Reinforcement Learning
Paradigma baru dalam Machine Learning

Ali Ridho Barakbah, S.Kom.
Soft Computation Research Group, EEPIS-ITS

Reinforcement Learning. Itulah topik yang akan saya perbincangkan disini. Walaupun ini sekedar tulisan biasa, bukan tulisan ilmiah, namun setidaknya ini mungkin bisa dianggap sebagai sharing knowledge yang mungkin bisa bermanfaat bagi rekan-rekan disini.

KILAS BALIK

Berbicara tentang Reinforcement Learning (RL), tidak terlepas dari sejarah berkembangnya bidang Artificial Intelligence (AI). Kalau anda tidak berkeberatan, saya akan mengajak anda memutar jam dinding anda, lalu menerobos dimensi waktu, dan pergi ke awal tahun 1950-an.

Adalah seorang yang bernama Alan Turing, seorang matematikawan Inggris, di tahun 1950-an, mencoba membuat suatu mesin yang dinamakan Turing Machine dimana di dalamnya berisi game yang dibangun dari serangkaian algoritma sehingga mesin tersebut mampu bermain dengan manusia. Pada tahun 1956, John McCarthy, seorang professor dari MIT, mulai memperkenalkan bidang baru secara spesifik yang bernama Artificial Intelligence. Beliau mendefinisikan bidang itu sebagai "Bidang yang memodelkan proses-proses berpikir manusia dan mendesain mesin agar dapat menirukan kelakuan manusia" [1].

Mulailah dunia AI berkembang pesat, menjadi daya tarik tersendiri bagi para peneliti dan pakar computer science. Hal ini bisa kita lihat dengan bermunculnya berbagai macam metode-metode yang dikembangkan pada bidang AI, mulai dari teori graph, teori tree, teori state, knowledge based system, sampai expert system yang berbasis probabilistic model.

Menariknya, sebelum masa berkembangnya AI yang berorientasi pada pemodelan cara berpikir manusia, para scientist sebenarnya juga berusaha mengembangkan pemodelan cara berpikir manusia. Pada awal tahun 1940-an, mereka melakukan riset terhadap mekanisme berpikir pada struktur otak manusia. Itulah awal berkembangnya apa yang dinamakan dengan Neural Network (NN).

Yang lebih menarik lagi adalah, meskipun berorientasi pada pemodelan cara berpikir manusia, berkembang NN dan AI seakan-akan dalam dua dunia yang berbeda. Oleh karena itu, bisa kita maklumi ada perbedaan pendapat di kalangan pakar computer science apakah NN bisa dikategorikan sebagai salah satu bidang AI.

Sebagian expert menyakini bahwa NN termasuk dalam salah satu metode pada bidang AI. Sebagian expert lainnya mengatakan bahwa NN lebih cenderung masuk dalam bidang Soft Computation daripada masuk ke bidang AI. Ada juga sebagian expert yang mengawinkan kedua kutub perbedaan perdapat tersebut dengan mengatakan bahwa NN lebih cocok masuk dalam bidang Computational Intelligent. Namun saya tidak akan membahas terlalu mendalam mengenai hal ini.

AI DAN KONSEP LEARNING

Namun seiring dengan berjalannya waktu, mulai masuk tahun 1990-an, perkembangan AI sudah mulai menurun popularitasnya di kalangan scientist dibandingkan dengan perkembangan NN sendiri. Mengapa bisa demikian?. Saya bisa merangkainya demikian.

AI yang pada mulanya dianggap suatu bidang keilmuan yang mencoba memodelkan cara berpikir manusia, tidak lain banyak didominasi oleh teori-teori logis yang sebenarnya tidak ada hubungannya dengan cara manusia berpikir. Metode-metode yang berkembang pada bidang AI bukanlah bertumpu pada konsep learning yang merupakan dasar teori manusia itu bisa berpikir.

Ambil saja satu contoh, Expert System (ES). ES itu sebenarnya bukan merupakan cara manusia berpikir, tapi lebih mengarah kepada perkawinan antara teori tree dengan teori probabilistik untuk menyelesaikan permasalahan pengambilan keputusan. Terlalu berlebihan jika kita beranggapan bahwa ES terilhami oleh cara manusia berpikir, apalagi cara expert berpikir, sebagaimana kata expert yang dipinjam pada teori tersebut. Namun ini bukan berarti ES itu metode yang kurang baik. Akan tetapi saya hanya mengatakan bahwa ES bukanlah teori yang dibangun dari cara manusia berpikir.

