

Submission 762

Premium

Conference

**News**

EasyChair


## BotSE 2023 Submission 762

Submission information updates are disabled. If you need to update your paper, contact BotSE 2023 chairs.

For all questions related to processing your submission you should contact the conference organizers.  
[Click here to see information about this conference.](#)

All **reviews sent to you** can be found at the bottom of this page.

### Submission 762

Title	Idiolect: A Reconfigurable Voice Coding Assistant
Paper:	 (Jan 21, 06:09 GMT) <a href="#">(previous versions)</a>
Author keywords	speech recognition voice programming bots
Abstract	This paper presents Idiolect, an IDE plugin for voice coding and a novel approach to building bots that allows for users to define custom commands on-the-fly. Unlike traditional chatbots, Idiolect does not pretend to be an omniscient virtual assistant but rather a reconfigurable voice programming system that empowers users to create their own commands and actions dynamically, without rebuilding or restarting the application. We present a case study of integrating Idiolect with the IntelliJ Platform, illustrate some example use cases, and offer some lessons learned during the tool's development.
Submitted	Jan 20, 23:02 GMT
Last update	Jan 01, 00:00 GMT

### Authors

first name	last name	email	country	affiliation	Web page	corresponding?
Breandan	Considine	breandan.considine@mail.mcgill.ca	Canada	McGill University		✓
Nicholas	Albion	nalbion@yahoo.com	Australia	Independent Developer		✓
Xujie	Si	six@cs.toronto.edu	Canada	University of Toronto		

## Reviews

### Review 1

*Overall evaluation*

-2: (reject)

### Summary:

The paper presents an experience report of the building, redesign and evaluation of a voice coding assistant for the IntelliJ IDE. The tool, Idiolect can be used to issue voice commands to the IDE, which are then parsed and executed. The paper reports the experiences of the deployment of Idiolect, and lists some of the challenges associated with its adoption. Moreover, it reports an evaluation of the voice-to-text model, and it lists the plans for future work.

### Strong points:

- Well written paper that chronicles the design and deployment of a voice programming assistant.

### Weak points:

- The paper positions itself as a case study, while it is clearly not.
- Related work is not discussed in enough detail, nor is it clear how the work presented in this paper relates to bots in software engineering.
- Section VI is titled Evaluation, but the section does not contain enough details on the experiment that was conducted.

### Detailed comments:

- There is a vast body of work that covers the adoption and acceptance of general purpose voice assistants. Given the positioning of this work I believe it makes sense to include work like this, and to indicate how a voice coding assistant is similar to a technique like Siri.

- Similarly, the paper focuses mostly on voice programming assistants and does not discuss bots as such.

- The paper does not clearly position itself as an experience report, instead the abstract states that the paper reports on a case study. However, the paper is clearly an experience report as it chronicles the experiences of working with Idiolect, and it does not follow any established guidelines to report or execute case studies. For a case study I would expect documented and motivated RQs, and a clearly outlined procedure for data collection and data analysis.

- The paper could be further improved through the inclusion of an outline of the architecture of Idiolect. Currently it discusses each part in isolation, but the integration of all of the individual parts is not covered.

- Section VI is titled evaluation, it briefly reports on the process used to evaluate the voice to text models used. However, the first paragraph of the section is quite brief, and does not contain enough details to fully be able to replicate the process. For instance, how were the 100 utterances selected? What is the pre-defined command list?

- Section VI evaluates three different models, but it does not conclude with an actual conclusion on which model was most suitable.

- Section VI also reports on the number of downloads, however, it also mentioned that the bug reports etc. are used to evaluate Idiolect, but the paper does not continue and report on the actual number and types of bugs reported. Also the detail that 30% of the downloads originate from the People's Republic of China does not align with the rest of the paper. I would expect that if the paper is interested in the origins of the downloads that the paper reports a distribution and more details on the origin of downloads.

