

ロジスティック回帰による言語モデルを用いた 将棋解説文の自動生成

亀甲 博貴¹ 三輪 誠² 鶴岡 慶雅¹ 森 信介³ 近山 隆¹

¹ 東京大学大学院工学系研究科

² マンチェスター大学コンピュータ科学科

³ 京都大学学術情報メディアセンター

¹{kameko, tsuruoka, chikayama}@logos.t.u-tokyo.ac.jp

²makoto.miwa@manchester.ac.uk

³forest@i.kyoto-u.ac.jp

1 はじめに

近年のコンピュータ将棋プログラムの棋力向上は目覚ましく、プロ棋士と公式に対局が行われるに至るまでその強さが認知されるようになった。また有力な将棋プログラムの中には、将棋ファンが入手し気軽に扱えるようになってきているものも多い。これらの将棋ソフトの多くは局面の形勢判断を評価値として返す機能を持つことから、プロ棋士同士の対局においてその形勢を将棋ソフトを用いて判断する光景もよくみられるようになった。棋力の高いプレイヤーの対局を観戦する上で、ソフトが判断する形勢や評価の高い指し手は大いに役立つといえる。

しかしこれらのプログラムが返すのは評価の高い指し手や形勢のみであり、なぜその指し手が選択されるのかを理解するには観戦者にある程度の棋力が求められる。そのため棋力があまり高くない観戦者がプロ棋士同士の対局を楽しむためには、プログラムの読み筋だけでは不十分である。一方でタイトル戦など注目されている公開対局ではプロ棋士が解説を行っている場合が多い。棋力の上でプロ棋士に近づきつつあるプログラムによる局面の解説が可能となれば、解説者がつかないような対局において将棋ファンの観戦に有益であると期待できる。

解説文生成の先行研究として、コンピュータ将棋が指し手を決定する際の探索結果を出力するものがある [1]。棋力の高いコンピュータ将棋プログラムの1つである GPS 将棋の推奨手順と評価値と呼ばれる数値化された形勢判断基準を、あらかじめ与えたルールによって人が読みやすい形に加工し、ミニブログサービスの1つである Twitter¹ に投稿することで広く将棋ファンに向けて公開するというものである。コンピュー

タ将棋プログラムが出力する手順やその評価値は多くの将棋ファンの将棋観戦に有益な情報であるが、一方でその手順の意味するところを解説するには至っていない。多様な局面において充実した解説を将棋ファンが理解できる形で提供するためには、自然言語を用いる必要があると考えられる。

本稿ではコンピュータ将棋プログラムによる将棋の解説文生成モデルを2段階の機械学習により獲得することを目指す。まず与えられた局面に対する解説文にふさわしい単語群を獲得する。その後得られた単語群と言語モデルを組み合わせ、解説文の生成を試みる。

本稿では2章で提案手法を紹介する。3章で提案手法の評価について述べ、4章でまとめを述べる。

2 提案手法

本研究では図1のような2段階での解説文生成手法を提案する。まず与えられた局面に対し、その局面の解説文に現れうる特徴的な単語を予測する。次に予測した特徴的な単語と言語モデルを組み合わせ、その局面に対する解説文を生成する。

2.1 局面に特徴的な単語の予測

ある局面が与えられたときにその局面を解説する際に現れそうな単語を予測する。

入力として、局面を表現する素性ベクトルを用いる。用いた素性はコンピュータ将棋プログラム「激指」が局面を評価する際に用いているものである。「激指」は素性として、自分の駒の価値や駒の利きなどの他に2駒間の関係を用いている。例えば自玉と自分の金の位置関係や自分の金銀の位置関係などは自玉の守りがどの程度効率的かを評価する指標になり、敵玉と自分の駒の位置関係は敵玉にどれだけ迫っているかの指標になる。これらは局面理解の上で重要な素性であり、局

