

Nama: Loaeza Septavial

NIM: 1103204003

Catatan Backpropagation

Backpropagation adalah metode menghitung gradien fungsi kesalahan sehubungan dengan bobot jaringan saraf. Ini adalah generalisasi aturan delta untuk perceptron ke jaringan neural feedforward multilapis. fungsi kesalahan. Mengingat jaringan saraf tiruan dan penurunan gradien menggunakan jaringan saraf tiruan, singkatan dari "backpropagation of error," adalah algoritme untuk pembelajaran terawasi

Yang "mundur" sebagian dari namanya berasal dari fakta bahwa penghitungan gradien dilakukan secara mundur melalui jaringan, dengan gradien lapisan bobot terakhir dihitung terlebih dahulu dan gradien lapisan bobot pertama dihitung terakhir. Perhitungan parsial gradien dari satu lapisan digunakan kembali dalam perhitungan gradien untuk lapisan sebelumnya. Aliran informasi kesalahan yang terbalik ini memungkinkan penghitungan gradien pada setiap lapisan secara efisien dibandingkan pendekatan naif dalam menghitung gradien setiap lapisan secara terpisah.

Popularitas propagasi mundur telah mengalami kebangkitan baru-baru ini mengingat meluasnya adopsi jaringan saraf dalam untuk pengenalan gambar dan pengenalan suara. Ini dianggap sebagai algoritma yang efisien, dan implementasi modern memanfaatkan GPU khusus untuk lebih meningkatkan kinerja.

Setelah mendapatkan nilai error, kita bisa mulai memperbaiki JST kita dengan backpropagation. Sebenarnya istilah memperbaiki JST ini kurang tepat jika menyebutnya Backpropagation, lebih tepatnya adalah Gradient Descent. Tapi ya di Indonesia lebih umum menyebutnya Backpropagation kadang disingkat Backprop.

Rumus utama untuk memperbaiki suatu bobot berdasarkan error E adalah:

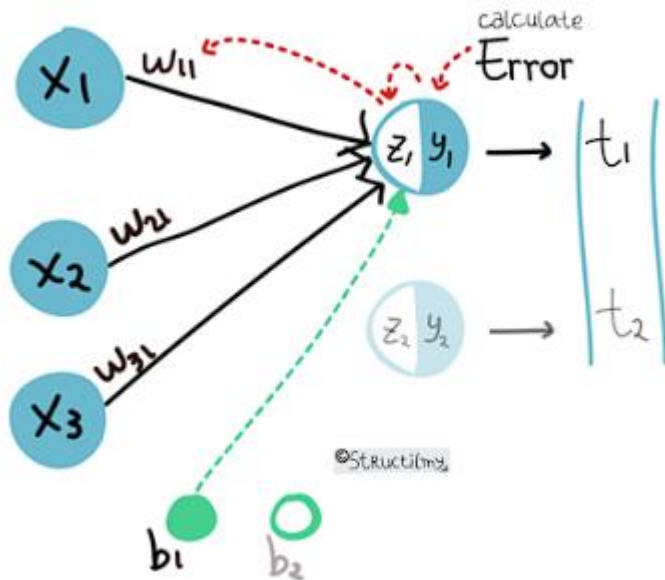
$$w_{new} = w_{old} - \alpha \frac{\partial E}{\partial w}$$

Rumus ini juga berlaku untuk memperbaiki nilai bias:

$$b_{new} = b_{old} - \alpha \frac{\partial E}{\partial b}$$

KONSEP CHAINING

Untuk menghitung $\partial E / \partial w$, pertama-tama kita coba berjalan mundur dulu. Urutan proses backward chaining digambarkan dengan garis merah pada gambar di bawah ini. Dari mana nilai E didapatkan dan apa hubungannya dengan w_{11} .



Nilai E diperoleh dari rumus Mean Square Error:

$$E = \frac{1}{2} ((t_1 - y_1)^2 + (t_2 - y_2)^2)$$

Dari rumus di atas tidak ada variabel w_{11} tetapi kita bisa coba “jalan mundur” lagi. Kita ingat-ingat lagi dari mana nilai setiap variabel y berasal.

$$y_1 = \sigma(z_1)$$

$$y_2 = \sigma(z_2)$$

Variabel y diperoleh dari menerapkan sebuah fungsi aktivasi terhadap variabel z . Sedangkan, variabel z sendiri dihitung dengan:

$$z_1 = w_{11}x_1 + w_{21}x_2 + w_{31}x_3 + b_1$$

$$z_2 = w_{12}x_1 + w_{22}x_2 + w_{32}x_3 + b_2$$

Dari sini terlihat variabel w_{11} ada di perhitungan z_1 yang secara tidak langsung berpengaruh ke nilai E . Hal ini yang disebut dengan chaining atau rantai.

Link Referensi:

1. <https://brilliant.org/wiki/backpropagation/#:~:text=Backpropagation%2C%20short%20for%20%22backward%20propagation,to%20the%20neural%20network's%20weights.>
2. <https://structilmy.com/blog/2019/07/31/contoh-perhitungan-algoritma-backpropagation/>