6S-09

Max-SAT ソルバーを用いた単語帳作成手法

杉山　幹太†　　岡本　圭史†
仙台高等専門学校†

1　はじめに

今日，英単語を覚えるための様々な単語帳が市販されている．特に例文を覚えることで単語を覚えるというコンセプトに基づき開発された単語帳として例文型単語帳がある．例文型単語帳では 1 例文につき記憶対象となる複数の単語を含ませることで，覚えるべき例文数を減らすという工夫がなされている．しかし，現在その工夫は英語を母語にする人などが適切な英文を考えることにより実現される[1]ため，例文型単語帳作成には多大なコストがかかる．

近年SAT問題に関しての研究が活発になされてきた．その結果この十数年で SAT ソルバーの性能は飛躍的に向上し，SAT ソルバーの応用についての研究も盛んになってきた[2]．Max-SAT ソルバーは SAT ソルバーの拡張であり，SAT ソルバーの発展とともに性能が向上し，最小頂点カバー問題や回路設計などに応用されている[3]．

本研究では著名な英文学作品等から，Max-SAT ソルバーを用いて適切な英文を選択する手法を提案し，例文型単語帳を作成する．この手法は単語帳作成のコストを削減し，さらに，覚えるべき英文を興味深くする効果が期待できる．

2　Max-SAT ソルバー

はじめに，CNF(Conjunctive Normal Form)式について解説し，その後Max-SATソルバーについて解説する．CNF式とは，以下の形式で書かれた命題論理式のことである．

・リテラル ≔ 命題変数 | 命題変数の否定
・節 ≔ リテラル | (リテラル ∨ … ∨ リテラル)
・CNF式 ≔ 節 | 節 ∧ … ∧ 節

以下，“真”を“1”，“偽”を“0”で表す．Max-SAT問題とは，CNF式及び各節がハード制約なのかソフト制約なのかという2つの入力に対し，全ハード制約節を充足し，充足するソフト制約節数を最大化するような命題変数への値の割り当てを求める問題である．

例として次のMax-SAT問題を考える．

$\psi = (p \vee q) \wedge p \wedge \neg p \wedge \neg p$，ただし最初の節のみハード制約節で残りはソフト制約節とする．

この場合pの値のみで充足するソフト制約節数が決まり，p=0のとき2つ充足して最大数充足することになるため，pには0を割り当て，ハード制約節を充足する割り当てを探す．このときハード制約節の真偽はqの真偽と等価であるが，ハード制約節は必ず充足しなくてはならないのでqには1が割り当てられる．

A method to generate wordbooks with a Max-SAT solver
†Kanta Sugiyama, †Keishi Okamoto
†Sendai National College of Technology

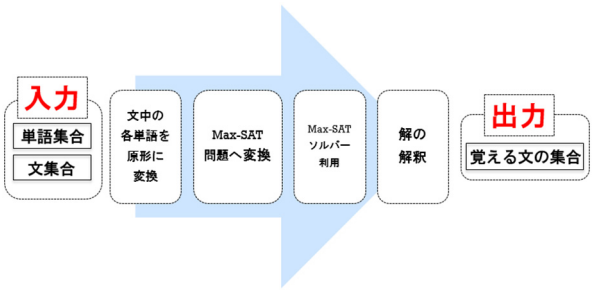

図1　単語帳作成手法の流れ

単語集合
W={remove,oil,stain}

文集合 S={s_1, s_2, s_3, s_4}
s_1=I removed the dishes from the table.
s_2=Oil and water don't mix.
s_3=Can you get this stain out?
s_4=Remove the oil stain.

図 2　単語帳作成問題の例

Max-SAT ソルバーとは Max-SAT 問題を解くプログラムである．Max-SAT問題の解は「基数制約を追加したSAT問題」を複数回解くことで得られる[3]．さらに，基数制約は入力数を数える回路，比較回路で SAT Encoding 可能[4]である．つまり，Max-SAT ソルバーは SAT ソルバーに基づいて実装可能である．

3　Max-SAT ソルバーを用いた単語帳作成手法

単語帳作成手法の流れを図 1 に示す．この流れを解説する．まず，覚えたい単語の集合を決める．次に，人力による手法ではこの単語集合から例文を考えてゆくが，提案手法では英文を英文学作品などから集める．その収集した英文から覚えたい単語を全て含むような英文の集合のうち英文数が最小になるような集合を出力する．この英文の集合が覚えるべき英文で，英文に含まれる単語の意味を日本語で追記し，単語帳を作成する．この適切な英文の集合を選ぶ問題を単語帳作成問題と呼ぶことにし，以下のように定式化する．

単語帳作成問題は，単語集合$W = \{w_1, \ldots, w_m\}$，文集合$S = \{s_1, \ldots, s_n\}$が入力されたとき，

$$f(s_i) = \{w \in W \mid w \text{は} s_i \text{で使われる}\}$$

で定義する写像$f: S \to P(W)$に対し，$X \subseteq S, f(X) = W, |X|$は最小という条件を満たすXを求める問題として定式化される．この問題は$f(s_1), \ldots, f(s_n)$それぞれを被覆集合，Wを全体集合とする集合被覆問題と等価である．

fを求め，集合被覆問題を解くことにより単語帳作成

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.

