

# NLP (Natural Language Processing) for NLP (Natural Language Programming)

Rada Mihalcea<sup>1</sup>, Hugo Liu<sup>2</sup>, Henry Lieberman<sup>2</sup>

<sup>1</sup> Computer Science Department, University of North Texas  
rada@cs.unt.edu

<sup>2</sup> Media Arts and Sciences, Massachusetts Institute of Technology  
{hugo,henry}@media.mit.edu

**Abstract.** Natural Language Processing holds great promise for making computer interfaces that are easier to use for people, since people will (hopefully) be able to talk to the computer in their own language, rather than learn a specialized language of computer commands. For programming, however, the necessity of a formal programming language for communicating with a computer has always been taken for granted. We would like to challenge this assumption. We believe that modern Natural Language Processing techniques can make possible the use of natural language to (at least partially) express programming ideas, thus drastically increasing the accessibility of programming to non-expert users. To demonstrate the feasibility of Natural Language Programming, this paper tackles what are perceived to be some of the hardest cases: steps and loops. We look at a corpus of English descriptions used as programming assignments, and develop some techniques for mapping linguistic constructs onto program structures, which we refer to as programmatic semantics.

## 1 Introduction

Natural Language Processing and Programming Languages are both established areas in the field of Computer Science, each of them with a long research tradition. Although they are both centered around a common theme – “languages” – over the years, there has been only little interaction (if any) between them<sup>3</sup>. This paper tries to address this gap by proposing a system that attempts to convert natural language text into computer programs. While we overview the features of a natural language programming system that attempts to tackle both the descriptive and procedural programming paradigms, in this paper we focus on the aspects related to procedural programming. Starting with an English text, we show how a natural language programming system can automatically identify steps, loops, and comments, and convert them into a program *skeleton* that can be used as a starting point for writing a computer program, expected to be particularly useful for those who begin learning how to program.

We start by overviewing the main features of a descriptive natural language programming system METAFOR introduced in recent related work [6]. We then describe in detail the main components of a procedural programming system as introduced in this

---

<sup>3</sup> Here, the obvious use of programming languages for coding natural language processing systems is not considered as a “meaningful” interaction.