

2 elmo の開発と技術

—第 27 回世界コンピュータ将棋選手権優勝プログラム
インタビューから



< elmo 開発者 > 瀧澤 誠 | KDDI (株) < 聞き手 > 伊藤毅志 | 電気通信大学

本稿について

本企画は、執筆の時期がちょうど 2017 年 11 月に開催される「電王戦トーナメント」のために開発者が準備をしている時期で、選手権で優勝した“elmo”も電王戦トーナメントに向けて、開発に忙しい時期であったため、原稿執筆が難しく、伊藤が開発者の瀧澤誠氏にインタビューを行い、口述内容を書き起こし、その内容から要点をまとめて、再構成したものである。

elmo 開発の経緯

伊藤「コンピュータ将棋に興味を持ち始めたのはいつ頃からですか？」

瀧澤「コンピュータ将棋自体は実はだいぶ前から見てはいました。実際触り始めたのが、“Bonanza^{☆1}”が公開されて 1, 2 年経ってからです。私が大学生の頃です。私が今 34 歳なので、20 歳前半ぐらいの頃ですかね」

伊藤「最初に勉強されたのは、Bonanza ということですか？」

瀧澤「そうですね、まずは、Bonanza を読むというかたちでした。Bonanza の機械学習を使ってというところが、大学でちょっと機械学習を習ったということもあって、やっていることがなんとなく

く分かったのです。実際、ソースコードを見てみるとあんまり綺麗な機械学習になってなくて、学習するときに乱数を使ってやっていたりして、いじってみたら強くなるんじゃないかなと思ったのがきっかけです。でも実際にはよくできていて、僕がいじるぐらいだったら全然強くない。触ってはみたんですけど、結局あまり強くないなあという時期が続いていまして、大会に出るというモチベーションにはなりませんでした」

伊藤「それが大会に出るというモチベーションに変わっていったきっかけを教えてください」

瀧澤「今、主流になっていますけど、プロの棋譜から学習するんじゃなくて、コンピュータが自分で考えて強くなるというのをやりたいと思っていて、それで試行錯誤し始めたのが、出たいと思ったきっかけです」

伊藤「それはいつ頃ですか？」

瀧澤「大会に出たのは去年 (2016 年) からですね、でも、もうちょっと前から触っていました。新しいアイデアがあって、そこで試してみようと思ったんですけど、あまり上手いかなかった。大会エントリーしてから、実際の大会までに 4 カ月ぐらいあるんですけど、エントリーしたときにはできそうだと思っていたんです。でも、やってみるとなかなかみたいな…… (笑)。ざっくり、2, 3 年前ぐらいからですかね」

伊藤「昨年から今年にかけて、相当に強くなったと思いますが、その間に何か特別なことをしたのですか？」

瀧澤「実は特別なことをやったわけではありません。

☆1 Bonanza は、2006 年世界コンピュータ将棋選手権に初出場初優勝を果たした保木邦仁氏開発の将棋プログラム。選手権出場前 2005 年頃から実行プログラムはネット上で公開され、2009 年 1 月にソースコードが公開された。選手権優勝レベルのプログラムの公開はそれまでに例がなく、これによって、コンピュータ将棋界全体のレベルは飛躍的に向上した。

たまたま上手くいったというところですかね」

伊藤「具体的にどんなところを改良したのですか？」

瀧澤「まず、全体の流れとして、コンピュータ将棋として、ちょうど1年前（2016年）の電王トーナメントのときに、コンピュータ同士が戦って強くしていくという方法が確立されて、去年の大会後にソースコードとして公開されました」

伊藤「それはどのプログラムですか？」

瀧澤「“Apery（エイプリー）^{☆2}”ですね。当時は“浮かむ瀬”という名前が出てました。そこで使われていた手法は、簡単に言うと、ちょっと深く読んだ時の評価値と、ほとんど読まない時の評価値の差を縮めていこうっていう手法だったんです」

伊藤「最近のコンピュータ将棋の流れですね」

瀧澤「はい、コンピュータが自己対局して、それを使って強くしていくというのが、約1年ぐらい前から主流となっている技術です」

伊藤「アイディアは前からあったんですけど、本格的に使われるようになったのは、最近ですね」

瀧澤「それで一気に強くなって、レーティングにして400点ぐらい上がっている感じです」

伊藤「そのあたりについても少し解説してもらえますか？ 具体的にどれぐらいの深さを読んで、それを学習に反映させているのかとか」

瀧澤「大体6～8ぐらいの深さになっていると思うんですよ」

伊藤「それはなぜその深さになるんですか？」

瀧澤「計算量が一番の要因ですね。大体これぐらいで頭打ちになるんです。それ以上やってもあまり効果が出ない。深さが1増えると、計算量が大体2倍になってくるんで、8でやっておけば文句はないだろうと（笑）」

