

語彙の勉強を助けるコンピュータ・プログラム

A Computer Program to Aid Vocabulary Study

Myles O'Brien

【Abstract】 A computer program was written for Windows, which allows users to record and recall vocabulary items, i.e., words or short phrases, and their translations. The display of recalled items may be in either language, or in both, and items can be recalled according to how they were classified by the user on entry. The classification system can be freely set up and modified to suit the user's individual needs and preferences. The operation of the program is described, and its usefulness and possible extensions are considered.

【Key words】 Vocabulary study, CALL, CAI

I Introduction

These days electronic aids for language study are commonplace. Dictionaries that are unwieldy in printed form may be carried around in the pocket with ease in the form of an electronic device, which also affords faster lookup. With a personal computer a great variety of more elaborate reference material is available incorporating pictures, sounds, and animations. Annotation software will display the translation of words in a document on the screen at the user's command. Translation software has also made great strides, although it is still very far short of being able to replace human translators. Many programs are also available which are explicitly designed for teaching rather than reference.

One form of study or reference material traditionally employed in language learning is a notebook or a set of cards on which vocabulary items and their translations are written. Preprinted

vocabulary lists and cards are also available, but it is common for learners to record their own items for personal use so that the items are tailored to fit their exact requirements. Typically, a learner will first study the items together in both languages, and then look at the native language or target language version only, trying to recall the equivalent in the other language for each item. It seemed that a computerized version of this process might be useful if it could offer some advantages over the traditional method, so the design and construction of such a program was undertaken using Visual Basic for Windows.

II Design considerations

The days when computer use was a novelty have long since passed. A computerized learning aid must have some clear advantage over its paper equivalent or people will not be inclined to use it. In general, people prefer to read

from paper than from a computer screen, and research shows that such reading is also faster and more effective.¹ Thus, despite many forecasts of the demise of print, electronic books have never really taken off, even though they have undoubted advantages. New display technology will probably change this situation eventually, but at the moment it is only in the area of reference works, where the advantage of computerized lookup is overwhelming, that electronic books have made a significant impact. Therefore, it was decided to introduce two types of lookup capabilities into the vocabulary items program, one based on the item's contents and the other on its category. The categories into which items are classified are decided by the user, and each item may be in multiple categories. As will be shown in detail later, this gives the system great flexibility and power, and sets it clearly apart from what might be accomplished on paper.

The program does not have any printing facilities, so it was decided to store the vocabulary data in a text file of tab-delimited records so that users with the necessary computer knowledge could easily read their vocabulary data into a word-processing, spreadsheet or database program for editing, formatting, printing, etc, while novice users could at least open and print the file as it is.

III Description of the Program

The program is called Tangocho (単語帳) and it was designed with Japanese learners of English in mind. However, it is not necessarily limited to these two languages. With very simple adjustments it could be altered to cater for any pair of languages. The user interface is shown in the Figure 1. In general, the set of items which will be displayed is determined by

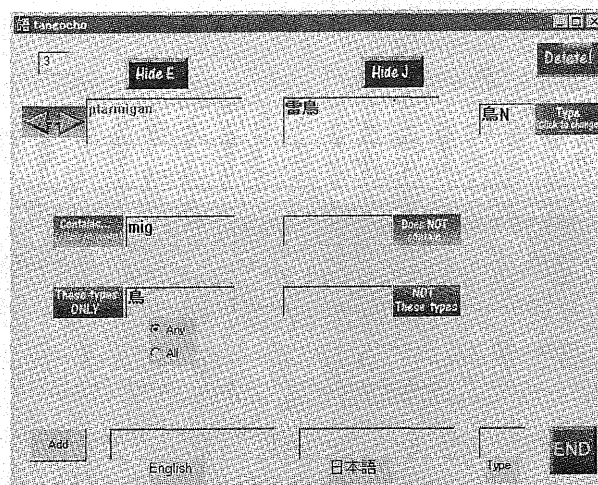


Figure 1 The program's user interface

the boxes in the middle, *Contains*, *Does NOT contain*, *These types ONLY*, and *NOT these types*. If nothing is entered in these boxes, all the items will be displayed in the order in which they were entered. The interface elements will be explained in turn, starting from the top left:

The box in the top left corner shows the sequence number of the item in the data file, 3 in this case. This is just for reference, and is usually not important. The buttons called *Hide E* and *Hide J* will hide the display of the English or Japanese item so that the user may test memorization. When clicked, the button title changes to *Show E* or *Show J*. The *Delete!* button deletes the displayed item.