問題が解ける．fを求めるには，任意の文sについてs中の各語を原形に変換し，単語集合中の任意の単語wを文s内で文字列探索すればよい．

単語帳作成問題を解くのに，厳密解が得られるのが望ましい．しかし，集合被覆問題はNP困難な問題であることが知られており[5]，厳密解を求めるのは一般に困難である．そこで[6]のように，近似解を得る研究が特に進められている．しかし，単語帳作成問題に限定した場合，厳密解を現実的な時間内で得られるサイズの問題であると予想される．そこで，集合被覆問題（単語帳作成問題）を Max-SAT 問題に変換し厳密解を求めることにする．命題変数p_iを「文s_iを出力する」という主張を表す命題とする．各$w \in W$について

$$J_w = \{j \in \{1, \dots, n\} | w \in f(s_j)\}$$

と置くとき，$f(X) = W$を満たすためには，単語wを含む文を出力しなければならないので，$\bigvee_{j \in J_w} p_j$が成り立つ必要がある．さらに，全ての$w \in W$について$\bigvee_{j \in J_w} p_j$が成り立つ必要があるので，結局$\bigwedge_{w \in W} \bigvee_{j \in J_w} p_j$が成り立つ必要がある．他方，$|X|$が最小とは，出力しない英文を最大にすること，すなわち$\neg p_1, \dots, \neg p_n$を最大限1にすることと同値である．以上より，集合被覆問題は次の Max-SAT 問題と同値である．

$$\left(\bigwedge_{w \in W} \bigvee_{j \in J_w} p_j\right) \wedge \left(\bigwedge_{k=1,\dots,n} \neg p_k\right)$$

ただし1つ目の括弧内にある節は全てハード制約節で，残りは全てソフト制約節とする．

例として図2に示す単語帳作成問題を考える．
この問題では，文s_1に含まれる単語を原形に直すと I remove the dish from the table.になり，remove という単語が使われていることがわかる．このようにして写像 f が以下のように得られる．

$$f(S) = \{\{remove\}, \{oil\}, \{stain\}, \{remove, oil, stain\}\}$$

よってこの単語帳作成問題は次の Max-SAT 問題と等価である．

$$(p_1 \vee p_4) \wedge (p_2 \vee p_4) \wedge (p_3 \vee p_4) \wedge \neg p_1 \wedge \neg p_2 \wedge \neg p_3 \wedge \neg p_4$$

ただし最初の 3 つの節がハード制約節で，残りはソフト制約節とする．これを解くと解$p_1 = 0, p_2 = 0, p_3 = 0, p_4 = 1$が得られるので，命題$p_i$の定義より$X = \{s_4\}$である．

4　実装評価と考察

Max-SAT ソルバーについて調査し，実用的な時間で解ける問題のサイズを見積もるため２章で紹介した方法[3][4]を用いて Max-SAT ソルバーを実装した．基となる SAT ソルバーには plingeling[7]を採用した．また，３章で提案した手法のうち Max-SAT 問題へ変換する部分，解を解釈する部分，他プログラムを呼び出す部分を Python 言語により実装した．ただし各単語を原形に変換する処理は MontyLingua[8]を利用した．

実装したプログラムに対して“The Adventures of Sherlock Holmes”[9]全文(7477 文)，その中での頻出度が 3001～6000 位の 3000 単語を入力した．その結果，1896 文が出力され，これらの文の集合は入力した全単語を含むことを確認した．

[1]では，単語と熟語を合わせて 2569 語を 560 文で覚える．この場合１文あたり平均約 4.6 語含んでいる．本研究の結果では 1 文あたり平均約 1.6 単語含んでおり，人力の場合よりは 1 文あたりの単語数が少ない．文の出力に要した時間は 63.19 秒（CPU:Intel(R) Core(TM) i5-4440，クロック周波数:3.10GHz，メモリ:4GB）で，人力による単語帳作成より高速である．

今回の実験では 1 文は平均約 1.6 単語を含んでいたが，例えば impatience，dread，restrain という 3 単語に対し Even my dread of losing a client could not restrain me from showing impatience.という文が選ばれたケースもあった．既存の文学作品中の文を用いて単語帳を作成するため，人手による単語帳作成と比較して 1 文に含まれる平均単語数は少ないが，入力する例文数を増やすことで 1 文に含まれる単語数の向上が期待できる．

今回は英語の単語帳を作成したが，提案する手法は言語への依存度が低いため，多言語対応が容易である考えられる．また，今回英文の数を最小にすることを目的としたが，出力する文の長さの総和を最小にすることにより学習者の負担を軽減することも考えられる．これは今回 Max-SAT 問題と考えたものを重み付き Max-SAT 問題に変えることにより実現可能である．

5　おわりに

本研究では Max-SAT ソルバーを用いた単語帳作成手法を提案し，それを実装した．実際に英文学作品から英文を入力し，提案する手法が単語帳作成のコストを削減できることを確認した．

参考文献

[1]鈴木 陽一．DUO 3.0．アイシーピー，2000，432p
[2]井上 克巳，田村 直之．“SAT ソルバーの基礎”．人工知能学会誌．2010，Vol.25，No.1，pp57-67
[3]平山 勝敏．“＊－SAT：SAT の拡張”．人工知能学会誌．2010，Vol.25，No.1，pp105-113
[4] Carsten Sinz，“Towards an Optimal CNF Encoding of Boolean Cardinality Constraints”．Principles and Practice of Constraint Programming - CP 2005．2005，pp827-831
[5] Bernhard Korte，Jens Vygen．Combinatorial Optimization．Springer，2012，659p
[6]岸田正博，柳浦睦憲，茨木俊秀．“集合被覆問題に対する局所探索法について”．数理解析研究所講究録．1999，Vol.1114，pp211-220
[7]Armin Biere．“Lingeling，Plingeling and Treengeling”．http://fmv.jku.at/lingeling/，（参照 2014-12-15）
[8]Hugo Liu．”montylingua：：a free，commonsense-enriched natural understander”．http://web.media.mit.edu/~hugo/montylingua/，（参照 2014-12-15）
[9]Arthur Conan Doyle．The Adventures of Sherlock Holmes．George Newnes，1892，

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.