

工学部情報系分野の就職活動過程を考慮した 2 種の説明能力に関する文章集合の言語特徴比較

上野 未貴

井佐原 均

豊橋技術科学大学 情報メディア基盤センター

{ueno, isahara}@imc.tut.ac.jp

1 はじめに

就職活動において、学生が他者に情報を伝える能力は非常に重視されており、人工知能分野では、グループ会話におけるコミュニケーションスキル能力の推定 [1] などの研究がある。この研究では、グループディスカッションを想定したマルチモーダル情報の中で、コミュニケーション能力の推定に有効な特徴をサポートベクターマシンにより精査している。

近年、社会的に学生のコミュニケーション能力や文章作成能力の低下が叫ばれており、各大学や企業が学生の就職活動時の説明能力向上のためのセミナー開催や資料作成、添削、面談により、能力向上を目指すという現状がある。そのため、早く適切に学生に指導して能力を向上させること、大学や企業内で学生の説明能力向上に関わる職員の負荷を下げる事が望まれている。特に、著者らの研究室や授業に関わる、エンジニアとして就職する情報系の学生は就職活動に際して、企業のエンジニアにソフトウェアの実装方法に関わる詳細な説明、人事にソフトウェアの概念的な説明をするという 2 種の能力を問われることが多い。

計算機的処理により、日本語文添削 [2] や就職活動に特化した文の添削をする手法も研究されているが、情報系分野の就職活動の過程を考慮した自動添削手法に関する研究は十分でない。

そこで、本研究では、情報系分野における就職活動時に問われる 2 種の能力向上のための文章自動添削などの教育応用に向けて、実際に学生が上記を意識して書いた 2 種の文章集合の言語特徴に見られる特徴を比較して考察することを目的とする。

2 情報系の就職活動過程と 2 種の説明能力

工学分野の情報系の就職活動においては、エンジニアへの説明と人事への説明という過程を経るのが普通である。なお、ここで、エンジニアへの説明、人事への説明という 2 種に大別するのは、ソースコードの作成経験の別によるものである¹。

エンジニア面接での説明能力 一般的に情報分野のエンジニア面接では、ソースコードが読めるエンジニアが、学生のプログラミングにおけるソースコードの作成能力、すなわち、ソフトウェアの設計に関する能力、具体的なプログラミング言語の記述能力などを問う。

人事面接での説明能力 一方で、人事への説明では、具体的なソースコード作成経験がない可能性を有する面接官が、ソフトウェア設計に関する概念的理解能力、ソフトウェアの実応用に関する着想能力や他者への説明のわかりやすさなどを問うことが多い。

近年、情報系でエンジニアとして、特に新卒で就職を希望する過程は、一般的な面接だけでなく、以下のように多様化している。

- Git [3] などのバージョン管理システムのリモートリポジトリ GitHub 上へのソースコード作成過程を提出させる
- 対面あるいは遠隔のインターネット上でソースコードをリアルタイムに書かせる
- 短期あるいは長期のインターンで直接人物像と能力を判断する

¹エンジニアとしてソースコードの作成経験がある人事や、人事経験のあるソースコードを書くエンジニアもいるが、説明を簡易にするため、この 2 種に大別している。

しかし、そのいずれの過程においても、就職を希望する学生は、上記で大別した 2 種の異なる視点から、他者に文や図、口頭で説明する能力を問われる。

3 実験

3.1 実験前準備と実験環境

3.1.1 授業

著者らが担当する、豊橋技術科学大学の 2016 年度後期の「ソフトウェア演習 IV」では、ソフトウェアの設計に関する幅広い知識を得て、研究過程や就職活動でも役立つよう、以下の方針で授業を設計している。

- 自ら作りたいソフトウェアの機能を考え、プログラミングして実装する
- オブジェクト指向を理解した上で、ソフトウェアの上流設計をする
- アクセス修飾子や static フィールドなどの理解をする
- UML や概念図を用いた図式化した設計の整理をして説明する
- Git を用いたバージョン管理に基づくチーム演習により、複数人でソフトウェアが開発できるようになる

3.1.2 教育応用に向けた現場での課題

近年、プログラミング教育を取り巻く状況は急速に変化し、大学や企業で教える内容の特色差の強調および差を吸収することの両方が望まれている。本授業では、各履修者がプログラミングしてソースコードを書いた上で、説明文、コード抜粋、UML もしくは概念図を用いた説明スライドを作成し、エンジニア向け、人事向けを意識した、説明文を作成する課題を課した。

3.1.3 実世界を想定した課題

著者らの大学では、学生証に IC チップを搭載しており、学内の IC カードへのチャージ機器で学生証を挿入し、現金を投入してチャージし、学生証を学内のコンビニで非接触の IC カードリーダーにかざすことで、商品の決済が可能なシステムを導入している。また、この IC チップ搭載の学生証を、本学の教室に備え付

けた出席管理リーダーにかざすことで、教務システムと連携して、各学生の出欠の管理をし、Web ブラウザ上で閲覧することが可能である。

上述のような学生にとって身近な機器の中身を想像する課題を 3.2.1 で設定した。

3.2 実験対象データの作成と実験手順

3.2.1 課題内容

プログラミング言語として Java を用いて、授業受講学生に、第一著者が作成した以下の課題に取り組みさせた。

課題

お金を入れると、学生証の学籍番号と学生名と残高を表示・更新する、IC チャージ機を考えよ。

1. まず、必要と考えられるクラス、変数とメソッドを考えよ。
2. 以下の 2 つのクラスを作成せよ：
 - (a) 学生証を表す StudentCard クラス
 - インスタンス変数：学籍番号、学生名、残高
 - クラス変数:存在する学生証のリスト²
 - (b) コンビニのレジを表す MainShopCharger クラス
 - インスタンス変数: 挿入されている学生証 insertedStudentCard
 - メソッド:
 - insertStudentCard 学生証の挿入
 - chargeMoney 残高への追加
 - main

