英作文統合支援環境 *phloat*

林部 祐太*
yuta-h@is.naist.jp
奈良先端科学技術大学院大学

萩原 正人
{masato.hagiwara, satoshi.b.sekine}@mail.rakuten.com
楽天技術研究所 New York

関根 聡

1 はじめに

ビジネスやアカデミックなどの，世界的なコミュニケーションの場において，英語は Lingua franca としての地位を確立している．そのため，世界中で多くの人が英語を学んでいる (English-as-a-second-language; ESL)．

ESL 学習者にとっての英作文は，語彙・文法・フレーズ・前置詞や冠詞などの様々な要素を適切に組み合わせなければならいため，難しい課題の 1 つである．そこで，スペルチェッカ・文法チェッカ・電子辞書・フレーズ検索システム等，ESL 学習者の作文を支援する様々なシステムが提案されてきた．しかしながら，実際に ESL 学習者が英作文するときには，それらの複数のツールを組み合わせて使わなければならないため，スムーズな英作文ができない．また，既存の文法チェッカは，誤りを含んだ ESL 学習者の文章を入力としているが，誤った文法を含む文の解析や書き手の意図を推測する必要が生じ，必ずしも高い精度であるとは言えない．そこで本稿では，フレーズを英作文中にサジェストすることで，ESL 学習者が（特に意味的な）誤りを犯すのを事前に防ぐ英作文統合支援環境，*phloat* (PHrase LOokup Assistant Tool) を提案する．

2 関連研究

2.1 自動作文添削システム

Microsoft Word[1] の最近のバージョンでは，スペル誤りや subject-verb agreement 等の単純な文法誤りの検出をリアルタイムに自動的に検出する機能があるが，大半のシステムは作文後に添削する．

ESL assistant [Leacock 09] は ESL 学習者が犯しやすい誤りに焦点をあてたウェブベースの英作文補助ツールである．誤りと思われる箇所に対して，ユーザが書いたオリジナルの表現と，システムが提案するより良い表現の 2 つでウェブ検索の結果を提示することで，どちらの表現が適切であるかの判断の支援する．Criterion [Burstein 04] は ESL 学習者の英文の質を自動的に評価する教育用ウェブシステムで，文体や文法についてのフィードバックも表示する．

他にも Grammarly[2], WhiteSmoke[3], Ginger[4] などの，多くの作文添削システムがある．

2.2 英語 Input Method Editor (IME)

AI-type[5] は，単語の部分マッチによって英単語の入力を補助する英語 IME である．さらに，単語 n-gram による単語のサジェストも行う．

PENS [Liu 00] や FLOW [Chen 12] は中国語母語話者のための英語 IME で，ピンイン（ラテン文字化された中国語）での入力に対し，英語の翻訳を提示する．FLOW には，ユーザが選択した英語のフレーズに対して，言い換え候補を提示する機能もある．中国語 IME である Google Pinyin IME[6]は，英語 IME も含んでおり，中国語による英語辞書の検索・同義語の推薦等ができ，スペルミスを含む入力にも対応できる．

英文名文メイキング [Doi 98] は日本語 IME と連携して英作文するシステムである．和英辞書の検索機能・例文の検索機能・日本語文から英語表現に変換する機能などをもつ．

AceWiki [Kuhn 08] は Attempto Controlled English (ACE) で編集する wiki システムである．入力する英語の文法に制限を加えることで，文法誤りを含まない英作文を可能にする．編集画面では，単語を選択すると自動的にそれに続けられる単語の候補を提示していく．

2.3 フレーズ検索システム

フレーズ検索システムは英語の翻訳や作文において有用である．[Kato 08, Wible 10, Takamatsu 12] などは，“there is a tendency for [noun] to [verb]” といった英語でのパターン検索が可能である．しかしながら，IME とは統合されておらず，フレーズ検索画面と作文画面とを切り替える必要があり，スムーズな英作文ができない．また，パターンを予めユーザが知っていなければ検索で

*本研究は楽天技研 NY でのインターンシップ中に行った
[1]http://office.microsoft.com/en-us/word/
[2]http://www.grammarly.com
[3]http://www.whitesmoke.com/
[4]http://www.getginger.jp/
[5]http://aitype.com
[6]http://www.google.com/intl/zh-CN/ime/english

