

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/329736252>

A step towards a Natural Language Programming Tool (NLPT)

Conference Paper · June 2014

CITATIONS

0

READS

83

6 authors, including:



Manori Gamage

Sri Lanka Institute of Information Technology

2 PUBLICATIONS 0 CITATIONS

SEE PROFILE



Nuwan Kodagoda

Sri Lanka Institute of Information Technology

50 PUBLICATIONS 112 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Road Infrastructure Data Collection and POC [View project](#)



SLIIT - eLearning Research Group [View project](#)

A step towards a Natural Language Programming Tool (NLPT)

K. J. A. Perera¹, K. A. I. Kuruppu², M. P. Gamage³, J. A.P.B. Jayakody⁴, K. S. G. S. Gunasekara⁵, G. Nuwan Kodagoda⁶

Sri Lanka Institute of Information Technology, Malabe Campus, Sri Lanka

jude.asanka@gmail.com¹, indika_kuruppu@yahoo.com², manorigamage@yahoo.com³,
padma_jayakody@yahoo.com⁴, geeth@yahoo.com⁵, nuwank@sliit.lk⁶

Abstract

This research project is taking a step towards developing a tool that can facilitate Natural Language Programming (NLP). As a result Natural Language Programming Tool (NLPT) is designed that generates source codes in response to Natural Language (NL) instructions. The major issue that needs to be addressed is identifying the meaning of the user's NL instruction. The research is aimed at solving the fundamental problems of symbolic NL understanding which is an evolving research area. As NL consists of a broad vocabulary Basic Mathematical Calculations and Basic Input/Output Operations are selected as the project domain. NLPT can be used as a programming tool to generate the relevant source code, plays as an interactive programming learning tool and serves as a brainstorming tool.

Key Words

Natural Language Programming, Artificial Intelligence, Regular Expressions, Hashing Technique

1. Introduction

NLP [1] is a rapidly growing field in Artificial Intelligence. NLP is intended to enable users to communicate with computers using NL the kind of language humans use to communicate with each other.

Creating computational systems that can understand NL has remained an elusive goal because of the tremendous complexity, ambiguity, and context-dependency of NL. The definition of "understanding" is one of the major problems in NLP. Understanding language means, among other things, knowing what concepts a word or phrase stands for and knowing how to link those concepts together in a meaningful way.

NL generation systems convert information from computer databases into normal sounding human language, and NL Processing systems convert samples of human language into more formal representations that are easier for computer

programs to manipulate. NL Processing can offer real advantages in summarization, information extraction, and on-line help such as dictionary support, "NL to code" tools becomes very useful as programming learning tools.

The intended users of the NLPT are mainly beginners and intermediate programmers. NLPT is considered to be a novel system which can accomplish at least three sub goals;

- Assist novice programmers in developing intuitions about programming.
- Facilitate intermediate programmers with an outlining tool, ahead of writing.
- Serves as a brainstorming tool for researchers.

2. Methodology

In NLPT, users are allowed to enter their problem as a NL sentence through a user interface. The NLPT is able to analyze and understand the NL. NLPT uses this information

to produce the relevant source code, in response to the NL instructions. As NL consists of a vast area the project domain is restricted to Basic Mathematical Calculations and Basic Input/Output Operations. One of the special features of NLPT is its ability to identify user mistakes but still to proceed on with the user's request. To enhance NLPT use as a learning tool, an English explanation of the word is generated in pop-up window when a user mouse over a particular word of the code.

The figure 1 displays the interactions among the main components of the NLPT.

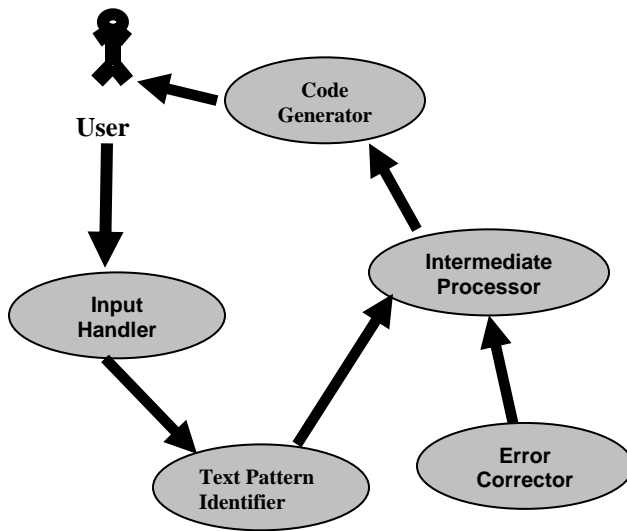


Figure 1: Interactions among the NLPT components

2.1 Input Handling module

NLPT uses a very user friendly GUI to handle input. In NLPT Graphical User Interface (GUI) there are 3 panels. Left panel is used to enter the user's instructions and the right panel is used to display the output. The bottom panel displays the error notifications and messages for the user. *Figure 2* displays the NLPT GUI.

Basic file operation functions are provided in the GUI (such as open, save, etc.) for the convenience of the user.

This module read the user input line by line and removes unwanted characters (space, tabs, etc.). Then the text is forwarded to text identification module to be processed.

