Automation of Educational Tasks for Academic Radiology

David L. Lamar, MD, PhD, Michael L. Richardson, MD, Blake Carlson, MD

Rationale and Objectives: The process of education involves a variety of repetitious tasks. We believe that appropriate computer tools can automate many of these chores, and allow both educators and their students to devote a lot more of their time to actual teaching and learning. This paper details tools that we have used to automate a broad range of academic radiology-specific tasks on Mac OS X, iOS, and Windows platforms. Some of the tools we describe here require little expertise or time to use; others require some basic knowledge of computer programming.

Materials and Methods: We used TextExpander (Mac, iOS) and AutoHotKey (Win) for automated generation of text files, such as resident performance reviews and radiology interpretations. Custom statistical calculations were performed using TextExpander and the Python programming language. A workflow for automated note-taking was developed using Evernote (Mac, iOS, Win) and Hazel (Mac). Automated resident procedure logging was accomplished using Editorial (iOS) and Python. We created three variants of a teaching session logger using Drafts (iOS) and Pythonista (iOS). Editorial and Drafts were used to create flashcards for knowledge review. We developed a mobile reference management system for iOS using Editorial. We used the Workflow app (iOS) to automatically generate a text message reminder for daily conferences. Finally, we developed two separate automated workflows—one with Evernote (Mac, iOS, Win) and one with Python (Mac, Win)—that generate simple automated teaching file collections.

Results: We have beta-tested these workflows, techniques, and scripts on several of our fellow radiologists. All of them expressed enthusiasm for these tools and were able to use one or more of them to automate their own educational activities.

Conclusions: Appropriate computer tools can automate many educational tasks, and thereby allow both educators and their students to devote a lot more of their time to actual teaching and learning.

Key Words: Automation; education; productivity; research.

INTRODUCTION

Academic life and real life abound with a large number of dull and repetitious tasks. We believe that many of these tasks should be handed over to one's computer. This paper details a number of specific tasks that we have automated in our radiological lives, as well as the tools we have used to accomplish this.

Some tasks are quick and easy to automate. Others require a lot of time and programming skills. Thus, one of the very first decisions one needs to make is whether a given task is worth automating. The following diagrams from the xkcd website summarize this issue nicely (1) (Fig 1).

The thought of writing one's own automation code may seem a bit overwhelming to some readers. We, therefore, suggest starting out with baby steps, using simple automation tools that do most of the heavy lifting. As one's skills and confidence increase, it becomes easier to take further steps with more powerful tools.

We also advise learning from the works of others. Finding someone else's example code that does much of what you need is invaluable. Many programmers post their code as open source and intend that it be reused freely. You will not only save a lot of time in your project, but also learn a lot by dissecting such code and by adapting it to fit your ends. Like the world of academic radiology, the coding world is often a gift economy (2), where knowledge is given away freely. As in academic radiology, it is also considered polite and ethical to cite others' programming work and give them credit when you borrow something.

Some automation tools are available on multiple platforms. Others only exist on a single platform. Some are quite easy to use and require no special computer expertise. Other tools have a steeper learning curve and require some basic knowledge of computer programming.

SIMPLE AUTOMATION TOOLS

We will first focus on automation tools that require no special computer expertise. Tools such as AutoHotKey (Windows, free) (3) and TextExpander (Macintosh, $34.95) (4) are specifically designed to make it easy to automate a series of steps.

Acad Radiol 2016; 23:919–932

From the UWMC Roosevelt Radiology, 4245 Roosevelt Way NE, Seattle, WA 98105. Received October 6, 2015; revised December 21, 2015; accepted December 27, 2015. **Address correspondence to:** M.L.R. e-mail: mrich@u.washington.edu

