

記述式小テスト支援システム

— 学生の理解状況把握のための情報抽出法 —

高瀬 治彦 *1・川中 普晴 *1・鶴岡 信治 *2

Email: takase@elec.mie-u.ac.jp

*1: 三重大学大学院工学研究科

*2: 三重大学大学院地域イノベーション学研究科

© Key Words 小テスト, テキストマイニング, 自然言語処理

1. はじめに

講義において学生の理解を深めるためには、講師が一方的に講義内容を教授するのみではなく、学生の状況を把握しそれに応じた授業改善（補足等）を行うとよい。しかし、大学の講義では学生の反応が乏しくさらに学生の数も多いため、効果的な授業改善が行えない。西森は、大学での数学基礎教育に関する調査⁽¹⁾を行い、これらの問題を解決する方法として小テストを多くの講師が利用していることを示した。しかし、小テストはその実施および学生へのフィードバックに少なからぬ手間がかかるという意見も示している。

近年の計算機およびそのネットワークの発達をうけて、さまざまな小テスト支援システムが構築され、講師の手間が軽減されている。例えば、Moodle⁽²⁾などのe-Learning システムでは、学生は Web を使用して解答することができ、その内容に応じてシステムがコメントを返すことができる。また、クリッカーと呼ばれる装置を用いることで、多肢選択問題に対する学生の解答を講師が即時に得ることが可能である⁽³⁾。しかし、これらのシステムが有効に働くのは主に多肢選択問題である。この種の問題は、慎重に選択肢を作成しないと、学生の理解状況を正確に把握することができない。そのため、講義の流れに応じて小テストを実施するには難がある。

そこで我々は、学生の理解状況を把握するために行われる記述式の小テストに着目する。以下、このような簡単な演習を、単に「小テスト」と呼ぶ。記述式の解答を課すことで、学生は自身の言葉で解答を記述するため、その理解状況が解答に現れる。しかし、それらを把握するためには解答を読む必要があり、講師の負担は増大する。また、計算機で支援するとしても、各解答の内容を正確にとらえることは困難であり、十分な支援がされていない。これをふまえて、我々は小テストを支援する計算機システムの構築をめざしている。

このような計算機システムの一環として、本稿では講師が小テストの解答群の内容を把握しやすくなるようなインターフェイスについて検討する。加えて、その実現に必要な、キーワードの自動抽出法についても検討する。具体的には、(1) 多数の学生が犯している誤りを早

期に発見できるように鍵となる情報を提供する、(2) 小テストの解答の特性に合った分析を行う、(3) 特別な準備を必要としないの3点を満たす方法について検討する。これにより、講師は少ない手間で学生の理解状況をすばやく把握できるようになる。

2. 記述式解答群の分析方法

この節では、レポート等も含めた解答群の分析・表示法について、これまでに提案されたものからいくつか紹介する。その後、学生の理解状況を把握するためのツールとして小テストの解答群の分析・表示法について必要なこと、従来手法が不足している点について議論する。

2.1 従来の分析手法

小テストの解答に限らなければ、講義で学生により記述される文書の分析方法が、これまでに数多く提案されてきた。これらは解答を自動的に採点するものと、解答群を教師にわかりやすく提示するものに大きく分けることができる。前者の例として、石岡らにより提案された JESS がある⁽⁴⁾。JESS は、日本語の小論文を3つの特徴量（修辞、論理構成、内容）に基づいて、10点満点で採点する。このシステムにより1,000字程度の小論文を用いて評価したところ、良好な結果が得られた。後者の例として、椿本らによる円錐型レポート採点支援マップがある⁽⁵⁾。これは、レポートをDマップと呼ばれる可視化手法により解答群を表示するものである。表示は使用単語と文章量に応じて行われる。この結果、未採点のレポートと採点済みのレポートとの位置関係を講師が把握しやすくなるため、採点の助けとなる。この効果は、300字程度の語句説明をテーマとしたレポートを用いた実験により確認されている。

2.2 システムの要件

小テストの解答群は、一般の文書群とも、従来システムが対象としている小論文、レポートとも、異なる性質を持つ。これをふまえて、講師を支援するシステムに対する要求事項を検討する。

第一に、講師が主要な誤りを早期に発見できるよう

に、その鍵となる情報を提供することが必要である。これは、学生の理解状況を把握するという目的のためには、各解答の採点結果だけでは十分ではないためである。ここで、主要な誤りに限定しているのは、授業中に行う授業改善としては、個別指導は現実的でなく一斉指導によるものが主要なものになると考えたためである。

