

Dasar-Dasar Pembelajaran Mesin (Foundations of Machine Learning)

Aldi Cahya Muhammad, B.Sc.Engg., M.Sc.Engg.

Ir. Anak Agung Gde Bagus Ariana, S.T., M.T.

Indo Intan, S.T., M.T.

Sumanto, M.Kom.

Pastima Simanjuntak, S.Kom., M.SI.

Satria, S.Kom., M.Kom.

M.Lutfi MA, S.T., M.Kom.

Nono Heryana, M.Kom.

Harrizki Arie Pradana, S.Kom., M.T., MTA.



Editor : Dr. Safrizal, S.T., M.M., M.Kom.

**DASAR - DASAR
PEMBELAJARAN MESIN
*(FOUNDATIONS OF
MACHINE LEARNING)***

Aldi Cahya Muhammad, B.Sc.Engg., M.Sc.Engg.

Ir. Anak Agung Gde Bagus Ariana, S.T., M.T.

Indo Intan, S.T., M.T.

Sumanto, M.Kom.

Pastima Simanjuntak, S.Kom., M.SI.

Satria, S.Kom., M.Kom.

M. Lutfi MA, S.T., M.Kom.

Nono Heryana, M.Kom.

Harrizki Arie Pradana, S.Kom., M.T., MTA.

DASAR - DASAR PEMBELAJARAN MESIN (FOUNDATIONS OF MACHINE LEARNING)

Penulis:

Aldi Cahya Muhammad, B.Sc.Engg., M.Sc.Engg.
Ir. Anak Agung Gde Bagus Ariana, S.T., M.T.
Indo Intan, S.T., M.T.
Sumanto, M.Kom.
Pastima Simanjuntak, S.Kom., M.SI.
Satria, S.Kom., M.Kom.
M. Lutfi MA, S.T., M.Kom.
Nono Heryana, M.Kom.
Harrizki Arie Pradana, S.Kom., M.T., MTA.

Editor : **Dr. Syafrizal, S.T., M.M., M.Kom.**
Tata Letak : **Asep Nugraha, S.Hum**
Desain Cover : **Septimike Yourintan Mutiara, S.Gz.**
Ukuran : **UNESCO 15,5 x 23 cm**
Halaman : **vii, 131**
ISBN : **978-623-09-2522-1**
Terbit Pada : **Maret 2023**
Anggota IKAPI : **No. 073/BANTEN/2023**

Hak Cipta 2023 @ Sada Kurnia Pustaka dan Penulis

Hak cipta dilindungi undang-undang dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa izin tertulis dari penerbit dan penulis.

PENERBIT PT SADA KURNIA PUSTAKA

Jl. Warung Selikur Km.6 Sukajaya – Carenang, Kab. Serang Banten
Email : sadapenerbit@gmail.com
Website : sadapenerbit.com & repository.sadapenerbit.com
Telpon/WA : +62 838 1281 8431

KATA PENGANTAR

Segala Puji dan syukur diucapkan kehadirat Tuhan Yang Maha Esa, Atas Rahmat dan Karunia-Nya penulis telah menyelesaikan buku Dasar-dasar Pembelajaran Mesin (*Foundations of Machine Learning*).

Istilah *Machine Learning* mengacu pada deteksi otomatis pola yang bermakna dalam data. Beberapa dekade terakhir ini telah menjadi suatu hal yang umum di hampir semua lini kehidupan yang membutuhkan ekstraksi informasi dari kumpulan data besar. *Machine Learning* banyak digunakan dalam aplikasi ilmiah seperti bioinformatika, kedokteran, dan astronomi. Buku Dasar - Dasar Pembelajaran Mesin (*Foundations of Machine Learning*) ini mencakup perihal terkait pengenalan *Machine Learning*, Kerangka Pembelajaran PAC (*The PAC Learning Framework*), *Support Vector Machine*, Metode Kernel, Pembelajaran Online, Ranking, Regresi, Model Entropi Maksimum Bersyarat, dan *Reinforcement Learning*. Semoga kehadiran buku ini mampu untuk menambah wawasan dan ilmu pengetahuan terkait *Machine Learning* serta berkontribusi terhadap ilmu pengetahuan.

Penulisan buku ini berhasil diselesaikan atas kerjasama tim penulis. Penulis ucapkan terima kasih kepada semua pihak yang telah mendukung dalam penyelesaian buku ini. Semoga buku ini dapat bermanfaat.