http://dx.doi.org/10.1016/j.acra.2015.12.028

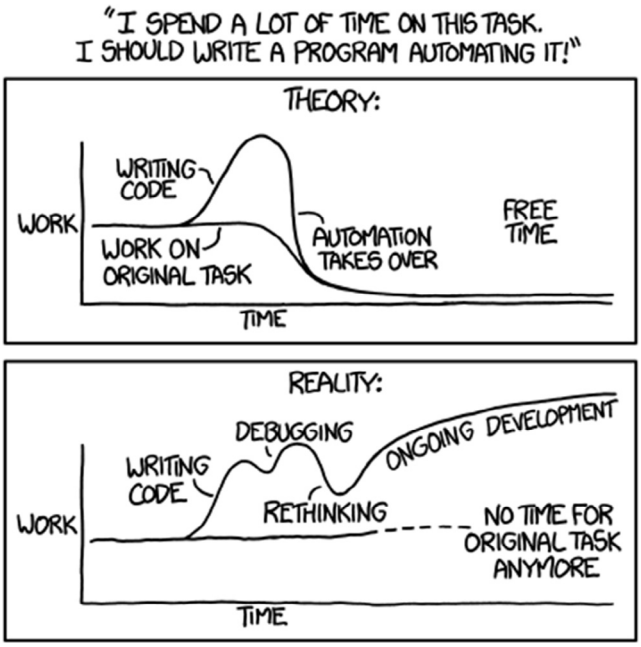

Figure 1. Theory vs reality in automation (figure used with permission from xkcd.com).

It is true that a number of e-mail programs and voice dictation systems offer their own proprietary set of rudimentary text and template macros. However, such macros are limited for use only within that particular program. AutoHotKey and TextExpander, however, create macros that can operate within any other program on the user's computer. Thus, a macro written with AutoHotKey will work on a Microsoft Word document, an e-mail written in Outlook Express, a Google Docs file, or the Windows command line. AutoHotKey and TextExpander also offer more powerful features, such as the ability to embed system commands and statistical calculations within the macro.

SIMPLE TEXT MACROS

Word processing is one of the most basic and necessary tasks performed by most computers. Text automation can rid one of some of the boring parts of word processing. The first step is, therefore, to identify some of one's common repetitive phrases.

Signature Files

One logical place to start is with signature files. These are added at the end of many e-mail messages and most formal letters. Signature files provide an easy-to-understand yet nontrivial example of automation for the less technically adept reader. The signature file macro provides a simple example that can be easily modified to meet some other situation.

TextExpander (4) and AutoHotKey (5) allow one to create "snippets" of commonly used text that can be invoked by typing a short abbreviation. Once typed, these abbreviations

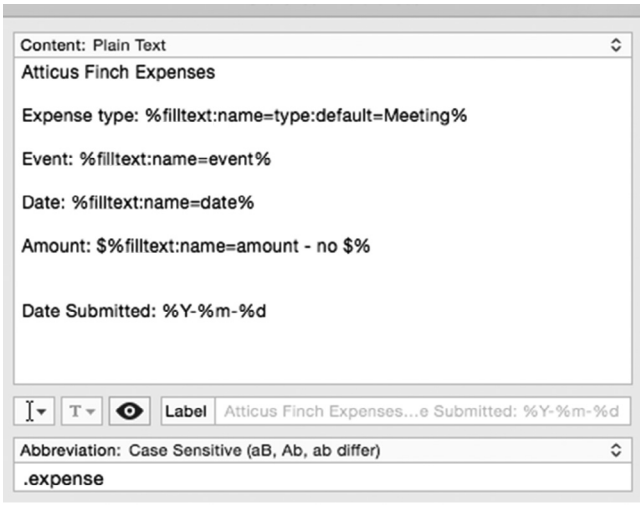

Figure 2. TextExpander macro for recording expenses.

are automatically expanded into the desired text. Ideally, abbreviations that do not normally appear in typed text should be chosen. For example, the abbreviation "afmd" does not pop up in most standard communications. When one of us types this at the end of an e-mail, it is automatically expanded by TextExpander to the following:

Atticus Finch, M.D.
Professor, Department of Radiology
Enormous State University

It is easy to teach these two programs to automatically replace one's commonly mistyped words, eg, replacing "teh" with "the," and "ntoed" with "noted."

Expense Report Template

One can also create special purpose templates for TextExpander and AutoHotKey. One of the common bits of drudgery during business travel is keeping track of expenses. One of us has a special macro for this named ".expense" (Fig 2). When the abbreviation ".expense" is typed in any application (word processor, e-mail client, etc.), a form pops up, pre-populated with the author's name, the time and date, and a few fields to fill in describing the expense (Fig 3).

Once "OK" is clicked, the abbreviation ".expense" is replaced by the following text:

Atticus Finch Expenses
Expense type: RSNA
Event: categorical course
Date: December 2015
Amount: $250
Date Submitted: 2015–08–15

Resident Performance Review Template

Our faculty are asked to create monthly performance summaries for the residents rotating on our services, which are