User Evaluation of InCoder Based on Statement Completion

01 Introduction

Thinking of a solution to a problem takes time, but then programming it will take even more time. Using state of the art models we can significantly reduce the time spent programming by autocompleting code. These models perform quite well on testing sets but do they perform as well when used for actual programming?

02 Methodology

This study focuses on evaluating the statement completion functionality of the InCoder model made by Facebook for Python [1]. A plugin was made that would suggest a statement completion to users in their IDE (Figure 1). This suggestion was compared to the final line of code after 30 seconds to evaluate performance. After enough usage users were also asked to fill in a survey.

```
def count_words(text):
    words = text.split()
    return
           len(words)
```

Figure 1: Example of plugin in PyCharm

	Total	Chosen
Occurrences	4164	663
Exact Match	21.95	62.14
Edit Similarity	52.73	83.03
BLEU-4	36.05	65.76
ROUGE-L	42.87	80.14

Figure 2: Evaluation of InCoder with user data

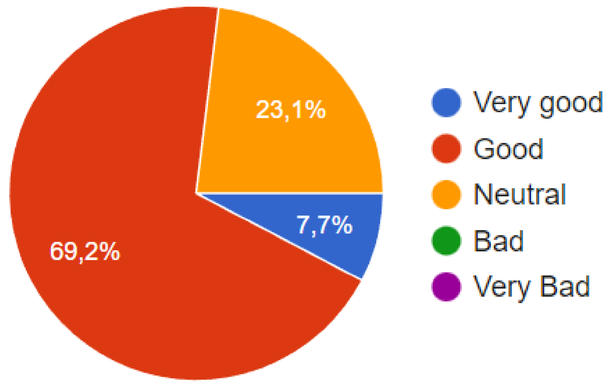

Figure 3: Perceived accuracy of users using plugin

03 Results

The results of the study are shown in the table (Figure 2) and pie chart (Figure 3). The "chosen" column indicates when a user explicitly selected the suggestion. The Exact Match shows perfect suggestions. Edit similarity uses single character edits. BLEU and ROUGE calculate the quality of the suggestion and are widely used in the natural language processing field [2,3].

04 Conclusion

The results show that the performance is a bit lower when evaluated in this setting but still quite good. Almost 1/4 suggestions are perfect and due to the high edit similarity part of the suggestion are also usable. If the suggestion is good users will use it and by looking at the pie chart we can conclude users like the functionality given and find it useful.

Marc Otten
m.j.c.otten@student.tudelft.nl
Supervisor: Maliheh Izadi
Advisor: Georgios Gousios
Professor: Arie van Deursen

[1] Daniel Fried et al., "InCoder: A Generative Model for Code Infilling and Synthesis", April 2022
[2] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation", July 2002
[3] Chin-Yew Lin, "ROUGE: A Package for Automatic Evaluation of Summaries", July 2004