ABSTRACT

NLTK, the Natural Language Toolkit, is a suite of open source program modules, tutorials and problem sets, providing ready-to-use computational linguistics courseware. NLTK covers symbolic and statistical natural language processing, and is interfaced to annotated corpora. Students augment and replace existing components, learn structured programming by example, and manipulate sophisticated models from the outset. It is intended to support. It is intended to support research and teaching in NLP or closely related areas, including empirical linguistics, cognitive science, artificial intelligence, information retrieval, and machine learning. NLTK has been used successfully as a teaching tool, as an individual study tool, and as a platform for prototyping and building research systems. There are 32 universities in the US and 25 countries using NLTK in their courses. NLTK supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities.

INTRODUCTION

Teachers of introductory courses on computational linguistics are often faced with the challenge of setting up a practical programming component for student a0ssignments and projects. This is a difficult task because different computational linguistics domains require a variety of different data structures and functions, and because a diverse range of topics may need to be included in the syllabus. A widespread practice is to employ multiple programming languages, where each language provides native data structures and functions that are a good fit for the task at hand. For example, a course might use Prolong for parsing, Perl for corpus processing, and a finite-state toolkit for morphological analysis. By relying on the built-in features of various languages, the teacher avoids having to develop a lot of software infrastructure.

An unfortunate consequence is that a significant part of such courses must be devoted to teaching programming languages. Further, many interesting projects span a variety of domains, and would require that multiple languages be bridged. For example, a student project that involved syntactic parsing of corpus data from a morphologically rich language might involve all three of the languages mentioned above: Perl for string processing; a finite state toolkit for morphological analysis; and Prolong for parsing. It is clear that these considerable overheads and shortcomings warrant a fresh approach.

Apart from the practical component, computational linguistics courses may also depend on software for in-class demonstrations. This con- text calls for highly interactive graphical user interfaces, making it possible to view program state (e.g. the chart of a chart parser), observe program execution step-by-step (e.g. execution of a finite-state machine), and even make minor modifications to programs in response to “what if” questions from the class. Because of these difficulties it is common to avoid live demonstrations, and keep classes for theoretical presentations only. Apart from being dull, this approach leaves students to solve important practical problems on their own, or to deal with them less efficiently in office hours. describe NLTK, the Natural Language Toolkit, which we have developed in conjunction with a course we have taught at the University of Pennsylvania. The Natural Language Toolkit is available under an open source license from http://nltk.sf.net/. NLTK runs on all platforms supported by Python, including Windows, OS X, Linux, and Unix.

National Significance

Natural Language Processing is often taught within the confines of a single-semester course at advanced undergraduate level or postgraduate level. Many instructors have found that it is difficult to cover both the theoretical and practical sides of the subject in such a short span of time. Some courses focus on theory to the exclusion of practical exercises, and deprive students of the challenge and excitement of writing programs to automatically process language. Other courses are simply designed to teach programming for linguists, and do not manage to cover any significant NLP content. NLTK was originally developed to address this problem, making it feasible to cover a substantial amount of theory and practice within a single-semester course, even if students have no prior programming experience.

International Significance

A significant fraction of any NLP syllabus deals with algorithms and data structures. On their own these can be rather dry, but NLTK brings them to life with the help of interactive graphical user interfaces that make it possible to view algorithms step-by-step. Most NLTK components include a demonstration that performs an interesting task without requiring any special input from the user. An effective way to deliver the materials is through interactive presentation of the examples in this book, entering them in a Python session, observing what they do, and modifying them to explore some empirical or theoretical issue. Apart from being dull, this approach leaves students to solve important practical problems on their own, or to deal with them less efficiently in office hours. describe NLTK, the Natural Language Toolkit, which we have developed in conjunction with a course we have taught at the University of Pennsylvania. The Natural Language Toolkit is available under an open source license from http://nltk.sf.net/. NLTK runs on all platforms supported by Python, including Windows, OS X, Linux, and Unix.