Lebih menguatkan hal tersebut, saya sedikit ingin bercerita. Seperti yang kita ketahui bahwa ES mulai dikembangkan pada tahun 1960 oleh komunitas AI. Teori ini sangat ampuh dalam menyelesaikan permasalahan pengambilan keputusan melalui pendekatan rule based reasoning dan case based reasoning. Pada tahun 1971, Thomas L.Saaty [2] berhasil mengembangkan suatu metode baru dalam menyelesaikan permasalahan pengambilan keputusan. Metode tersebut bernama Analytical Hierarchy Process (AHP). Meskipun sama-sama metode yang dipakai untuk kasus pengambilan keputusan, di beberapa sisi AHP mempunyai kelebihan dibandingkan ES. Diantara kelebihan tersebut adalah AHP bisa melibatkan nilai preferensi, dimana hal itu tidak pernah dibayangkan oleh ES. Di sisi lain, AHP bisa melakukan koreksi kesalahan input yang mana hal itu tidak bisa dilakukan oleh ES. Lalu kenapa AHP yang sama-sama dipakai untuk penyelesaian kasus pengambilan keputusan sebagaimana ES dan di satu sisi mempunyai kelebihan dibandingkan ES, tidak dimasukkan ke dalam salah satu bidang di AI?. Jawabannya sederhana. Itu karena AHP bukan dikembangkan oleh seorang pakar dari komunitas AI, sehingga tidak pernah terbayang sama sekali untuk mengaitkan antara AHP dengan AI.

Cobalah anda lihat metode-metode lain yang masuk dalam bidang AI, maka akan anda temukan banyak metode yang sebenarnya bukan berasal dari memodelkan cara manusia berpikir. Kalaulah AI disebut kecerdasan buatan, maka belum bisa dikatakan kecerdasan

tersebut adalah berasal dari pemodelan cara berpikir manusia sebagaimana yang didefinisikan oleh McCarthy pertama kali, atau dengan kata lain, bukan human artificial intelligence. Namun sekali lagi, ini bukan berarti metode-metode di AI merupakan metode-metode yang tidak baik. Selama ini, metode-metode di AI banyak berhasil menyelesaikan permasalah-permasalahan yang kompleks dimana manusia sendiri merasa kesulitan untuk memecahkannya. AI banyak memberikan dasar-dasar logis dalam menyelesaikan berbagai masalah komputasi. AI bahkan merupakan pintu gerbang yang harus dimasuki untuk mengenal lebih jauh tentang berbagai disiplin ilmu pada computer science.

SEPUTAR LEARNING THEORIES

Itulah sebabnya, para scientist lebih tertarik dan mulai intens melakukan riset dalam bidang yang berbasis pada learning theory. Prof. Sugiyama [3] membagi bidang learning itu dalam 3 bidang riset:

1. Memahami konsep human brains (dikaji pada bidang physiology, psychology, neuroscience)
2. Mengembangkan learning machines (computer and electronic engineering)
3. Mentranformasi esensi learning secara matematik (computer and information science)

Karena NN berbasis pada learning theory, sehingga itulah sebabnya mengapa NN sangat menarik dan intens dikaji oleh para pakar computer science. Pada learning theory itu pula, para scientist mengelompokkan NN ke dalam salah satu tipe learning, yaitu supervised learning, di antara tipe yang lain, unsupervised learning.

Supervised learning diibaratkan sebagai proses belajar dari seorang murid yang berada dalam sebuah kelas. Si murid diperbolehkan bertanya kepada guru yang telah mengetahui aturan jawabannya, dan kemudian si dosen menjawab pertanyaan tersebut. Dari hasil tanya-jawab berkali-kali, si murid akan bisa memahami rule dari permasalahan, sehingga jika ada permasalahan lain, si murid akan membandingkan dengan rule yang ia simpulkan sebelumnya, sehingga ia bisa memberikan jawaban. Oleh karena itu, tipe supervised learning ini memerlukan apa yang disebut training. Semakin lama training, semakin pintar pula si murid memecahkan masalah. Itulah basic concept dari supervised learning.