¹GPS 将棋のアカウント: <https://twitter.com/gpsshogi>

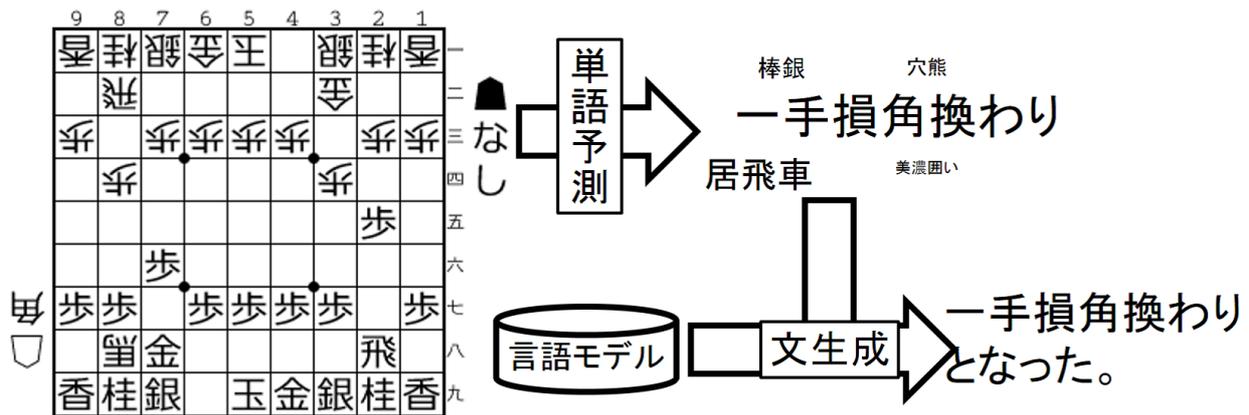


図 1: システムの全体像

面を表現する単語の予測に有効であると期待できる。

出力として、教師値は対象となる各単語が解説文中に現れるまたは現れないことを示すベクトルを用いる。予測する際は各単語について 0 から 1 までの範囲の値を持つベクトルを返す。

学習器には 3 層パーセプトロンを用いる。与えられた局面の素性ベクトルを入力すると、その局面の解説文への各単語の現れやすさを表現する語彙数次元の予測ベクトルを出力するよう学習する。

2.2 解説文生成

前項で得られた単語の予測結果と言語モデルを組み合わせて解説文を生成する。

ある局面 ϕ が与えられたとき、その局面に対して解説文 $S = w_1, w_2, \dots, w_n$ が生成される確率 $P(S | \phi)$ を

$$\begin{aligned}
 &= P(S_N | \text{length}(S_N) = n) \\
 &\quad \times P(w_1 | \phi) \times P(w_2 | \phi, w_1) \times P(w_3 | \phi, w_1, w_2) \\
 &\quad \times \dots \times P(w_n | \phi, w_1, w_2, \dots, w_{n-1}) \\
 &= P(S_N | \text{length}(S_N) = n) \\
 &\quad \times \prod_i^n P(w_i | \phi, w_1, w_2, \dots, w_{i-1}) \quad (1)
 \end{aligned}$$

と定義する。ここで $P(S_N | \text{length}(S_N) = n)$ は長さ n の解説文が生成される確率である。これを最大にする S が与えられた局面に対して生成されるべき解説文であるとする。

式 (1) 中の $P(w_i | \phi, w_1, w_2, \dots, w_{i-1})$ は、文が途中まで生成されているときに次に w_i が続く確率である。これを前項で得られた予測ベクトルと言語モデルから推定する。言語モデルとしてトライグラム言語モデルを用いる。スムージングにはワン・カウント法 [2] を用いる。

