Multi-Purpose Development and Operation Environments for Natural Language Applications

Sergei Nirenburg, Peter Shell, Ariel Cohen, Peter Cousseau, Dean Grannes and Chris McNeilly
Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA   15213
{sergei, pshell, arik, cousseau, grannes, mcneilly}@cs.cmu.edu

1 Introduction

Interactive user environments have been a central efficiency-enhancing feature of many modern computer applications, including natural language processing. There are two major classes of users for whom NLP environments can be constructed — developers and end users, such as technical writers and translators. Developers need help in the various knowledge acquisition tasks, such as dictionary and grammar writing for NLP applications. End users look for efficiency enhancements in document preparation beyond the level of word processing support. There are two approaches to the solution of this problem. A dedicated workstation can be developed for each of the required functionalities. Alternatively, workstations can be configured as sets of application routines attached to a universal user interface.

In this report we describe a general-purpose user environment, under development at the Center for Machine Translation of Carnegie Mellon University, capable of supporting a number of dedicated workstation configurations. Among the types of end users whom this system will benefit are technical writers, text revisors and translators. In the framework of NLP system development this tool supports dictionary and ontology acquisition.

A number of separate functionalities included in this system have been developed and used either in commercial word processing software packages or in NLP projects (e.g., the translator's tools described in Macklovitch, 1989, ; and the developer environments IRACQ (Ayuso et al., 1987), LUKE (Wroblewski and Rich, 1988) or ONTOS (Monarch and Nirenburg, 1989), among many others). Our system allows a merge of the two directions in the tool development. One direct reason to put the two previously separate kinds of functionality into a single system was to support the knowledge-based machine-aided translation environment which involves an interactive human editor who uses an interface to help the machine understand the source text.

A standard Unix- and X-windows-based workstation platform has been selected for our system, whose working name is TWS, for "Translator's Workstation." We also used the C-based X11 toolkit called MOTIF (Motif, 1991) and its CommonLisp interface called CLM (Babatz et. al., 1991). In practice, TWS consists of a number of application (functionality) modules which are integrated through the central MOTIF-based user interface module. For reasons of uniformity, each of the applications uses the facilities of the user interface for display and input. Each module uses a standard window to interact with the user, and each window has *standard* menus which,

among other functionalities, allow the user to invoke any other module. Each module also has *special* menus.
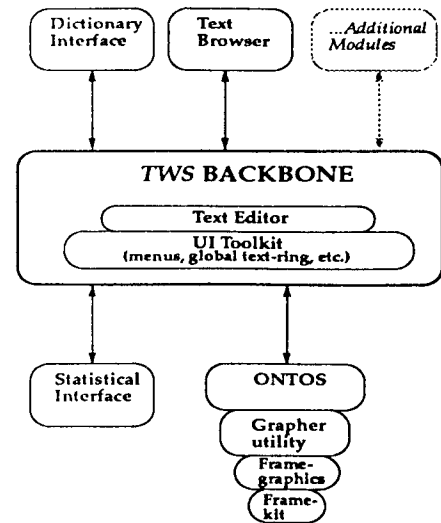
The architecture of TWS is presented in Figure 1.

Figure 1: TWS architecture. The system was designed to make it easy to plug new modules into TWS.

We illustrate the basic interface and several application utilities that have been connected to the TWS. Specifically, we illustrate the text editor (an end-user functionality), the utility for statistical processing of corpora (useful both for end users and for developers) and the ontology acquisition and maintenance tool, based on ONTOS (a developer functionality).

Due to the lack of space, we do not describe our interface to reference sources, such as online dictionaries or user-defined glossaries, or our interface-building interface which allows both the TWS developer and TWS user define and modify the user interfaces the system uses.

TWS is designed in such a way that various utilities can be *imported* into the system, so that there is no need to implement every functionality locally. One example of an existing imported module in TWS is a set of bilingual text alignment routines. The concept of importing utilities is very important to our work. The user interface facilities of the kind we developed makes it easy to import applications with a minimum of effort and with a capability of making them "look and feel" similarly to the rest of the applications in the system. In our future work we intend to mix the development of utilities with importing and integrating existing applications, whenever that is desirable and feasible.

Figure 2 shows a basic editing window of TWS. The central feature of the TWS editor is its capability to emulate various existing word processors. This feature is important from the standpoint of acceptance of the workstation environments by end users. Currently the editor emulates WORD PERFECT[TM] and EMACS.