Selain supervised learning, ada lagi tipe learning yang lain yang dinamakan dengan unsupervised learning. Ilustrasi yang mudah misalkan hubungan antara murid dan dosen pada contoh yang sebelumnya. Ketika si murid menjumpai masalah, ia harus dapat menjawab masalah tersebut dengan sendirinya. Semakin banyak ia berusaha menjawab sendiri, ia akan semakin pandai dalam menemukan rule yang dapat digunakan untuk memecahkan permasalahan di kemudian hari. Unsupervised learning ini akan sangat bermanfaat jika memang permasalahan yang dihadapi relatif tidak bisa atau sulit sekali dijawab oleh sang guru.

Berbeda dengan supervised learning, unsupervised learning tidak memerlukan proses training. Ketika si murid menjumpai masalah, ia harus dapat menjawab masalah tersebut dengan sendirinya. Semakin banyak ia berusaha menjawab sendiri, ia akan semakin pandai dalam menemukan rule yang dapat digunakan untuk memecahkan permasalahan di kemudian hari.

Tahun-tahun belakangan ini, para ilmuwan terus menggali konsep-konsep seputar learning theory, dan sampai akhirnya mereka menemukan tipe learning yang lain yang disebut dengan Reinforcement Learning.

REINFORCEMENT THEORY

Konsep dasar RL diambil dari suatu teori dalam ilmu psikologi yang disebut dengan Reinforcement Theory. Reinforcement Theory ini merupakan suatu pendekatan psikologi yang sangat penting bagi manusia.Teori ini menjelaskan bagaimana seseorang itu dapat menentukan, memilih dan mengambil keputusan dalam dinamika kehidupan. Teori ini bisa digunakan pada berbagai macam situasi yang seringkali dihadapi manusia.

Reinforcement Theory ini mengatakan bahwa tingkah laku manusia itu adalah merupakan hasil kompilasi dari pengalaman-pengalaman yang ia temui sebelumnya, atau dalam bahasa lainnya disebut "Consequences influence behavior".

Contoh yang paling mudah yang bisa saya gambarkan disini adalah bagaimana sikap yang diambil oleh seorang siswa di dalam kelas. Asumsikan bahwa sang guru sudah menjelaskan seperangkap aturan yang harus ditaati oleh siswa di dalam kelas. Suatu ketika, seorang siswa berteriak di dalam kelas. Maka sang guru langsung memberikan hukuman kepada siswa tersebut. Dari hukuman itu, siswa tadi akan merubah sikapnya untuk tidak berteriak lagi. Juga demikian, kepada siswa yang tekun mengikuti pelajaran di dalam kelas, maka sang guru memberikan kepada mereka semacam hadiah atau penghargaan. Jika sistem ini berjalan dalam jangka waktu tertentu, maka keadaan siswa tadi pasti akan konvergen untuk mengambil sikap yang baik di dalam kelas.

Dalam Reinforcement Theory, terdapat 3 konsekuensi yang berbeda, yaitu :

1. Konsekuensi yang memberikan reward
2. Konsekuensi yang memberikan punishment
3. Konsekuensi yang tidak memberikan apa-apa

Seorang siswa yang bersikap baik di dalam kelas, ia akan mendapatkan reward. Dengan reward itu, ia akan bersikap lebih baik lagi. Jika ia bersikap lebih baik lagi, ia akan mendapatkan reward lagi. Demikian seterusnya yang terjadi sehingga ia pasti akan semakin konvergen dalam bersikap baik di dalam kelas. Sebaliknya, jika ia bersikap buruk, maka ia akan menerima punishment. Dengan punishment itu, ia akan merubah sikapnya. Jika punishment itu tidak cukup untuk membuatnya berubah, maka ia akan mendapatkan punishment lagi, sehingga dalam batasan tertentu, ia pasti akan berubah sikap yang hasilnya adalah ia akan mendapatkan reward. Demikian seterusnya, sehingga pada suatu saat nanti, ia akan konvergen bersikap baik di dalam kelas.

Ini adalah teori yang luar biasa dalam menjelaskan dynamic system pada real system. Akan tetapi, sangat sulit sekali untuk memodelkan dan mentransformasikannya dalam bentuk computational system.

