

2023 年度 芝浦工業大学 工学部 情報工学科

卒 業 論 文

StandarML の開発環境向上のための関数の引数記述ミス
の補完

学籍番号 **AL20047**

氏 名 庄司 歩夢

指導教員 篠埜 功

目次

第1章	はじめに	1
1.1	背景	1
1.2	本論文の構成	1
第2章	関連研究	3
2.1	グループ1	3
2.2	グループ2	3
第3章	補完機能の実装概要	4
第4章	字句解析手法	5
第5章	構文解析手法	6
第6章	補完候補の取得	7
第7章	実装	8
第8章	考察	9
第9章	まとめと今後の課題	10
	謝辞	11
	参考文献	12

第1章 はじめに

1.1 背景

コード補完機能を利用することで、プログラム開発を効率的に行うことができる。コード補完とは、統合開発環境 (IDE) などのコードエディタから利用可能な支援機能で、入力途中のコードに対してその続きを入力する候補を自動表示する機能である [1]。この機能を利用することによって開発者は必要なコードをすべて直接記述する必要がなくなるため、開発の効率を向上させることができる。

しかし、すべてのプログラミング言語に対してこのような機能が十分に整っていないのが現状である。実際に、関数型言語の Standard ML[2] はパターンマッチングによって複雑な条件分岐の記述を簡潔に書くことができたり、コンパイル時に型が検査され安全性が高いなどの特徴がありながらも、補完機能が充実していない。

また、三浦 [3] はコード補完の有意性について調査し、補完機能を利用することで括弧の入力数が有意に下がり、補完機能の有用性を示した。よって本研究では Standard ML に対して、関数の引数の括弧の記述忘れの補完について統合開発環境 VScode[4] で実装していく。

1.2 本論文の構成

本論文の構成は次の通りである。まず2章において関連研究について述べる。3章において補完機能の実装概要についてのべる。4章において字句解析手法について

て述べる．5章において構文解析手法について述べる．6章において補完候補の取得について述べる．7章において実装方法について述べる．8章において考察を述べる．9章においてまとめと今後の課題を述べる．

第2章 関連研究

2.1 グループ1

初学者が環境構築にかかる手間をブラウザのみで利用できる Web IDE を利用することによって軽減し、プログラミング学習を進めるうえで、タイピングスキルによって影響が出ないようにするための手法として自動補完の実装が行われた [3]. 三浦は、効率的なプログラミングのために一般的に導入されている自動補完機能に着目し、初学者がどのように利用するかを調査した. その結果、自動補完を活用する学習者とそうでない学者に二分された. 利用頻度の高い補完候補には、for 文や if 文、画面描画関連の関数呼び出し、および void であった. for 文や if 文のスニペットには、括弧やカーリーブラケットを含んでいたため、補完機能を利用した編集行為の前後の文と、補完機能を利用しなかった編集行為の前ぼの文と比較すると、括弧やカーリーブラケットの入力数が有意に低かった.

2.2 グループ2

先行研究 [5] では、Standard ML のサブセット言語において error トークンを挿入することで構文誤りが存在しても補完が行われる実装が行われた. 実際に統合開発環境である VScode で Standard ML のサブセットに対して関数宣言による括弧の記述忘れと引数の記述忘れに対して変数名の補完が行われた. また、福本ら [6] は、error トークンの数と補完精度についての議論が行われた.

第3章 補完機能の実装概要

第4章 字句解析手法

字句解析器を生成するプログラムである Flex[7] というツールを用いて字句解析を行う。

第5章 構文解析手法

構文解析器を生成するパーサージェネレーターの1つである Bison[8] を用いて実装を行う。

第6章 補完候補の取得

Language Server で補完候補の計算を行う [9].

第7章 実装

第8章 考察

第9章 まとめと今後の課題

謝辭

参考文献

- [1] 中鉢欣秀. ソフトウェア開発演習のための仮想環境の構築と今後の展望. 産業技術大学院大学研究論文, Vol. 20, No. 13, pp. 93–98, 2019.
- [2] Robin Miler, Mads Tofte, Robert Harper, and David MacQueen. *The Definition of Standard ML (Revised)*. 1997.
- [3] 三浦元喜. 初学者向け processing プログラミング環境におけるコード補完機能の導入と実践. 教育システム情報学会誌, Vol. 37, No. 2, pp. 167–172, 2020.
- [4] Visual Studio Code - Code Editing. Redefined. <https://code.visualstudio.com/>.
- [5] 佐藤直輝. Standard ML に対する構文誤りを許すコード補完機能の実装. 芝浦工業大学情報工学科卒業論文, 2022 年度.
- [6] 福本大介, 平尾俊貴, 藤原賢二, 飯田元. 開発プロジェクトのドメインに特化した roberta によるコード補完モデルの提案. 電子情報通信学技報, Vol. 121, No. 416, pp. 49–53, 2022.
- [7] Flex - the GNU project. <http://gnu.ist.utl.pt/software/flex/flex.html>.
- [8] GNU Bison - The Yacc-compatible Parser Generator. <https://www.gnu.org/software/bison/manual/>.

[9] Language Server Protocol. <https://microsoft.github.io/language-server-protocol/>.