Chinese, we look forward to extending research in this area by automating the analytical process for extracting information from the text being analysed. SFG's weakness is most apparent when it comes to performing lower level syntactic analysis. It is here that we rely on Kit Chunyu(1993)'s LFG parser to provide input for subsequent semantic interpretation. The Kit Parser, originally developed in PDC Prolog for the DOS environment, is being re-developed as a Windows application using LPA WinProlog.

Kit's unificational active chart parser (UniPureChart) is based on a Chinese Lexical-Functional Grammar (C-LFG) constructed by Kit. The FSP will display the parse of a sentence on an 'f-structure' form in the AI. As with other structures, e.g. transitivity structure, the user may edit the parse result on the form prior to saving it to the database.

Attention must be given to the interaction, via the database, between the various modules that make up the FSP, each representing a type of structural representation, a dimension of meaning.

I do not anticipate fully automatic functional-semantic analysis of texts. Even with the addition of a parser, and a system for facilitating module interaction, the FSP will continue to require human participation in the analytical process.

6. APPLICATIONS

The FSP is being used to study the transitivity, theme-rheme, and information structures of Windows on-line Help documentation. The study is looking at both Chinese and English versions of on-line Help. The database of clause-level analyses facilitates the search for consistent patterns of usage.

Greater automation of the functional semantic analytical process raises significant implications for contrastive studies most immediately, and ultimately, machine translation. Providing greater insight into the structures and their elements in English and Chinese that realise meaning can only serve to advance the prospects for an improved state of the art in machine translation.

7. REFERENCES

|1| Feigenbaum, E. (1988). Forward to R. Engelmore & T. Morgan (Eds), Blackboard Systems, New York: Addison-Wesley, pp.v-viii.

|2| Halliday, M.A.K. (1985). Introduction to Functional Grammar, London: Edward Arnold.

|3| Kit, C.Y. (1993). A Chinese Lexical Functional Grammar for Sentence Parsing. MPhil Dissertation, City Polytechnic of Hong Kong.

|4| McDonald, E. (1992). Outline of a Functional Grammar of Chinese for Teaching Purposes. Language Sciences, 14, 4. pp.435-448.

|5| Tsao, F. F. (1979). A Functional Study of Topic in Chinese: The First Step Towards Discourse Analysis. PhD Dissertation, University of Southern California.