Coba bayangkan pada kasus diatas. Seandainya saja siswa tersebut berteriak dan ia mendapatkan punishment, maka bisa jadi punishment itu tidak berpengaruh pada dirinya. Atau sebaliknya, punishment itu sangat berpengaruh pada dirinya, sehingga ia menjadi sangat malu, dan akhirnya bunuh diri!. Juga demikian dengan bagaimana memodelkan bentuk konsekuensi yang tepat, baik dari segi kategori konsekuensi maupun dari segi intensitas konsekuensi. Kesulitan yang lainnya adalah bagaimana memodelkan sistem yang dinamik dalam aturan-aturan Reinforcement Theory.

Sehingga bisa diambil kesimpulan bahwa Reinforcement Theory itu bukan merupakan teori yang sederhana, akan tetapi merupakan teori yang sangat kompleks yang benar-benar dapat menjelaskan keadaan dynamic system pada real system. Jika saja teori ini dapat dimodelkan dan ditransformasikan dalam bentuk computational system, maka akan terjadi perubahan yang luar biasa pada computational learning theory.

REINFORCEMENT LEARNING DALAM LINTASAN MASA

Berkembangnya teori yang berbasis pada Reinforcement Learning diawali dengan munculnya prinsip psikologi klasik yang dinyatakan oleh Thorndike di dalam teorinya yang dikenal dengan “Law of Effect”pada tahun 1911. Dalam teorinya beliau menyatakan,

"Of several responses made to the same situation, those which are accompanied or closely followed by saticfaction to the animal will, other things being equal, be more firmly connected with situation, so that, when it recurs, they will be more likely to recur; those which are accompanied or closely followed by discomfort to the animal will, other things being equal, have their connections with that situation weakened, so that, when it recurs, they will be less likely to occur. The greater the satisfaction or discomfort, the greater the strengthening or weaking of the bond."

Meskipun teori itu menimbulkan kontroversi di kalangan psikologi, namun teori tersebut banyak mempengaruhi munculnya berbagai teori yang menghubungkan antara behaviour dan enviroment.

Law of Effect mulai pertama kalinya diaplikasikan dalam computational field pada tahun 1954 oleh Minsky. Dalam disertasi PhD-nya, beliau membuat suatu analog machine yang disebut SNARC (Stochastic Neural-Analog Reinforcement Calculator) yang bekerja dengan prinsip learning melalui trial and error. Tahun 1960-an, Donald Michie membuat suatu program yang disebut MENACE (for Matchbox Educable Noughts and Crosses Engine) yang dapat bermain Tic-Tac-Toe dengan mengaplikasikan Reinforcement Learning yang sederhana. Tahun 1963, Andrea membuat suatu reinforcement learning machine yang disebut STeLLA. Pada tahun 1968, Michie dan Chambers menyempurnakan MENACE dengan mengaplikasikan Reinforcement Learning yang lebih advanced dan menamakan programnya dengan

GLEE (Game Learning Expectimaxing Engine). Demikian seterusnya Reinforcement Learning berkembang dari tahun ke tahun.

Hanya saja, mungkin di benak kita timbul pertanyaan, "Mengapa Reinforcement Learning tidak begitu terkenal dibandingkan teori-teori learning lainnya ataupun AI?." Saya menjawab, "Ya, memang benar."

Sejak tahun 1960-an, teori Reinforcement Learning secara perlahan-lahan tertutupi dengan berkembangnya teori-teori AI dan supervised learning, salah satunya Neural Network. Saat itu munculnya teori-teori AI dan supervised learning disambut sebagai teori-teori yang sangat menjanjikan dalam hal mentransformasi human brain. Tak ayal lagi, para scientist mengalihkan pandangan dan memutar konsentrasi mereka untuk menekuni dan melakukan riset pada teori-teori tersebut. Inilah yang menyebabkan semakin berkembangnya teori-teori AI dan supervised learning pada periode masa itu. Setelah sekian lama bertahun-tahun melakukan riset, sampailah pada tahun 1990-an, para scientist akhirnya dapat mengambil kesimpulan terhadap teori-teori yang mereka pelajari. Teori-teori yang pada awalnya mereka yakini sebagai teori-teori learning yang dapat mengarah untuk bisa mengembangkan suatu learning machine (mesin/program yang dapat belajar), tidak lain cuman berhenti pada learned machine (mesin/program yang diajari), suatu machine yang pintar setelah diajari, bukan machine yang pintar setelah belajar.