図 1: “okuru” に対する反応

図 2: “okuru” のスロットに対する反応

図 3: “okur” に対する反応

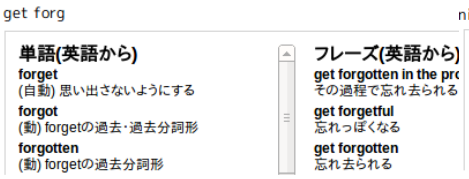

図 4: “get forg” に対する反応

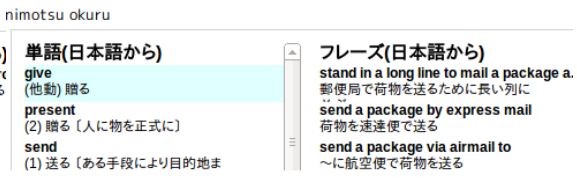

図 5: “nimotsu okuru” に対する反応

きなかったり，母語による検索が不可能であったりと，実用に際しては改善の余地がある．

2.4 翻訳支援システム

機械翻訳技術を用いて，翻訳を支援するシステムも提案されている．例えば，TransType2[Esteban 04] や TransAhead [Huang 12] は，翻訳元の原文に基づいて，翻訳候補の単語の自動サジェストするシステムである．TWP (Translation Word Processor) [Muraki 94, Yamabana 97] は，インクリメンタルかつインタラクティブに翻訳先の候補を提示するシステムである．

3 提案システム

2 章で様々な ESL 学習者のための入力支援システムを挙げたが，

- 辞書引きやフレーズ検索を行うには，画面の切り替えが必要であり，シームレスに作文できない
- 検索結果がリアルタイムに表示されない
- 既存の英語 IME でサジェストされる候補は単語のみである

といった問題点がある．本稿ではフレーズを作文中に IME のようにサジェストする英作文統合支援環境，*phloat* (PHrase LOokup Assistant Tool) を提案する．

3.1 システムの概要

phloat はテキストエディタに組み込まれて動作し，ユーザが文字をタイプするごとに前後の文字列をクエリとして英語語句データベースの検索を行い，その検索結果をもとにリアルタイムにユーザへサジェストする．ユーザが入力する文字列は英語であってもローマ字化された日本語であっても構わない．

図 1 はユーザが “okuru” と入力したときの画面である．最左カラムには，英単語のサジェスト候補が表示され，その右側にはフレーズのサジェスト候補が “okuru” の語義ごとにクラスタリングされて[7]表示されている．もしユーザが「E メールを送る」と書きたいのであれば，一致するフレーズ（2 列 2 行目）をクリックする（またはキーボードの矢印キーで選択して Enter キーを押す）と，“okuru” が “email the address of” へ自動的に置換される．このフレーズは適当な何らかの語句を埋める必要のある箇所（以下、スロットと呼ぶ）を示す「〜」を含んでおり，適当な何らかの語句を埋める必要があることを示している．システムは図 2 のように自動的にスロットに入りうる語句の候補を提示する．

このように，サジェスト候補中にユーザが入力したい表現があれば，それを選択するだけで意図する適切な単語やフレーズを入力することができる．もし，ユーザが望む表現が見つけられなければ，続けて文字をタイプするごとに新しいクエリで再度データベースの検索が行われ，異なるサジェストを得られる．

phloat は “okur” や “get forg” のように部分文字列に対しても候補を提示する（図 3,4）．“nimotsu okuru”, “nimotsuwookuru” といった複数の語の組み合わせに対しても検索する（図 5）．

3.2 システムの実装

システムの全体像を図 6 に示す．システムは，(A) 単語・フレーズのサジェスト，(B) フレーズをクラスタリングしてサジェスト，(C) スロットのサジェスト，の 3 種類のサジェストを行う．通常は (A) のサジェストを行い，フレーズを日本語から検索する際にそれが動詞であれば，(B) のサジェストを行う．(A),(B) でユーザが選択したフレーズにスロットが含まれている場合，自動的

[7] 1 列の表示ではその中から適切なものを選ぶのが難しいため，クラスタごとの表示を行う．

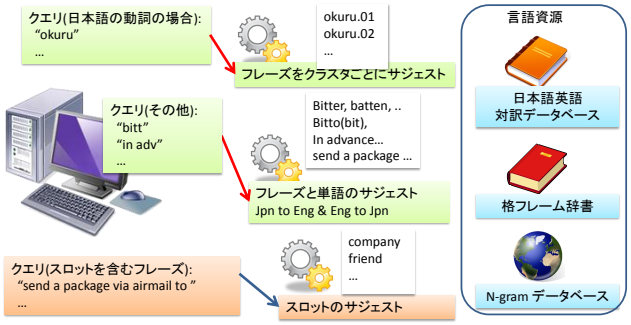

図 6: システムの全体像

に (C) のサジェストを，スロット全てが埋まるまで繰り返す．(A),(B) のサジェストを行うために，システムはキャレットの 30 文字前からキャレット以後にある空白までの文字列 L からクエリを生成する．

(A) 単語・フレーズのサジェスト クエリは次の手順で生成する．空白スペースで L を分割してクエリ配列 $Q_{Japanese} = Q_{English} = [w_1 \cdots w_n]$ を作成する．“nimotsuwookuru” といった分かち書きされていないローマ字化された日本語列の入力にも対応するため，$Q_{Japanese}$ の各要素に対して，ひらがな化したものを KyTea[8] で形態素解析を行い，複数形態素に分割できる場合は，それらに置換する．

そして，システムは単語とフレーズの検索を同時に行う．単語検索では，w_n をクエリにする．

- 単語を日本語から検索
  （例） *hashi* → chopsticks（箸）, edge（端）, post（柱）...
- 単語を英語から検索
  （例） *cong* → congregation, congenital, congressman...

