

CODING BY VOICE AS A STAGE OF PROGRAMMING EVOLUTIONARY PROGRESS

ADAM PARFIANOVICH, ANATOLIY NADOLSKIY

Belarusian State University of Informatics and Radioelectronics, Minsk

Summary: The article researches the pros and cons of coding by voice. This mode of programming can become a useful tool for routine coding and it can also help people, especially handicapped, to control their electronic devices easily.

The main goal for a programmer is not just simply write a program suitable to fulfill the task by means of any language structures. The main goal is always a set of interesting solutions how to implement these structures. In fact, a programmer needs to translate the tasks to a computer in the most laconic and appropriate form.

Programming as a task can be divided into four stages:

- 1) Writing
- 2) Debugging
- 3) Compilation
- 4) Performance

A programmer usually writes a short section to solve a concrete task. Then this part of code is compiled and, in case the compilation is successful, it starts and is tested with debugging tools. When the code performs correctly it is expanded to solve a bigger task. The process is repeated till the code solves the whole task. Thus coding is followed by correcting, testing and debugging. A programmer usually works with mediums that offer tools to fulfill all four actions, e.g. Microsoft Visual Studio.

Rather logical question can be brought up as the following: are so various programming languages too necessary to use and to learn without approaching to the direct communication with a system. To answer this question we should as soon as possible make at least several steps in our way to the "dream." Coding by voice is one of these steps.

How coding by voice works. Handicapped people and those who would prefer to code from their smartphones should become the main consumers of coding by voice. For example, Carpal tunnel and repetitive strain injuries can prevent programmers from typing for month at a time. Clearly, that smartphones have been being popularized for long by February 2019. Even cars can be controlled by voice commands through smartphones, and more and more ideal artificial intellects to search information or to communicate with other people are appearing.

So, what is the reason for the coding by voice not to be developed yet? There is an opinion that a lot of people are afraid of such simplifying of coding because it will lead to deterioration of coding quality (that will cause speed reduction of programs performance) and wage declines subsequently. One more crucial factor is a very small opportunity to make profits when coding commercial software. Also there is an existent horrible factor that thieves do not even need to crack your computer to steel the code, if necessary; they will just listen to your voice for some period of time. It is, certainly, simplify the task for them and complicate it for honest citizens: because the techniques to defend it will naturally subject to material and spiritual resources of humanity. It is understandable now that the idea to code some secret projects by voice even sounds rather horrible.

Software. Code by voice is fortunately possible by replacing the keyboard with speech recognition, for example, as David Williams-King writes Linux systems. The key is to develop a voice grammar customized for programming. A community has evolved around hacking the commercial Dragon NaturallySpeaking to use custom grammars, but this method suffers from fragmentation, a steep learning curve, and frustrating installation difficulties. In an attempt to make voice coding more accessible David Williams-King created a new speech recognition system called Silvius that is built on open-source software with free speech models. It can run on cloud servers for easier setup, or locally for the best latency. David Williams-King and his collaborators have also prototyped a hardware dongle which types Silvius keystrokes using a fake USB (Universal Serial Bus) keyboard, and requires no software installation. They hoped that Silvius would lower the bar for experimentation and innovation in this field, and encourage ordinary programmers to try coding by voice, instead of waiting until a crippling injury throws them in at the deep end. The idea was that the technique would become more handfull for the whole process of programming and controlling different devices.

VoiceCode is an Open Source initiative of Institute for Information Technology (IIT) within National Research Council Canada [1]. The task is to develop tools for compliant components to support current best practices in programming in voice using main mediums.

ShortTalk and EmacsListen is a development of a special spoken language for a human-computer interaction [2].

Voice Grip is an additional macros for Emacs editor that has been created to simplify the usage of commercial software for speech recognition by programmers [3].

Java by voice is a series of macros for Emacs editor that have been created to simplify the Java language code input [3].

Cache Pad is macros for Emacs editor for caching recent function names and variables for subsequent re-usage.

Emacs VR Mode is macros for editor that adding Select and Say [3].

Common coding in Microsoft Visual Studio, with the help of Speech Recognition in Windows, can become a good example.

Recently-developed program Code by voice can be offered as an example.

The steps of work are as follows:

- 1) Signal processing: finding features in sound signals;
- 2) Acoustic modeling: recognizing phonemes;
- 3) Language modeling: valid sequences of words [4].

And if the signal processing is easy, the other steps could not be the same. To recognize phonemes we need to train with hundreds of hours of speech. Learn individual phonemes, for example, by the GMMs or DNNs. And even with 24-core server with 48GB RAM you can lose several days to fulfill it and get some results.

To go through Step 3 we even need sometimes to model a new language.

The final work means dictating necessary language constructions by a programmer to a computer while the program is coding itself. The only inconvenience for a programmer will be the necessity to correct mistakes and to hand-type variables. The reason why typing becomes an inconvenience of the presented technique is evident – it will be necessary to dictate a simple variable “GetInfoFromMas” as “Get, no space, big letter, info, no space, big letter, from, no space, big letter, mas, no space,” as it is offered in one of the most well-known style of coding “CamelCase” [5].

Google has issued Android program “Voice Access” to use in smartphones and applications for them that can help handicapped people to use their smartphones. This fact proves the possibility to code by this program.

Acceleration of the modern world requires new approaches in a lot of spheres. Coding by voice is a tool that offers new opportunities for ordinary users of the software. The spheres of our life, where this technique can be used, are numerous. Of course, as it was mentioned above it will not be used in different secret services like state secrets, military, accounting, keeping some personal data and others. But the development of this technique is being waited by people who are involved in hand-typing and hand-coding of ordinary codes along with people who are handicapped or unable to type by any reason.

Coding by voice can make our routine life easier. And as a part of the whole evolutionary process of programming this stage is important. In this case we need to try to promote this technology by any means. We should not be afraid of all minuses and disadvantages which have been mentioned in this work above.

REFERENCES

1. Java Support To Voicecode, Adding. [Electronic resource] / Java Programming Using Voice Input. (2019). – Mode of access: https://www.researchgate.net/publication/265866290_Java_Programming_Using_Voice_Input. – Date of access: 19.02.2019.
2. ShortTalk and EmacsListen [Electronic resource] / SourceForge. – Mode of access: <http://shorttalk-emacs.sourceforge.net/>. – Date of access: 12.02.2019.
3. The features of GNU Emacs [Electronic resource] / GNU Emacs. – Mode of access: <https://www.gnu.org/software/emacs/>. – Date of access: 14.02.2018.
4. Spoken Language Processing [Electronic resource] / Indian Institute of Technology Guwahati. – Mode of access: http://www.iitg.ac.in/samudravijaya/tutorialSlides/OMFCC_DTW_HMM_GMM_LM_spkrReco_TTS.pdf. – Date of access: 19.02.2019.
5. CamelCase [Electronic resource] / CamelCase Definition // TechTerms. – Mode of access: <https://techterms.com/definition/camelcase>. – Date of access: 15.02.2019.