第二に、システムは小テストの解答群の特性に応じた分析をしなければならない。小テストの解答群の特徴として以下の2点がある。まず、各解答の文章量が少ないことである。講師は小テストの結果をもとに授業改善を行うため、小テスト自体に長い時間をかけることは避けたい。そのため、解答に必要な文字数を多くするべきではない。本研究では、多くても3文(200字)程度の文章量を想定する。もう一つは、各解答が類似しやすいことである。小テストとして理解状況を把握するための質問をするため、多くの場合、その解答は講義で説明した内容となる。そのため、解答に使用される語彙は限定的なものとなり、さらに文章表現も類似しやすくなる。システムはこれらの場合に対応できる必要がある。

第三に、システムを使用する際に、事前に特別な準備を必要としないことが望ましい。これは、事前に問題を登録したり、問題以外にも必要な情報を登録したりすることは、気軽に小テストを実施することの妨げとなるためである。

前節で取り上げた従来システムは、想定しているよりも多い文字数の文書を対象としている。またいずれの方法も、揺らぎなく採点することをめざしているため、必ずしも主要な誤りを把握するための情報を提供しない。これらの点について検討の余地がある。

3. 提案する解答群の提示法

前節での議論をふまえて、我々はこれまでに、未提出の解答もふくめたすべての解答を講師に逐次提供するシステムを提案した⁽⁶⁾。このシステムが講師に提供する画面の例を、図1に示す。この一覧により、講師は学生が入力した解答を逐次閲覧できる。また、操作により類似した解答群を抽出することなどもできる。このシステムを使用することで、講師は特別な準備なしに、解答提出以前から解答群の内容を把握することができ、小テスト終了と同時に授業改善を行うことが可能になる。しかし、講師は逐次更新される多数の解答を読まなければならない、その負担が大きかった。そこで講師が解答を逐次把握することを可能にしたまま、その負担を減らす解答群の提示方法を検討する。

検討に先立ち、筆者らの所属する三重大学工学部電気電子工学科において、小テストを多用する教員数名に対して、「小テストの解答を採点する際の手順を教えてください」という質問を行った。その結果、ほとんどの教員が以下の手順をふんでいることが分かった。まず使用

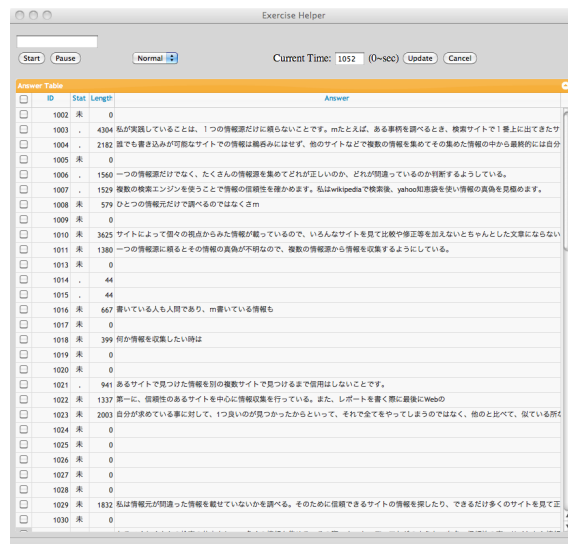


図1 従来システムによる解答提供画面

しているキーワードをチェックし、次にキーワードの使用法をチェックする。最後に全体の構成を確認する。この結果は、解答全文を講師に最初から提供する必要がないことを示唆している。

そこで、キーワード、フレーズ、解答全文の順に解答群を提示するインターフェイスを提案する。ここでキーワードとは、学生の理解状況を把握するための鍵となる単語を指すものとする。また、フレーズとは解答からキーワードに関連する部分を抽出したものとする。システムは、解答群から自動で抽出したキーワードのリストをまず表示する。講師は、その中からキーワードを選択することで、その使用法がわかるフレーズを得る。さらに必要に応じて、フレーズを選択することで、それを含む解答全文を得る。キーワード、フレーズについては、その使用頻度も併せて表示することで、講師が主要なものを把握する助けとする。キーワードを自動抽出するのは、講師がこれをあらかじめ登録する手間を省くためである。また自動で抽出することで、講師が想定していなかった語を学生が使用した場合にも、その存在を知ることができる。フレーズは、指定したキーワードで始まる(終わる)数単語分の文字列とする。これは、解答を逐次分析することを考えると、計算量の観点から、複雑な自然言語処理を行うことは現実的ではないと考えたためである。提案法では、キーワードを検索しその前後を表示するだけなので、計算量の観点からは問題ない。また、フレーズの長さが一定の場合、キーワードをフレーズの端に配置することで、キーワード以外の部分の意味をとりやすくなる。