検索結果は 1-gram の頻度の降順でソートされている．

フレーズ検索では，はじめに w_1 から w_n の要素をクエリにして検索する．その検索で結果が 0 件であれば，w_2 から w_n の要素をクエリにして検索する．以下同様に，検索結果が 0 件であれば，次の要素を削除して，クエリにして検索していく．

- フレーズを日本語から検索
  （例） *nimotsu* → carry an armload of packages (nimotsu wo yama no youni kakaete iru)...
- フレーズを英語から検索
  （例） *in adv* → in advance of 〜, in advanced disease...

検索結果は stupid backoff [Brants 07] の言語モデルのスコアの降順でソートされている．

[8] http://www.phontron.com/kytea/

システムはそれぞれの検索結果を図 3 や図 4 のように 1 列にまとめて提示する．

(B) フレーズをクラスタリングしてサジェスト 検索語が日本語の動詞の場合は，フレーズを動詞の格フレームごとにクラスタリングしたデータベース（3.3 章参照）を検索し，図 1 のようにユーザに提示する．

(C) スロットのサジェスト フレーズにスロットを含む場合，スロット部分をワイルドカードに置き換え，n-gram データベースを検索し，言語モデルのスコアの降順でソートして図 2 のようにユーザに提示する．

3.3 データと前処理

格フレーム辞書 フレーズのクラスタリングには京都大学格フレーム辞書 (KCF) ver 1.0 [Kawahara 06][9] を用いた．KCF は 1.6 億文以上のウェブ上の日本語文から自動構築された辞書で，述語は 4 万個，格フレームは平均各述語 13 個含んでいる．

日本語英語対訳データベース 単語・フレーズの対訳辞書として英辞郎 version 134[10] を用いた．英辞郎は，翻訳家によって人手で作られた巨大な対訳データベースで，単語は 330,000 個以上，フレーズはスロット無しのものは 1,434,000 個以上，スロット有りのものは 256,000 個以上を含む．

各エントリの日本語表現は，MeCab 0.994 [11] と IPA 辞書 2.7.0-2007080 [12] で形態素解析し，単語とフレーズのデータベースを構築した．

また，CaboCha0.64 [13] で係り受け解析を行い述語の項構造の取得した．そして，各フレーズの述語項構造と KCF の格フレームの項分布をもとに，フレーズが属する最尤格フレームを求め，クラスタリングしたフレーズのデータベースを構築した．

N-gram データベース 言語モデルのスコアの取得や文脈に基づくスロットのサジェストには Web 1T 5-gram Version 1 [14] を用いた．スロットのサジェストでは “.”, “?”, “< /S >” 等の記号を含むものは除外した．検索には Search System for Giga-scale N-gram Corpus (SSGNC) 0.4.6 [15] を用いた．

[9] http://www.gsk.or.jp/catalog/GSK2008-B/catalog.html
[10] http://www.eijiro.jp/
[11] https://code.google.com/p/mecab/
[12] http://sourceforge.jp/projects/ipadic/
[13] https://code.google.com/p/cabocha/
[14] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13
[15] http://code.google.com/p/ssgnc/

4 評価実験

phloat の有効性を検証するため，10 人の日本人 ESL 学習者に対して英作文問題を課し，*phloat* の有無で流暢性・十分性・作文に要する時間がどのように変化するのかを測る実験を行った．問題は英文電子メールの空所文補充問題・写真の説明文作成問題・日本語文の翻訳問題の 3 種類を用いた．また，流暢性・十分性の評価は 2 人の英語母語話者が 5 段階で行った．しかしながら，統計的に有意な差を得ることができなかった．そのため，ここでは，被験者の作文事例を紹介する．

phloat の利用が効果的だった事例は,「ぶち」(spotted, tabby) といった普段あまり使わない単語に対する翻訳であった．また,「忠告に従う」というフレーズに対して，*phloat* を利用しなかった群は “follow the advice” と訳し，利用した群はシステムのサジェストを用いて（おそらくどの被験者も知らなかったであろう）“Comply with the advice” と訳した事例は，どちらの翻訳も妥当であるが，*phloat* は個人の語彙力を伸ばす可能性をもっている点で興味深い．