Hal itulah juga yang bisa menjawab mengapa Reinforcement Learning justru mulai muncul kembali pada awal 1990-an. Memang tak aneh, karena Reinforcement Learning merupakan suatu fenomena teori yang berangkat dari, "Bagaimana membuat suatu machine itu dapat menjadi pintar setelah berinteraksi dengan environment?". Semakin banyak berinteraksi, maka semakin pintarlah machine tersebut. Sekarang saya-lah yang akan balik bertanya kepada anda, "Bukankah ini adalah the real human artificial intelligent?"

TEORI DASAR

Reinforcement Learning adalah salah satu paradigma baru di dalam learning theory. RL dibangun dari proses mapping (pemetaan) dari situasi yang ada di environment (states) ke bentuk aksi (behavior) agar dapat memaksimalkan reward. Agent yang bertindak sebagai sang learner tidak perlu diberitahukan behavior apakah yang akan sepatutnya dilakukan, atau dengan kata lain, biarlah sang learner belajar sendiri dari pengalamannya. Ketika ia melakukan sesuatu yang benar berdasarkan rule yang kita tentukan, ia akan mendapatkan reward, dan begitu juga sebaliknya.

RL secara umum terdiri dari 4 komponen dasar, yaitu :

1. Policy : kebijaksanaan
2. Reward function
3. Value function
4. Model of environment

Policy adalah fungsi untuk membuat keputusan dari agent yang menspesifikasikan tindakan apakah yang mungkin dilakukan dalam berbagai situasi yang ia jumpai. Policy inilah yang bertugas memetakan perceived states ke dalam bentuk aksi. Policy bisa berupa fungsi sederhana, atau lookup table. Policy ini merupakan inti dari RL yang sangat menentukan behavior dari suatu agent.

Reward function mendefinisikan tujuan dari kasus atau problem yang dihadapi. Ia mendefinisikan reward and punishment yang diterima agent saat ia berinteraksi dengan environment. Tujuan utama dari reward function ini adalah memaksimalkan total reward pada kurun waktu tertentu setelah agent itu berinteraksi.

Value function menspesifikasikan fungsi akumulasi dari total reward yang didapatkan oleh agent. Jika reward function berbicara pada masing-masing partial time dari proses interaksi, value function berbicara pada long-term dari proses interaksi.

Model of environment adalah sesuatu yang menggambarkan behavior dari environment. Model of environment ini sangat berguna untuk mendesain dan merencanakan behavior yang tepat pada situasi mendatang yang memungkinkan sebelum agent sendiri mempunyai pengalaman dengan situasi itu. Saat masa-masa awal RL dikembangkan, model of environment yang ada berupa trial and error. Namun modern RL sekarang sudah mulai menjajaki spektrum dari low-level, trial and error menuju high-level, deliberative planning.

EXPLOITATION AND EXPLORATION

Salah satu keunggulan Reinforcement Learning dibandingkan teori-teori learning yang lain adalah kemampuannya dalam mengadopsi proses exploitation dan exploration yang memang biasanya dilakukan oleh human being. Exploitation dan exploration inilah yang menjadi kunci keberhasilan proses learning dari RL.

Seringkali manusia itu mengambil keputusan untuk melakukan sesuatu dengan berdasarkan pada informasi yang ia terima sebelumnya daripada perbuatan-perbuatan yang ia lakukan di masa lalu. Proses menggali sebanyak mungkin informasi tersebut dinamakan dengan exploitation.

Namun seringkali juga manusia itu mengambil keputusan untuk melakukan sesuatu dengan tidak berdasarkan pada informasi yang ia terima sebelumnya daripada perbuatan-perbuatan yang ia lakukan di masa lalu, akan tetapi lebih cenderung ia mencoba melakukan sesuatu yang memang benar-benar baru bagi dirinya untuk melihat bagaimana hasil daripada perbuatan tersebut. Proses inilah yang disebut dengan exploration.

Seseorang yang exploitation-nya relatif besar akan cenderung bertindak over-pasive dan ekstra hati-hati, bahkan mungkin dia tidak akan berani melakukan sesuatu apapun. Ini disebabkan ia hanya menggali informasi yang ia terima sebelumnya daripada perbuatan-perbuatan yang ia lakukan di masa lalu. Jika ia berhadapan dengan suatu keadaan

dimana hal itu belum pernah ia alami sebelumnya, maka ia akan cenderung tidak berbuat apa-apa.