4. 提案する解答群の分析法

前節で提案した表示を行うためには、キーワードを少ない計算量で自動的に抽出する必要がある。キーワードとして解答を記述するうえで重要な単語を抽出すれば、

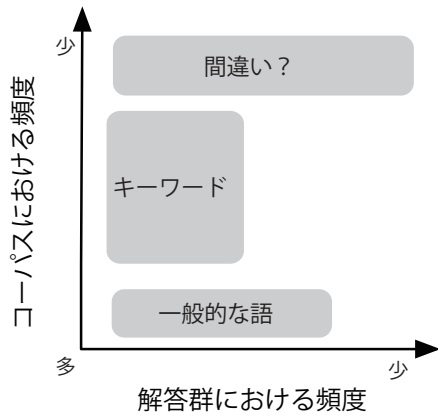


図2 単語の頻度情報からのキーワードの推定

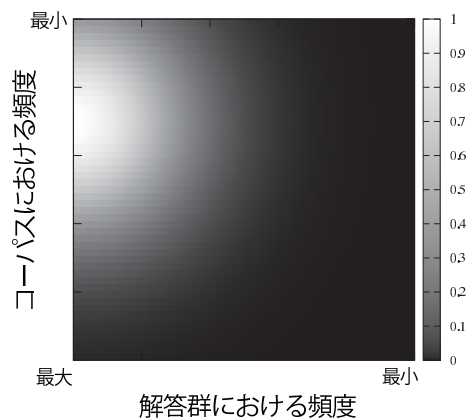


図3 各単語の重要度

講師はそれを含むフレーズから学生の理解状況を把握できるだろう。

ここで、各単語の使用頻度に着目する。解答を記述するうえで重要な単語は、多くの解答中に現れる。また、一般的に用いられる語は、一般の文書群にも多く現れる。そこで、全解答中での各単語の頻度と、一般的な内容の多量の文書(コーパス)中での頻度を用いて、キーワードとして抽出することを提案する。具体的には、各単語の頻度をもとに、その単語の種類を図2に示すように判断する。すなわち、解答群中では頻度が高いが、コーパス内ではそれほどではない単語をキーワードとみなす。実際には、各単語の頻度情報から図3のような関数を用いることで各単語の重要度を求め、上位の単語をキーワードとみなす。ただし、コーパス中での頻度は単語により大きく異なるため、頻度そのものではなく、対象となる単語中の頻度による順位を用いて重要度を求める。なお、キーワード単体で見たときに意味をとらえやすいように、名詞、動詞、形容詞のみを対象とする。

5. システムの実装

提案した方法を、講師用の端末としてタブレット端末(Apple社 iPad)を使用し実装した。学生の解答の収集・分析は別途用意したサーバで行い、講師用端末で分

析結果を表示する。サーバは、Webサーバと共同して動作するモジュールとして、プログラミング言語 perl, PHP を用いて実装した。日本語文から単語(形態素)を切り出すためのソフトウェアには MeCab⁽⁷⁾ を使用した。以下の結果は、コーパス内の各単語の頻度として、Webより収集された文書(200億文, 2550億単語)から計数したもの⁽⁸⁾を用いたものである。講師側の端末用のソフトウェアは、Objective-Cを用いて実装した。

以下に、システムを用いて実際の解答を閲覧する様子を示す。使用した解答は、2007年度に三重大学工学部電気電子工学科1年生向けに開講された講義「計算機基礎I及び演習」中で行われた小テストのものである。設問は「機械語について説明せよ。ただし、『機械語は』で文章を始めること。」であり、解答数は79、平均文字数は78字であった。解答時間は15分だったが、開始後7分の時点での解答を分析した結果を以下に示す。この時点での解答は平均56.7字(提出者数20名)であった。なお、解答例は「機械語は、数値列により表現された、CPUが直接実行可能なプログラムを記述するための言語である。」である。

最初に、システムはこの時点の全解答からキーワードを抽出し、その頻度とともに提示する(図4)。講師は、これを閲覧し注目した単語を選択する。ここでは、「言語」を選択したとする。次に、システムは選択されたキーワードで始まる(終わる)フレーズを抽出し、その頻度とともに提示する(図5)。なお、抽出するフレーズはキーワードを含めて6形態素とし、かつ、文の先頭・末尾をまたがないようにした。講師は、これを閲覧し注目したフレーズを選択する。ここでは、「言語のことである。」を選択したとする。最後に、システムは選択されたフレーズを含む解答群の全文を表示する(図6)。