推定はロジスティック回帰により行う。ロジスティック回帰は

$$\log \frac{p(w_i = w_v)}{1 - p(w_i = w_v)} = W_{w_v}^T \phi \quad (2)$$

で定義される。これを変形し、

$$p(w_i = w_v) = \frac{1}{1 + e^{-W_{w_v}^T \phi}} \quad (3)$$

と表せる。これを正規化し、

$$P(w_i = w_v) = \frac{p(w_i = w_v)}{\sum_j p(w_i = w_j)} \quad (4)$$

これを言語モデルとして用いる。

得られた言語モデルを用いて定義される生成確率が最大となる文の探索を行う。空の状態をルートノードとし、語彙中の 1 単語が続いたものを子ノードとして展開する。文の終端記号が現れるまで展開を行い、それを 1 文とする。本研究では単語数が 30 を超えたものについても生成確率 0 の文であるとして展開を打ち切る。

最良優先探索により生成確率の高い文を探索する。全探索を行おうとするとその時間計算量・空間計算量はともに膨大になる。そこで本研究ではビームサーチを行い計算量の削減を図る。途中までの生成確率でソートされた優先度付きキューを用意する。この中から途中までの生成確率が最も高いノードを取り出す。このノードを展開し、次の 1 単語を追加した生成確率を求める。これら展開された子ノードを優先度付きキューに追加する。以上の操作を、生成確率の最も高いノードが終端記号に到達するまで繰り返す。取り出されたノードが終端記号に到達している場合、これが生成確率の最も高い文であるとして出力する。

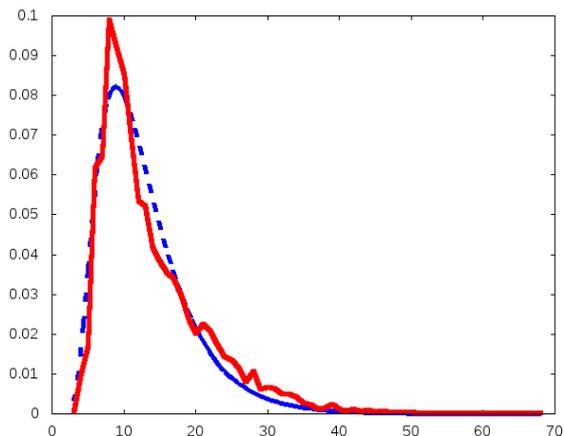


図 2: 単語長と出現確率の関係 (実線が実際の値, 点線が近似関数)

3 評価

提案手法による解説文生成の評価を行った。評価には、名人戦棋譜速報²において有料で配布されている名人戦と順位戦の人による解説文つき棋譜のうち 2011 年度以前のものを用いた。簡単のため対象とする解説文は戦型に言及するもののみとした。そのため各解説文が戦型に言及するものであるか否かの判別器を事前に Bag of Words を入力とするパーセプトロンによって学習し、これが戦型に言及するもののみを用いた。戦型に言及するもの文は 10,540 文あり、このうち 8,000 文を学習用データ, 2,540 文を評価用データとした。

形態素解析器は「KyTea」[3]を用いた。KyTea は分野適応に優れるという特長を持つ形態素解析器であり、本研究においては将棋に関連する 300 文を用いて学習を行ったものを用いた。

単語長ごとの文の出現確率を図 2 に示す。本研究ではこれを、

$$\sqrt{\frac{52}{2\pi n^3} \exp\left(\frac{-52(n-12.85)^2}{2 * 12.85^2 n}\right)} \quad (5)$$

で近似し、これを $P(S_N | length(S_N) = n)$ とする。

3 層パーセプトロンの実装は Fast Artificial Neural Network Library (FANN) [4] を用いた。隠れ層は 50 次元とし、それぞれの層でシグモイド関数を用いて隠れ層は $[-1, 1]$, 出力層は $[0, 1]$ の連続量とした。

単語の生成確率を推定するロジスティック回帰の実装には LIBLINEAR [5] を用いた。単語の予測ベクトルについては長さ 1 となるように正規化し、言語モデルと同程度の大きさとなるようにした。探索時のビーム幅は 1,024 とした。