実行確認方法の例

MainShopCharger クラスの main メソッド内で、学生証のインスタンス StudentCard1 と StudentCard2 を 2 つ作成し、insertStudentCard で学生証を挿入する処理をし、charge メソッドで、適当な金額を追加して残高を更新した後、printAccountBalance を呼んで、残高をコンソールに表示する。

²簡易的にデータベースを表している。StudentCard クラスのインスタンスを、StudentCard クラスのコンストラクタ内で本リストに add して、インスタンスを格納するための静的フィールドを意識している。

さらに、参考資料を与えて、以下のように機能追加と修正をさせ、Git を用いて開発させた。

1. チャージの際に ICCharger 機器（オブジェクト）の最新チャージ年月日を格納し、最新チャージ年月日を表示可能なようにコードを修正せよ。
※ヒント：日時の取得は Java の Calender クラスを使う。
2. ICCharger 機器の最新チャージ年月日だけではなく、最新 5 回分のチャージ履歴を表示可能なようにコードを修正せよ。
※ヒント：
 - （機器チャージ）履歴 id, ユーザ id, 日時, 残高
 - (HashMap) と List を使う
3. StudentCard クラスと MainShopCharger クラスという 2 クラスではなく、今後のこの機器の拡張性を考えて、より良い設計を考えて実装せよ。ただし、クラス数は 2 以上 5 以下とする。
※拡張性のヒント：同じ機器の台数（インスタンス）を増やす、機能を追加した、上位機種を作る
4. コーディング規約を調べて、適切にフィールド名、メソッド名を修正せよ。

3.2.2 課題の説明内容

前節で作成した、原則として 2 以上 5 以下のクラスの概要に関する以下の 2 種の説明文を各 200 文字以内で作成させた。

1. エンジニア向け：ソースコードが読める人（企業ではエンジニア、身近なところではこのクラスの他者）向けの説明
2. 人事向け：ソースコードがわからない人（企業ではエンジニア経験のない人事、身近なところでは小学生³や情報系に特に関連のない知人）向けの説明

3.2.3 実験手順

以下の手順で実験を実施し、同意が得られた被験者のデータを分析した。

³年齢による文の平易さも考慮する必要がある。

表 1: 26 名の 2 種の文対から得た言い換え表現

エンジニア向け	人事向け
JAVA	プログラミング
プログラム	作成物
クラス	部品
コンソール	画面
格納	記録
サーバー	-(語句削除)
メソッド	～といった機能

1. 20 代の工学系情報系分野の学生 26 名に 3.2.1 の課題を与え、3.2.2 に沿って、2 種の各 200 文字以内の文章を自由に作成させ moodle [4] へ提出させた。
2. 得られた 2 種 × 26 名のテキスト集合の言語特徴を比較して考察した。

3.3 実験結果

表 1 に 26 名の結果から、2 種の文章対を比較して得た言い換え表現を示す。

3.4 実験考察

26 名の結果から、2 種の文章集合を比較し、各文章群に以下のような傾向が見られた。

エンジニア向け

- 具体的なソフトウェアのクラス名⁴や変数名が頻出。
- チャージ履歴の設計に関する ArrayList や Map など Java の Collection 型についての説明が頻出。

人事向け

- コンピュータ用語を省く、言い換える。
- 変数名の説明は見られない。
- 機能に関する説明が頻出。

⁴なお、具体的なクラス名は被験者によって異なる。

4 まとめと今後の課題

本研究では、情報系分野の就職活動で問われる2種の説明能力に着目し、言語特徴の比較実験をした。被験者から得た、2種の文集合間でコンピュータ用語の明らかな言い換えや省略という差が見られた。今後の課題は以下の通りである。

- 定量的な実験結果について詳細に考察する。
- 企業内のエンジニア、人事に他者の理解度に応じた説明時に着目する箇所を尋ね、被験者から得られた特徴を評価する。
- ソフトウェア設計に関する説明文の自動添削に向けて、クラスを定義し、コーパスを作成する。

謝辞

著者らの大学の授業「2016年度後期ソフトウェア演習IV」に関するRAとして授業内容を確認、補佐し、特に細かい設計やGitに関するバージョン管理方法を確認してくれた森下真敬くんに謝意を示す。同授業のTAとして、各回の授業内容を確認、補佐してくれた、斎藤研太くん、中村尚道くん、早川大貴くんに謝意を示す。本研究の趣旨を理解して、説明文を提供することに同意してくれた「2016年度ソフトウェア演習IV」の受講者の一部である26名の学生に謝意を示す。

参考文献

- [1] 岡田 将吾, 松儀 良広, 中野 有紀子, 林 佑樹, 黄 宏軒, 高瀬裕, 新田 克己, “マルチモーダル情報に基づくグループ会話におけるコミュニケーション能力の推定”, 人工知能学会論文誌, vol 31.(6), AI30-E-1-12, (2016)
- [2] 池原 悟, 小原 永, 高木 伸一郎, “自然言語処理技術の応用: 2. 文書校正支援システムにおける自然言語処理”, 情報処理, vol.34(10), (1993)
- [3] Git: <https://git-scm.com/>
- [4] M. Dougiamas and P. Taylor, “Moodle: Using learning communities to create an open source course management system”, pp.171-178, (2003)