Namun sebaliknya, jika seseorang yang exploration-nya relatif besar, maka ia akan cenderung bertindak over-active dan nekad. Orang yang seperti ini termasuk tipe orang yang tidak belajar dari pengalaman-pengalaman yang ia dapatkan sebelumnya. Akibatnya, tindakan apapun yang ia lakukan merupakan tindakan yang mengandung tingkat probabilistik yang sangat besar, atau dengan kata lain 'gambling'.

Keseimbangan antara exploitation dan exploration inilah yang menjadi kunci keberhasilan proses learning dalam kehidupan manusia. Yang perlu kita garis bawahi, seimbang bukan berarti sama, atau dengan kata lain fifty-fifty, akan tetapi prosentase keduanya akan berfluktuasi sesuai dengan berbagai macam keadaan yang jelas sangat beragam.

Nah, Reinforcement Learning mengadopsi konsep exploitation dan exploration yang ada pada human being. Yup!, satu nilai tambah lagi buat RL sebagai human artificial intelligence. Lalu seperti apakah bentuk exploitation dan exploration yang ada pada RL, ikutilah seri-seri tulisan ini berikutnya.

EVALUATIVE FEEDBACK

Sesuatu yang paling penting yang membedakan antara Reinforcement Learning dengan tipe-tipe learning lainnya adalah penggunaan evaluasi aksi yang telah diambil lebih daripada memberikan instruksi aksi manakah yang seharusnya dilakukan. Proses evaluasi ini membuka perlu adanya exploration secara aktif, dengan mencoba trial and error untuk menemukan behavior yang baik. Evaluative feedback mengindikasikan bagaimana sebaiknya aksi itu diambil, tetapi bukan menentukan kemungkinan apakah itu aksi yang terbaik atau terburuk.

Evaluative feedback merupakan basis berbagai metode pada permasalahan optimasi, termasuk juga metode-metode evolutionary. Evaluative feedback juga merupakan basis dari supervised learning yang seringkali berbicara tentang pattern recognition, artificial neural network dan system identification. Begitulah yang dipaparkan oleh Sutton dan Barto dalam bukunya yang terkenal "Reinforcement learning: an introduction" mengawali pembahasan tentang evaluative feedback.

Evaluative feedback dalam RL memuat pembahasan yang sangat luas, namun kali ini saya hanya mengenalkan yang sederhana agar bisa memberikan gambaran yang mudah anda pahami. Jika seseorang melakukan sesuatu untuk mencapai goal tertentu (katakanlah a), dan setiap kali ia melakukan hal tersebut, maka ia akan mendapatkan reward Ri dimana i adalah waktu dimana seseorang itu melakukan sesuatu. Jika aksi yang ia lakukan relatif mendekati a, maka ia akan mendapatkan reward yang besar, dan demikian pula sebaliknya. Yang jelas, ketika ia sudah melakukan beberapa aksi, maka ia akan mendapatkan rata-rata dari reward-reward yang ia terima, sehingga action value Qt(a) yang ia dapatkan adalah:

$Qt(a) = (R1 + R2 + \ldots + Rn) / n$

dimana t adalah fungsi waktu yang ia perlukan untuk melakukan n aksi. Ini berarti bahwa Qo(a)=0, sebab ia belum melakukan sesuatu apapun pada fungsi waktu t=0. Jika Q*(a) adalah action value yang ideal untuk mencapai goal a, yang berarti bahwa jika ia berusaha melakukan sebaik-baiknya aksi, maka Qt(a) akan mendekati Q*(a). Inilah yang dinamakan dengan metode sample-average untuk mengestimasi action value. Namun ini hanyalah metode yang paling sederhana dan bukan yang terbaik untuk mengestimasi action value.

Lalu apakah itu berarti kita harus menyimpan semua data-data Ri untuk mendapatkan Qt(a)?. Kalau saja hal ini dilakukan, pasti ini akan membutuhkan komputasi yang besar. Hal ini disebabkan karena RL biasanya dipakai untuk dynamic system sehingga t biasanya bernilai besar. Lalu bagaimana menghindari hal tersebut, RL menyelesaikannya dengan incremental implementation. Mau tahu, ikutilah dalam tulisan berikut…

INCREMENTAL IMPLEMENTATION

Incremental implementation yang akan bisa menjawab pertanyaan yang lalu tentang apakah itu berarti kita harus menyimpan semua data-data Ri untuk mendapatkan Qt(a)?. Hal itu bisa dihindari dengan cara menyederhanakan persamaan sebelumnya sebagai berikut.