Figure 3 illustrates the corpus processing functionality. It can be used by end users to compare the usage of a term in various places in a set of documents and by the developer to gain insights for dictionary and ontology acquisition.
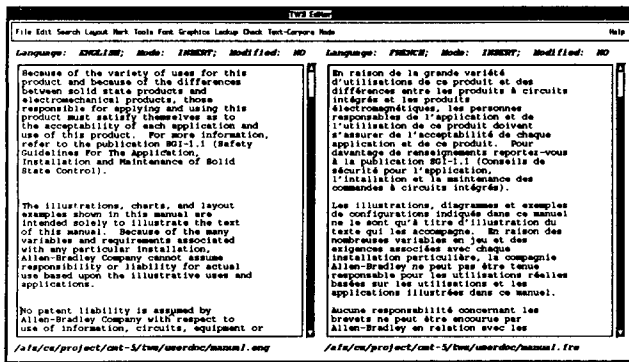
255

Figure 2: The TWS EDITOR Window. The source and target texts can be aligned paragraph by paragraph. The windows can be scrolled in parallel.
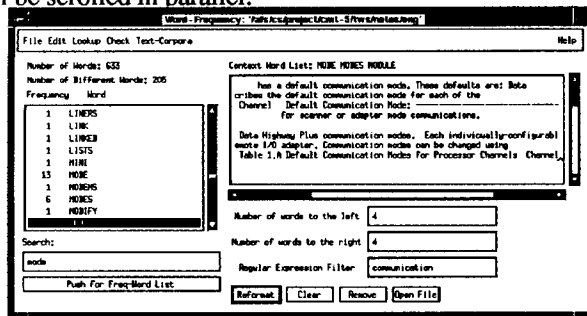

Figure 3: A Word-Frequency Window. The sub-window on the left shows frequency counts for each word in the chosen text, while the right sub-window shows the context of selected words.

A detailed discussion of the ONTOS system see in Monarch, 1989, Carlson and Nirenburg, 1991. To include the ONTOS functionality into TWS, a new grapher has been built. The grapher facilitates browsing through, editing, creating and deleting frames. An important feature is its ability to handle graphs rather than trees, so that multiple inheritance can be graphically represented. The ONTOS module inside TWS also supports a structured editor for manipulating ontological concepts symbolically.
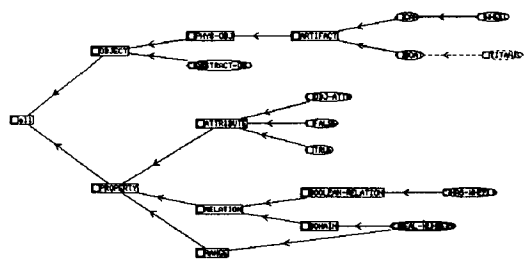

Figure 4: A sample ontology. The user may dynamically select portions of the ontology to display.

We believe that the TWS GRAPHER compares favorably with other graphers, notably the AGGREGRAPHS (see Dannenberg et. al., 1991) which is part of the GARNET system and the ISI

GRAPHER (see Robins, 1988). Unlike AGGREGRAPHS, which can only *display* a knowledge base, our grapher lets the user edit it. As for the ISI GRAPHER, the latter cannot easily support more than one type of link between nodes, while the TWS grapher can support an unlimited number of links, as well as validity and consistency tests for them.

2 Conclusions

Our work on TWS is continuing. At the time of this writing, several additional functionalities are being developed, including a machine-tractable lexicon acquisition module and a user interface system for a machine-aided translation environment, a descendent of the augmentor module of the KBMT-89 MT system (Brown, 1991). On the end-user side, a general text alignment facility is being developed, which will support the work of mono- and multilingual document revisors by pinpointing the differences among different versions of a document.

References

Ayuso, D., V. Shaked and R. Weischedel. 1987. An Environment for Acquiring Semantic Information. Proceedings of Annual Meeting of ACL. Stanford, CA.

Babatz, R., A. Backer, C. Beilkeu, T. Berlage, and M. Spenke. 1991. CLM — A Language Binding for Common Lisp and OSF/Motif User Guide and Reference Manual Version 2.0, Technical report, German National Research Center for Computer Science.

Bass, L., B. Clapper, E. Hardy, R. Kazman and R. Seacord. 1990. Serpent: A User Interface Management System. Proceedings of the Winter 1990 USENIX Conference.

Brown, R. 1991. Augmentation. In: K. Goodman and S. Nirenburg (eds.) **KBMT-89: A Case Study in Knowledge-Based Machine Translation**. San Mateo, CA: Morgan Kaufmann.

Carlson, L. and S. Nirenburg. 1991. World Modeling for NLP. CMU-CMT Technical Report.

Dannenberg, R. B., Marchal, P., Kosbie, D., Mickish, A. and Loyall, A. B. 1991. Aggregadgets & Aggrelists Reference Manual. Technical Report, Department of Computer Science, Carnegie Mellon University.

Macklovitch, E. 1989. An Off-the-Shelf Workstation for Translators. Proceedings of the 30th American Translators Conference, Washington, D.C.

Monarch, I. 1989. ONTOS: Reference Manual, Technical report, Center for Machine Translation, Carnegie-Mellon University.

Myers, B., Giuse, D., Dannenberg, R., Zanden, B., Kosbie, D., Pervin, E., Mickish, A., Marchal, P., 1990. Garnet: Comprehensive Support for Graphical, Highly-Interactive User Interfaces. *IEEE Computer*, Vol. 23, number 11.

Robins, G. The ISI Grapher Manual. 1988. Technical Report, ISI, University of Southern California.

Wroblewski, D.A. and E.A. Rich. 1988. Luke: An Experiment in the Early Integration of Natural Language Processing. Proceedings of Second Conference on Applied Natural Language Processing. Austin, TX.