The back and forward arrowheads at the left of the second row will show the previous or next applicable item when clicked. The same function is available in a more convenient way by pressing the back or forward arrow keys on the keyboard. The next boxes show the English and Japanese (ptarmigan and 雷鳥). Note that the items are not limited to single words; short phrases may also be used. The next item is *Type*. This is where users can customize the system to suit their individual needs exactly. The example here shows 鳥 N,

so the word has been put into the two categories 鳥 and N. 鳥 obviously stands for bird, and N might stand for noun. Any single character, Japanese or English, can be used as a type, and the view can be limited to items from a certain type or set of types (or *not* from certain types). Thus, items related to a certain topic (e.g., law) might be given the type L, and the user can set L as the type to review only items with a law connection. An item's classification can be changed freely at any time, so a user might, for instance, add 済 to the type of items which have been thoroughly learned and suppress their display using the *NOT these types* box, but might check through the 済 items a few months later to confirm recollection.

The next row, *Contains* and *Does NOT contain*, limits the display to those items whose English portion contains (or does not contain) the text in the box. The example shows "mig", so "ptarmigan" is displayed. If the English portions of other items also contains "mig", they will be displayed by clicking the right arrow icon or pressing the right arrow key, provided they conform to any other conditions set up (in this example the type must also be 鳥). The *Contains* box might be used to check through words containing certain suffixes (e.g., "ism" or "ment") or to display a certain word just by typing it in. The *Type* boxes might be sometimes used in conjunction here to help avoid displaying unwanted items (e.g., in the "ment" suffix case, limiting the display to words in a noun category to avoid words like "mention")

The next row specifies the type to display. The *Any* and *All* buttons decide whether the item will be displayed if it belongs to *any* of the specified types, or only if it belongs to *all* of them. Thus it is possible to be highly specific as regards which types to display. As soon as

the user makes any changes in the conditions, the new conditions will automatically be in effect the next time an arrow symbol is clicked of an arrow key is pressed.

The last row is for entering new items. The text may be typed in directly, pasted, or dragged and dropped from another application. Only after clicking the *Add* button will the new item actually be added. The *END* button quits the program.

The vocabulary data is stored in a file called *tangocho.txt* which must be in the same folder as the program file *tangocho.exe*. It is a simple tab-delimited file in the order *English, Japanese, Type*. This affords the advantages mentioned above in the *Design considerations* section, as well as allowing direct editing by more confident users.

IV Conclusion

The program should be a useful aid to vocabulary study for learners who have their own Windows personal computer. Versions are not available for other platforms at the moment, but it is planned to make a Macintosh version if there seems to be a demand for it. The present version is available for download from www.mcn.ac.jp/fa/m.obrien.

The interface has been kept as simple as possible with the avoidance of menus. From one point of view the simplification may have been overdone, in that there is no provision for the use of alternative data files. It was felt that this might give rise to more confusion than help, especially for inexperienced computer users, since the type system allows selection of which data to display. However, it is thinkable that some users might like to have a number of

quite separate data files for different purposes. In such a case, a crude but fairly effective mechanism is available: if separate copies of the program file *tangocho.exe* are kept in different folders, each will work with the *tangocho.txt* file in its own folder. Some care would be necessary in backing up the data files to avoid unwanted overwriting, but it is a workable system and not very wasteful of disk space, since the size of the *tangocho.exe* file is only 92K. Again, if there is a demand, the development of a more sophisticated interface will be considered. One issue which stands out is that of printing: the facility to print subsets of the data based on type should be very useful.

References

- 1 O'Hara, K. and Sellen, A.: A Comparison of Reading Paper and On-Line Documents, CHI 97: Conference on Human Factors in Computing Systems, Atlanta, USA, March 1997. Available at: <http://www.acm.org/sigchi/chi97/proceedings/paper/koh.htm>.
Muter, P. and Maurotto, P.: Reading and Skimming from Computer Screens and Books: The Paperless Office Revisited?, Behaviour & Information Technology, 10, 257–266, 1991. Available at: <http://psych.utoronto.ca/~muter/pmuter2.htm>.