伊藤「どれぐらいの局面を学習するんでしょう」

瀧澤「1回に使う局面数は、人によってまちまちですけど、大体数十億ですかね。最近では20億あると足りていると気がするんですけど」

伊藤「その20億局面はどのように作るんですか？」

瀧澤「自己対局です」

伊藤「自己対局だと、どうしても局面の偏りが出てしまったりすると思うんですが、そういう問題はどうしてるんですか？」

瀧澤「最初に乱数入れてます。序盤の局面をバラけさせるということです。対局途中で乱数を入れてしまうと結果が悪くなってしまいますので、序盤を分散させるしかないんです」

伊藤「具体的に序盤でどんなふうに関数を加えているんですか？」

瀧澤「私は初手から24手目までの間に5手乱数を加えています。この5手というのも、“やねうら王”のデフォルトをそのまま使っているんですけど。実際、24手全部乱数を入れてもほとんど強さ的に変わらないんです……。なんでですかね」

伊藤「あんまり増やしてもあり得ない局面ばかり増やしてしまうことになるんでしょうかね」

瀧澤「ただ、そこからどんどん対局を進めていくと、ありそうな展開になっていくんだろうなって気がするんですけど」

伊藤「もしかすると、数字に表れないけど、どこかで強くなっているかもしれないですね」

瀧澤「ああ、そうですね。あると思います。やっぱり乱数を増やしたほうが学習の時間がかかっていて、たぶん、そっちのほうが広く値がついているんだろうなって思いました」

伊藤「あまり見かけない局面になったときに強いとか」

瀧澤「それは、あるかもしれません」

伊藤「瀧澤さんの工夫した点について、詳しく聞かせてください」

瀧澤「従来のものに加えて、自己対局したときにその局面から進めて勝ったか負けたかっていう情報

^{☆2} 平岡拓也氏による将棋プログラム。2014年の世界コンピュータ将棋選手権で優勝している。プログラム名の由来は、「猿真似」で、良いものは何でも真似するという設計方針とのことである。「浮かむ瀬」とは、2016年の電王トーナメントの出場時に、同プログラムの命名権をチャリティ募金に出品して付けられた名前である。

を、学習するときに加えたんです。それが効果があったみたいです」

伊藤「単に深く読んだ評価値の結果だけを使うのではないということですね？」

瀧澤「そうですね。両方使っています」

伊藤「その勝敗の結果を具体的にどのように導入したのですか？」

瀧澤「完全にロジスティック回帰です。勝ったら1、負けたら0というふうに」

伊藤「それがほかのプログラムはやっていない工夫ということですか？」

瀧澤「やっていなかったみたいです。 “ponanza^{☆3}” は分かりませんが、勝ったか負けたかだけで学習するっていうのは、当然、強化学習であるので、それは激指がやっていたのですが、それを組み合わせるってのがなかった」

伊藤「それで実際どれくらい強くなりましたか？」

瀧澤「そうですね。持ち時間によって全然違って、持ち時間が長ければ長いほど既存のものに比べて勝つ傾向にあって、だいたいレーティングにして200点ぐらい上がった感じです」

伊藤「200点というのはかなりの向上ですね。勝敗の結果の使い方について詳しく教えてください」

瀧澤「ある局面から対局させたとして、勝つか負けるかで、勝つと1で、負けると0というふうに、勝率で最適化させていくんです。それとは別に、従来 Apery とかが使っていた手法のように、深く読んでいったときの評価値が△だとしたら、それをその局面における評価値に近づける。たとえば、ある局面の評価値が+500点だとすると、勝率でいうと大体70%ぐらい。実際はもう少し高いと思いますが、ここでは70%ぐらいとします。そして、仮にこれで勝ったとすると、たとえば

300点ぐらい加算してやります。いくら加算するかというのは、色々なパラメータによるのですが、大体500点から100点ぐらいの間の数字に決めました」

伊藤「その加算の計算方法っていうのはあるんですか？」

瀧澤「評価値が500点の勝率が70%で……。あまり詳しい値は覚えてないんですけど、評価値が300点だと60%ぐらいだとしますね。勝率計算は、割とロジスティック回帰は単純で、70% - 60%で、+10%になるじゃないですか。勝った場合は100%から（現在の）60%を引いて+40%。で、これとこれのパラメータの係数があるんですけど、勾配計算なんで、あの勝ったときは、いくつかがあって、プラスいくつっていう数字が出てきて、これを100万局面数ずつ決めて、プラスになるのかマイナスになるのかなんですけど、プラスだったら “AdaGrad” で更新していきました。僕の場合 eta = 50 とかにして、だんだん小さくしていく」