²<http://www.meijinsen.jp/>



単語	値	単語	値
い	0.964	穴熊	0.730
対	0.451	美濃	0.426
最近	0.279	ゴ	0.222

生成文: ゴキゲン 中 飛車。

正解: (後手棋士名) のトレードマークである ゴキゲン 中 飛車。

図 3: ゴキゲン中飛車の局面

生成された解説文の BLEU スコア ($N = 4$) は 0.0719 と、人間による解説文と同様のものを生成できているとは言えない。

局面の例と実際に生成された解説文を以下に示す。

図 3 は後手がゴキゲン中飛車と呼ばれる戦法を選択した局面である。なお形態素解析器による単語分割は「ゴ / キゲン / 中 / 飛車」である。この局面に対し、出てきやすい単語として穴熊や美濃という囲いの名前が現れている一方で、ゴキゲン中飛車にかかわる単語は「ゴ」のみ、かつその値も他の単語と比べると小さい。しかし生成される文を見ると正しくゴキゲン中飛車であると言えており、生成に成功している例であると言える。正解の文は実際に選択した後手の棋士がゴキゲン中飛車をよく指す棋士であるということで、その情報を元にして生成している。これと比較して生成の精度を BLEU で評価しようとする、生成されえない前半部分がないためにそのスコアは低くなってしまふ。人間がつけた解説文は将棋の解説文のみではなくファンが読んで楽しいものであるが、そのために文生成の正解としてそのまま用いるには適さない可能性がある。

図 4 は先手が矢倉へと向かった局面である。生成文と正解文の間の単語の一致は「矢倉」と「。」のみでありバイグラム以上の共通項はないが、どちらも矢倉囲いへと向かっていることを言っておりその意味は概



単語	値	単語	値
下	0.964	矢倉	0.900
道	0.581	相	0.447

生成文：矢倉 模様 の 出だし。
 正解：矢倉 を 目指 す 一 手。

図 4: 矢倉模様の局面

ね一致する。BLUE スコアでの比較をするとその値は悪いが、この局面の成功の例であるといえる。

図 5 は後手が△6 四歩とし、△6 三銀～△5 四銀の腰掛け銀へと向かおうとした局面である。この局面に対して予測単語中に腰掛け銀は現れず、生成文も「ゴキゲン中飛車。」と全く誤っている文を生成している。この局面の先後それぞれの陣形を見ても中飛車に進むことはなかなか考えられない。単語の予測精度が低いために解説文生成に失敗している例であるといえる。

4 おわりに

本稿では将棋解説文の自動生成手法として 2 段階での生成手法を提案し評価した。人間がつけた解説文を参照文とした BLEU スコアによる比較では 0.0719 と非常に低いものであったが、実際に局面と生成文を見みると有効な文を生成できた例が複数見られた。

本稿では客観評価として人間による解説文を正解とした BLEU スコアを用いたが、人間による解説文の中には本研究で対象としていない、対象ゲーム外の情報を含んでいるためにそのまま正解であるとするのには問題がある。機械学習の教師値として、あるいは評価時の正解データとして人間による解説文をどのように使うかの検討が必要であると考えられる。

今後の課題として、単語の予測精度向上が挙げられる。用いる素性の設計などにより、より高精度な単語予測とそれに伴う文生成が期待できる。

また文生成においても、次の単語選択においては



単語	値	単語	値
キゲン	0.829	棒銀	0.620

生成文：ゴキゲン 中 飛車。
 正解：後手 は 腰掛 け 銀 を 目 指 す。

図 5: 腰掛け銀の局面

トライグラムしか用いておらず、全体としての評価を行っていない。発話内容の評価を行うことでより適した文生成が期待できる。

参考文献

- [1] 金子知適. コンピュータ将棋を用いた棋譜の自動解説と評価. 情報処理学会論文誌, Vol. 53, No. 11, pp. 2525–2532, 2012.
- [2] Stanley F Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pp. 310–318, 1996.
- [3] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable japanese morphological analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pp. 529–533, 2011.
- [4] Steffen Nissen and Evan Nemerson. Fast artificial neural network library. Available at leenissen.dk/fann/html/files/fann-h.html, 2000.
- [5] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, Vol. 9, pp. 1871–1874, 2008.