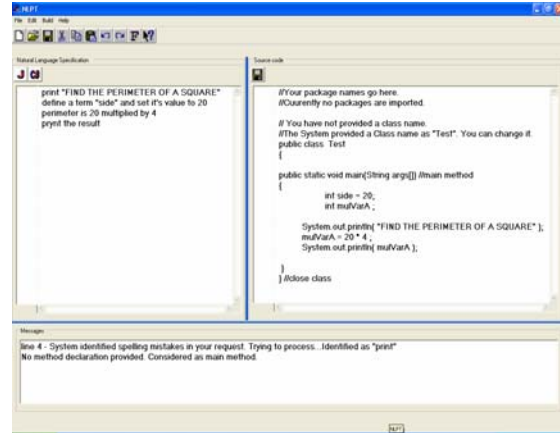


Figure 2: NLPT user interface

2.2 Text Pattern Identification Module

This module is the core of the system. It first identifies what the user has asked to perform with the help of keywords repository and regular expressions.

Regular expressions are used to split and extract important data elements. These words are again matched with results of previously generated outputs. As an example if the user uses pronouns such as "it", "that" etc, it refers to the previous answer of early inputs. Previously processed answers are kept in a stack and reference to the stack is made if necessary. After identifying the meaning of the input text, this module redirect derived data in to intermediate code generation module to generate a defined set of intermediate code to a file.

2.3 Error Correction Module

If the system could not identify the matching words corresponding to the Hash Table, then words in the line of text are directed to soundex module. Soundex returns a set of suggested words that sounds like the given word. Then those suggestions are matched with keyword repository. If these words are also not matched then NLPT ignores the input as incorrect input.

Also this error correction module is used to generate appropriate warnings and error messages with relevant error codes. As example if the system could not find the minimal number of data to process, it generates a warning message and error code. This leads to high degree of maintenance and debugging.

2.4 Intermediate Processing Module

The NLPT uses intermediate code generation module to convert identified information in to well defined set of codes. As the intermediate code structure team uses the Java class file format. This module is independent from any programming language concepts.

Since NLPT generates an intermediate code for identified language patterns, the system can easily be expanded in to other languages such as C++. Then no extra effort to understand the NL pseudo code is needed.

2.5 Source Code Generation Module.

This module uses the intermediate code structure to get information. The major out put of the NLPT is high level source code for the given NL instruction. The NLPT's generated source code may not be similar to an expert written source code. Rather this tool can be used to as a help to write programs and as a tool to learn basic programming concepts.

3. Research Findings

▪ Identified NL-Processing Method

Since the NL is very ambiguous there are no proper algorithmic methods to achieve the solution. Therefore the development team has identified several methods that could be used to achieve the expected goals. After the evaluation Regular expressions with Hashing technique was selected as the NL Processing method.

In this approach Regular Expressions [3] are used to identify the user request. Before processing the user's input the system has to identify the request type. This method can be combined with the hashing technique to identify the request. Using a Hash Table [2], similar words can be categorized in to one group and can be given the same value to that group.

In this method source code for a particular pattern or set of words can be reduced. Using regular expressions all the important words can be separated without worrying about the occurrence of other unnecessary words in the instruction. This reduced the effort to identify data elements in the given text. As an example numbers can be easily distinguished from other words in the given text. Then type casting and

other operations over those numbers can be done very easily without any exceptions or errors.

▪ Identified Error Identification Methods

One of the main disadvantages of many of the modern Programming Languages is that they are not very user friendly. Other than it acting as an unfamiliar language to humans, once a person makes a mistake in a single letter in a word, the program generates error messages making the user confused. One of the special features of NLPT is it's ability to identify user spelling mistakes without ignoring the whole input as a meaningless thing. The identified solution of the team is the SoundEx [4].

SoundEx is perhaps the most infamous text searching algorithm around. SoundEx promises a great deal, that of matching words with similar sounding words, and actually delivers at best, a large number of accurate matches. It may sometimes fail depending on the word. For example, 'Sinclair' and 'St. Clair' may fail. Finally, it should be understood that SoundEx, may sometimes provide pitfalls as well as lending a hand.

▪ NLPT as a Learning Tool

One of the objectives of the research is to develop NLPT as a programming learning tool. The NLPT could be used as a tool to learn programming by having a user friendly interactive GUI and allowing them to watch the results of their NL input of a particular command. Also the team tries to enhance it's usability by introducing a special feature, that is when a user mouses over a particular word of the code, an English explanation of the code is generated in pop-up window.

4. Future Enhancements

- In future NLPT could be expanded to support a wider range of instructions relating to other domains too.
- In order to reduce the problems faced by the highly ambiguity nature of NL, a mechanism to facilitate a dialogue between the user and NLPT can be introduced. NLPT can ask user what it doesn't understand, so that user can supply missing information, and correct any misconception.

- Use a Programming by Example method, which combines NL descriptions with example based demonstrations. Sometimes it is easier to understand demonstrations rather than word descriptions.

5. References

- [1] *An Introduction to Natural Language Processing.* (06/04/2006). [online]. Generation5. <<http://www.generation5.org/content/1999/nlpoverview.asp>>. [15/03/2006].
- [2] *Hash list-Hash table meets Array List.* (23 May 2003). [online]. The Code Project. <<http://www.codeproject.com/csharp/hashlistarticle.asp>>. [05/04/2006].
- [3] *C# Regular Expressions.* (01/18/2001). [online]. O'Reilly. <http://www.windowsdevcenter.com/pub/a/oreilly/windows/news/csharp_0101.html>. [02/04/2006].
- [4] *Understanding Classic SoundEx Algorithms.* [online]. Creativyst Docs.ss <<http://www.creativyst.com/Doc/Articles/SoundEx1/SoundEx1.htm>>. [03/04/2006].