伊藤「この勝敗による加算が改良に効いたってことでしょうか？」

瀧澤「はい」

伊藤「なんだかモンテカルロの結果を学習に入っているような感じですね」

瀧澤「まさにそうですね。将棋も囲碁もやっていることは大して変わらないですね」

伊藤「理にかなっている感じがしますね。もしかすると勝敗だけでやったほうがいいんじゃないですか？」

瀧澤「そんな気がしたので、やってみたんですが、ちょっと弱くなってしまいました。多分局面数が少ないからかと」

伊藤「なるほど、十分な局面数が必要なのか。評価値の改善に勝敗の結果を加えたというところがミソなんですね」

☆3 山本一成氏をメインプログラマとする将棋プログラム、2013、2015、2016年電王トーナメントで優勝、2015、2016年コンピュータ将棋選手権優勝と近年コンピュータ将棋界をリードするプログラム。2013年に出場した第2回将棋電王戦において、佐藤慎一四段（当時）に勝利し、初めて現役男性プロ棋士に平手で勝利したプログラムでもある。

今後の改良について

伊藤 「今後の改良についてもお聞かせいただきたいのですが、特に注目していることはありますか？」

瀧澤 「今やっているのは、4駒関係^{☆4}ですかね」

伊藤 「これはどうでしょうか、効果がありそうですか？」

瀧澤 「まだバグってて、結果が出てないですけど、Apreyの平岡さんは4駒関係でドンドン強くなっているっぽって言っているの」

伊藤 「4駒関係になると、教師データのスパース性の問題も出てきますよね？」

瀧澤 「そうですね。それより先にメモリの消費量が激しいので、どこを持ってくるかっていうところですね」

伊藤 「その辺りはどうやって解決しようとしているんですか？」

瀧澤 「メモリの話は駒の数を絞ってとか、色んなアプローチがあるんですけど」

伊藤 「特定の駒に注目したりとか？」

瀧澤 「そうですね。特定の駒に絞ったり、王様の位置を下段だけにするとか、あとは駒を区別しないとかも良いかもしれないですね。あとは、正則化を効かせるような何か、ペナルティを効かせるようなものを入れるしかないのかな……。まだ、その段階まで行けるのか不安なんですけど」

伊藤 「ほかには何かありますか？」

瀧澤 「そうですね。今、コンピュータ自身で考えるように作ったんですけど、どっかで頭打ちになる。なんで頭打ちになるのかイマイチ仕組みがよく分かってなくて、そこをなんとかしていかないといけないだろうと考えています。なんか別な技術というか、まあ昔からやられていた技術ではあるんですけど、評価関数のパラメータをそれぞれ

別々で作ったやつを足して平均とると、なぜか強くなる。アンサンブル学習の1つなんですけど、それが普通に学習したのでは、たどり着かないような、まあ150点近く強くなるんです。それが、なんでこんなに効果があるのかが、分かんなくて……。推測することはできるんですけど、結局表現する空間って、そのパラメータの関数自体は変わっていないので、普通に学習してたどり着かずなんですけど、そこまでたどり着かないっていうのが、やっぱり何か問題があるんだろうということで、そこを何とかしたいっていうのがありませんね」

伊藤 「一度優勝してしまうと目標を見失うってことはないですか？ 開発のモチベーションは維持できそうでしょうか。1回の優勝ではそこまでモチベーションが下がることはないですか？」

瀧澤 「そうですね。僕は一から作っている人たちとは立ち位置が違うというか、今までの人たちは、一から作っている人たちだったので、僕はライブラリを使ってここだけ改良するっていう感じなので、ちょっとスタイルが違うんですけど。そうですね。やっぱり、開発し続けるというのは本当に大変だと思います」

伊藤 「電王戦も近い時期のお忙しい時期にお時間をいただいてありがとうございました」

瀧澤 「ありがとうございました」

■瀧澤 誠 inf232@gmail.com

2009年北海道大学大学院情報科学研究科メディアネットワーク専攻修士課程修了。現在、KDDI(株)勤務。趣味でコンピュータ将棋elmoを開発中。

■伊藤毅志(正会員) ito@cs.uec.ac.jp

1994年名古屋大学大学院工学研究科情報工学専攻博士課程修了(工学博士)。現在、電気通信大学情報理工学研究所助教。コンピュータ囲碁フォーラム副会長。思考ゲームを題材にした人間の思考過程、熟達化に興味を持つ。著書に『先を読む頭脳』(新潮社、共著)ほか。

^{☆4} 棋の評価関数を機械学習する際に、複数の駒の相対的な位置関係を調整すべき良い形のパラメータとして学習する手法がよく用いられる。前述のBonanzaが初めて導入した手法であるが、3駒間の関係を用いるのが主流であった。これを4駒関係にするというものである。