ここで、提案したシステムが2.2節で示した三つの要件を満たしているのかを検討する。第一の要件については、講師が解答を採点する際の手順に従いキーワード、フレーズ、全文の順に段階的に解答を提供しているので、解答を把握するのに不足はないと考える。加えて、キーワード、フレーズについては頻度も併せて提示しているため、講師は主要なものを判断しやすい。第二の要件については、この時点で、最終的な解答と比べ文字数にして73%、提出者数にして25%の解答群から、この間に解答するために重要な単語「命令」、「数字」、「実行」を重要度で上位の10単語(表1)中に含むように、キーワードを抽出した。このように学生が解答途中であっても、提案システムはその時点の解答群を分析し必要な情報を提供できる。第三の要件については、システムは学生の解答群のみを使用して分析を行うため、講師は問を登録する必要すらない。以上より、提案するシステムは三つの要件を満たしていると言える。

表 1 自動抽出されたキーワード (上位 10 個)

順位	キーワード	順位	キーワード
1	命令	6	機械
2	言語	7	コンピューター
3	コンピュータ	8	プログラム
4	プログラミング	9	数字
5	実行	10	表さ

6. おわりに

本稿では、講義中に気軽に記述式小テストを実施できるように支援する計算機システム構築の一環として、講師が解答群の内容を把握しやすくなるようなインターフェイスを提案した。これは、講師にキーワード、フレーズ、全文の順に解答を提示するものである。加えて、これに必要な、解答からのキーワードの自動抽出法も提案した。この方法は、解答中の各単語についてその頻度情報から重要度を求め、重要度が高いものをキーワードと見なすものである。

このシステムにより、小テストの解答群から学生の誤りを発見し授業改善を素早く行うことを、講師が少ない労力でできるようにすると期待される。

謝辞

本研究は、文部科学省科学研究費助成金（基盤研究 (C) No. 23501106）からの補助を受けた。

参考文献

- (1) 西森敏之：“大学生の授業における態度と数学教師の対策—日本数学会のある調査より—”，高等教育ジャーナル—高等教育と生涯学習—，vol. 6, pp. 1-31, 1999
- (2) Moodle.org: “open-source community-based tools for learning,” <http://moodle.org>, 2012 取得
- (3) Jane E. Caldwell: “Clickers in the Large Classroom: Current Research and Best-Practice Tips,” CBE-life Sciences Education, vol. 6, no. 1, pp. 9-20, 2007
- (4) 石岡恒憲：“記述式テストにおける自動採点システムの最新動向”，行動計量学, vol. 31, no. 2, pp. 67-87, 2004
- (5) 椿本弥生, 柳沢昌義, 赤堀侃司：“レポート内容とその評価を可視化する円錐形レポート採点支援マップの開発と評価”，日本教育工学会論文誌, vol. 31, no. 3, pp. 317-326, 2007
- (6) Taisuke Nakamura, Takuma Imai, Haruhiko Takase, Naoki Morita, Hiroharu Kawanaka, Shinji Tsuruoka: “e-Learning System to Support Teacher’s Awareness for Misunderstanding in Quiz,” Proceedings of the International Joint Conference on Neural Networks 2010, pp. 1704-1708, 2010
- (7) Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto, “Applying Conditional Random Fields to Japanese Morphological Analysis,” Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp. 230-237, 2004
- (8) 工藤拓, 賀沢秀人, Web 日本語 N グラム第 1 版, 言語資源協会, 2007

キーワード	頻度
命令	47
言語	96
コンピュータ	29
プログラミング	20
実行	64
機械	112
コンピューター	10
プログラム	25
数字	9
表さ	8
用い	7
低級	10
直接	71
表す	4
マシン	4
書か	15

図 4 提案システムの表示: キーワード

フレーズ	頻度
が直接実行できるプログラミング言語	4
理解し直接実行できる言語	2
機械語はプログラミング言語	2
実行することができる言語	2
命令することのできる言語	2
言語である。	29
言語のことである。	3
言語であり人間が	2
言語。	2
言語であり二進数で	2
言語のことでの言語	2
言語	2

図 5 提案システムの表示: フレーズ

134 機械語は、0と1のみで構成される2進数で表される言語のことである。→られたプログラムによる命令は、コンピュータが直接実行可能である。

138 機械語は、CPUが直接実行できるプログラミング言語のことである。→語で書かれた全てのプログラムは、最終的にこの機械語に翻訳される。

119 機械語は、マシン語とも呼ばれCPUが直接実行することの出来る二進数の電気信号でできているプログラミング言語のことである。

図 6 提案システムの表示: 全文