- The Threats to Validity section does not comment on any threats related to the evaluation of

the voice-to-text model. Given that the authors used the text-to-voice voices of OSX I would expect that there is a threat to internal validity as a result of this choice.

## Review 2

### Overall evaluation

1: (weak accept)

1. Paper summary

The paper presents Idiolect, a voice-enabled bot that enables developers to perform actions in the IDE using voice commands. Idiolect can be a useful tool for developers to access or trigger actions within the IDE without the need to traverse complex nested menus. The paper presents an overview of the process taken by Idiolect to convert the utterances by the developer into actions within the IDE. Idiolect is a newer version of a plugin developed in 2015 that leverages new voice recognition models and newer IDE versions.

The paper presents the results from a preliminary study on how well idiolect recognizes utterances when using different voice recognition models as well as historical data on the usage of the tool. The preliminary study involved the creation of synthetic voices to speak 100 utterances of the predetermined commands available in the tool. Unfortunately, the study results show very high WER, which could negatively impact the tool's usefulness.

Overall, the paper presents an intriguing application of bots in software development with the potential to benefit developers and stimulate relevant discussions. However, the paper's evaluation of Idiolect leaves room for improvement, and further research on the tool's effectiveness is needed to make it more useful to developers.

2. Points for the paper

- + Well-presented overview of the tool
- + Potential for discussion

3. Points against the paper

- Re-release of updated tool
- Lack of analysis on preliminary study

4. Supporting argumentation for your points

The strongest part of the paper is the introduction of idiolect. Although as outlined in the paper, Idiolect is not a new approach but rather a re-implementation from scratch from a plugin developed in 2015, the approach towards supporting voice-assisted support for developers' interactions with the IDE is a great use case for bots in software development.

The paper presents a clear and detailed overview of how Idiolect leverages VOSK (neural network language recognition models) and integrates it with IntelliJ IDEA to enable developers to trigger actions on the IDE using the intent described by the developer's utterance.

In addition to the potential discussions revolving around the use of bots for voice-assisted programming, the paper has the potential to promote discussion regarding the role of bots like idiolect as an accessibility tool for developers with disabilities.

On the other hand, the paper's evaluation of Idiolect is a significant weakness. The evaluation section is split into two parts, the first section presents the word error ratio (WER) of multiple

voice recognition models across different synthetic voices used for testing 100 utterances. However, the paper only presents the results in Figure 1 without any type of analysis of the results. The discussion of the results is crucial in any evaluation. Especially, since the WER is extremely high in most instances meaning that the only utterances in which the tool works are for those made by the voice labeled as Samantha.

Furthermore, the paper presents the results from a very small study test using synthetic voices, which is not acknowledged as a threat to validity in the threats to validity section. Synthetic speech and the commands being spoken and used by actual users of the system might differ substantially.

#### 5. Suggested paper improvements

The paper references the original plugin from 2015 and reports on the usage statistics and rationale regarding the stalling of downloads since the original version of the plugin was no longer maintained in 2017. However, the paper does not present a link to the original plugin website or paper. Having a link to the original plugin will make the paper more self-contained.

Similarly, a link to the current version of Idiolect in the IntelliJ marketplace would increase the discoverability of the plugin.

Future work on the tool would benefit by having the full evaluation mentioned in the future work section. In which human users are presented with the tool and their voice is used. Furthermore, if possible, the paper would benefit from presenting a side-by-side comparison of the 2015 plugin and Idiolect.

#### Review 3

##### Overall evaluation

2: (accept)

The paper describes an IDE-voice assistant that allows users to flexibly create new commands by voice.

I found the idea and paper interesting. Some of the write-up was a bit confusing. First of all, I did not realise until very late in the paper that the tool is 5 years old and was recently rewritten. You should explain this early in the paper. Similarly, the evaluation section is confusing. What version did you do the error rate analysis? What does WER really measure? Why are 5 year old download number interesting?

Yet, the idea is great and also to serve impaired developers with this sounds promising.