$Q(k+1) = (\sum R(i)) / (k+1)$ dimana $i=1..k+1$

$= [R(k+1) + \sum R(i)\ ] / (k+1)$ dimana $i=1..k$
$= [R(k+1) + k \,.Q(k) + Q(k) - Q(k)] / (k+1)$
$= [R(k+1) + (k+1) \,.\, Q(k) - Q(k)] / (k+1)$
$= Q(k) + \{[R(k+1) - Q(k)] / (k+1)\}$

Dengan demikian, implementasi action value tersebut tidak membutuhkan banyak memori. Dalam bahasa manusia, persamaan diatas dapat disederhanakan sebagai berikut:

NewEstimate ← OldEstimate + StepSize [Target – OldEstimate]

Ekspresi [Target–OldEstimate] pada persamaan diatas merupakan besaran selisih reward untuk proses estimasi terhadap aksi yang dilakukan. Ketika Target ternyata memberikan reward, maka [Target–OldEstimate] akan menjadi positif, yang kemudian mengakibatkan nilai NewEstimate akan menjadi lebih besar daripada OldEstimate. Tapi sebaliknya, ketika Target ternyata memberikan punishment, maka [Target–OldEstimate] akan menjadi negatif, yang kemudian mengakibatkan nilai NewEstimate akan menjadi lebih kecil daripada OldEstimate.

Sedangkan StepSize merupakan besaran untuk mengatur seberapa besar agent melakukan exploitation atau exploration. Semakin besar nilai StepSize, dengan batasan

maksimum 1, maka semakin besar kemungkinan agent untuk melakukan exploitation. Begitu juga sebaliknya, semakin kecil nilai StepSize, dengan batasan minimum 0, maka semakin besar pula kemungkin agent untuk melakukan exploration.

Pada saat nilai StepSize diset besar, maka ini akan menyebabkan besaran reward (baik reward ataupun punishment) menjadi sangat besar, sehingga dampaknya akan berpengaruh pada nilai NewEstimate yang didapatkan mempunyai selisih yang sangat besar dengan OldEstimate. Karena selisih yang sangat besar tersebut, sehingga dalam proses aksi selanjutnya, agent akan lebih besar kemungkinannya melakukan proses exploitation.

Tapi sebaliknya, pada saat nilai StepSize diset kecil, maka ini akan menyebabkan besaran reward (baik reward ataupun punishment) menjadi sangat kecil, sehingga dampaknya akan berpengaruh pada nilai NewEstimate yang didapatkan mempunyai selisih yang sangat kecil dengan OldEstimate. Karena selisih yang sangat kecil tersebut, sehingga dalam proses aksi selanjutnya, agent akan lebih besar kemungkinannya melakukan proses exploration.

REFERENSI:

Sri Kusumadewi, *Artificial Intelligence (Teknik dan Aplikasinya)*, edisi I, penerbit Graha Ilmu, Yogya, 2003.

Sri Mulyani, *Riset Operasi*, Lembaga Penerbit Fakultas Ekonomi UI, Jakarta, 2002.

Masashi Sugiyama, *Pattern Information Processing*, Department of Computer Science, Tokyo Institute of Technology, Japan.

R.S. Sutton, A. Barto, *Reinforcement Learning: an Introduction*, second printing, 1999, The MIT Press.

S. Booth, Reinforcement Theory, http://www.as.wvu.edu/~sbb/comm221/chapters/rf.htm, 1999.

M.E. Harmon, S.S. Harmon, Reinforcement Learning: a Tutorial, http://citeseer.nj.nec.com/harmon96reinforcement.html, 1996.

D. Finton, What is Reinforcement Learning, http://www.cs.wisc.edu/~finton/what-rl.html.
S. Mahadevan, Glosarry of Terminology in Reinforcement Learning, http://www-anw.cs.umass.edu/rlr/terms.html.

Y. Mansour, Lecture Notes on Reinforcement Learning, http://www.math.tau.ac.il/~mansour/rl.html.

S.T. Hagen, B. Kröse, A Short Introduction to Reinforcement Learning, http://citeseer.ist.psu.edu/tenhagen97short.html, 1997.

S. Keerthi, B. Ravindran, A Tutorial Survey of Reinforcement Learning, http://citeseer.ist.psu.edu/keerthi95tutorial.html, 1995.