一方，*phloat* の利用が効果的でなかった事例は,「彼女は約束を破ったといって彼を責めた」の「〜を責めた」に対する翻訳であった．*phloat* を用いた群は,「責める」に対するサジェスト結果に含まれる “pillory somebody for ...” や “berate someone for ...” といった意味的には正しいが，あまり一般的な言い方ではない表現を選択した事例である．また,「屈服する」の意味で用いられている「屈する」に対して誤って “bow” を選んでしまうという事例もあった．これらは，日常的ではない不適切な表現のサジェストを行ったことと，複数の似たような表現がサジェストされたとき被験者には選択の手がかりがなかったことが原因であると考える．

5 今後の課題

5.1 語彙選択を助けるための例文提示

語句の微妙なニュアンスの違いを ESL 学習者が区別するのは難しい．例えば，“home” と “house,” “at last” と “finally,” “must” と “have to” の違いなどである．これらをシステムが自動的に文脈から判断するのは難しいため，幾つかの用例をユーザに提示することで，ユーザの候補選択を支援できると考える．

5.2 文脈情報を用いたより良いサジェスト

現在は，サジェスト内の候補のランキングに英語の語句の頻度・言語モデルスコアのみを用いているが，“一般的” な語句が上位にランクされるという問題が見つかった．例えば，“blue” を「海」の語義で用いることは稀であるが,「海」の訳語の 1 つとして英辞郎には “blue” が載っており，“blue” の頻度は “sea” よりも格段に高いため，現在のシステムでは “blue” を高くランク付けしてしまう．この問題を避けるには，表層形ではなく語義ごとの頻度の情報が必要である．

さらに，前後の文脈情報も用いることで，より良いランキングを生成できると考える．例えば，後続する語句の品詞は文脈によっては絞りこみが可能である．動詞の直後には名詞（句）が，“have” や助動詞の直後にはそれぞれ過去分詞や動詞の原形が続く可能性が高い．

また，語句のコロケーションも考慮してサジェストする必要もある．例えば,「おおきい」は “large” や “many”, “big” 等に翻訳し得るが，どれが適切であるかは修飾される語句によって異なる．例えば “population” に対しては “large” が最適である．

参考文献

[Brants 07] T. Brants *et al.*: Large Language Models in Machine Translation, *EMNLP-CoNLL*, pp. 858–867 (2007)

[Burstein 04] J. Burstein, M. Chodorow, C. Leacock: Automated essay evaluation: the criterion online writing service, *AI Magazine*, Vol. 25, No. 3, pp. 27–36 (2004)

[Chen 12] M. Chen *et al.*: FLOW: A First-Language-Oriented Writing Assistant System, *ACL*, pp. 157–162 (2012)

[Doi 98] S. Doi, S.-i. Kamei, K. Yamabana: A text input front-end processor as an information access platform, *COLING*, pp. 336–340 (1998)

[Esteban 04] J. Esteban *et al.*: TransType2: an innovative computer-assisted translation system, *ACL*, pp. 94–97 (2004)

[Huang 12] C.-c. Huang *et al.*: TransAhead: A Writing Assistant for CAT and CALL, *EACL*, pp. 16–19 (2012)

[Kato 08] Y. Kato *et al.*: English Sentence Retrieval System Based on Dependency Structure and its Evaluation, *ICDM*, pp. 279–285 (2008)

[Kawahara 06] D. Kawahara *et al.*: Case Frame Compilation from the Web using High-Performance Computing, *LREC*, pp. 1344–1347 (2006)

[Kuhn 08] T. Kuhn, *et al.*: Writing Support for Controlled Natural Languages, *The Australasian Language Technology Association Workshop 2008*, pp. 46–54 (2008)

[Leacock 09] C. Leacock, *et al.*: User input and interactions on Microsoft Research ESL Assistant, *The 5th Workshop on Innovative Use of NLP for BEA*, pp. 73–81 (2009)

[Liu 00] T. Liu *et al.*: PENS: A Machine-aided English Writing System for Chinese Users, *ACL*, pp. 529–536 (2000)

[Muraki 94] K. Muraki *et al.*: TWP: How to Assist English Production on Japanese Word Processor, *COLING*, pp. 847–852 (1994)

[Takamatsu 12] 高松優ら: 英作文支援のための用例検索システムの構築, 言語処理学会 第 18 回年次大会予稿集, pp. 361–364 (2012)

[Wible 10] D. Wible, *et al.*: StringNet as a Computational Resource for Discovering and Investigating Linguistic Constructions, *The NAACL HLT Workshop on Extracting and Using Constructions in Computational Linguistics*, pp. 25–31 (2010)

[Yamabana 97] K. Yamabana *et al.*: A hybrid approach to interactive machine translation: integrating rule-based, corpus-based, and example-based method, *IJCAI*, pp. 977–982 (1997)