Tim Penulis

DAFTAR ISI

KATA PENGANTAR	iii
DAFTAR ISI	iv
BAB 1 PENGENALAN <i>MACHINE LEARNING</i>	1
Definisi <i>Machine Learning</i>	1
Jenis-jenis Algoritma.....	4
Aplikasi <i>Machine Learning</i>	8
Teknik <i>Preprocesssing</i> Data.....	10
Daftar Pustaka.....	11
Profil Penulis.....	16
BAB 2 KERANGKA PEMBELAJARAN PAC (<i>THE PAC LEARNING FRAMEWORK</i>)	17
Pengantar.....	17
Definisi	17
Sejarah.....	19
Terminologi	19
Studi Kasus.....	20
<i>Generalization Error dan Empirical Error</i>	23
<i>Overfitting dan Underfitting</i>	24
Perhitungan <i>PAC</i>	26
Daftar Pustaka.....	29
Profil Penulis.....	30
BAB 3 <i>SUPPORT VECTOR MACHINE</i>	31
Pendahuluan	31
Struktur <i>Support Vector Machine</i>	32

<i>Hyperplane dan Margin Hyperplane</i>	33
<i>Maximal Margin Classifier (Pengklasifikasi batas Maksimal)</i>	34
<i>Cara Memilih Hyperplane</i>	35
<i>Margin</i>	35
<i>Support Vector Hyperplane</i>	39
<i>Model Support Vector Machine SVM</i>	40
<i>Slack Variable</i>	42
<i>Objective Function</i>	43
<i>Gradient Descent</i>	44
<i>Jenis-jenis Kernel</i>	45
<i>SVM Non-Linier</i>	46
<i>Contoh Implementasi Python Menggunakan Metode SVM Linier-Soft Margin Classifier</i>	48
<i>Daftar Pustaka</i>	51
<i>Profil Penulis</i>	54
BAB 4 METODE KERNEL	55
<i>Apa itu Kernel?</i>	55
<i>Kernel Dalam Machine Learning</i>	55
<i>Linear Kernel</i>	56
<i>Polynomial Kernel</i>	57
<i>Convolution Kernel</i>	58
<i>String Kernel</i>	59
<i>Graph Kernel</i>	60
<i>Daftar Pustaka</i>	61
<i>Profil Penulis</i>	62
BAB 5 PEMBELAJARAN ONLINE	63
<i>Pengertian Pembelajaran Online</i>	63
<i>Konsep Pembelajaran Online</i>	65

Metode Pola Pembelajaran	66
Analisis Kebutuhan Pembelajaran <i>Online</i>	67
Ciri-ciri Pembelajaran <i>Online</i>	71
Daftar Pustaka	74
Profil Penulis	77
BAB 6 RANKING	78
Definisi Ranking	78
Model Ranking	80
Matrix Evaluasi Peringkat	81
Model Pembelajaran Mesin Untuk Peringkat/Ranking	82
Daftar Metode Algoritma <i>Learning-To-Rank</i>	86
Kesimpulan	88
Daftar Pustaka	89
Profil Penulis	91
BAB 7 REGRESI	92
Pengantar Pembelajaran Mesin	92
Pengantar <i>Regresi</i>	94
Metode <i>Regresi</i>	98
Implementasi Analisis <i>Regresi</i>	101
Daftar Pustaka	103
Profil Penulis	105
BAB 8 MODEL ENTROPI MAKSIMUM BERSYARAT	106
Pengantar	106
Penerapan Model <i>Entropi</i> Maksimum Bersyarat dalam Bidang <i>Machine Learning</i>	107
Cara Memilih Model <i>Entropi</i> Maksimum Bersyarat yang Tepat	108
Kelebihan dan Keterbatasan	109

Langkah-Langkah & Algoritma untuk Mengimplementasikan Model Entropi Maksimum Bersyarat.....	111
Contoh Kasus Penerapan Model <i>Entropi</i> Maksimum Bersyarat	112
Daftar Pustaka.....	117
Profil Penulis.....	119
BAB 9 REINFORCEMENT LEARNING.....	120
Pendahuluan	120
Karakteristik <i>Reinforcement Learning</i>	122
Pendekatan Pada <i>Reinforcement Learning</i>	123
Komponen-Komponen Pada <i>Reinforcement Learning</i>	123
Jenis-Jenis <i>Reinforcement Learning</i>	125
Kelebihan Perhitungan <i>Reinforcement Learning</i>	125
<i>Eksplorasi dan Eksploitasi</i>	127
<i>Multiagent Reinforcement Learning</i>	127
Daftar Pustaka.....	128
Profil Penulis.....	131

BAB 1 PENGENALAN *MACHINE LEARNING*

Aldi Cahya Muhammad, B.Sc. Engg., M.Sc. Engg.

Business Development at Radiant Group

Renewable Energy Engineer

Definisi *Machine Learning*

Menurut buku "*An Introduction to Statistical Learning*" oleh Gareth James, Daniela Witten, Trevor Hastie, dan Robert Tibshirani (2013), "*Machine learning* adalah proses membuat komputer belajar dari data tanpa diberikan instruksi yang jelas." Ini berarti bahwa sistem *Machine Learning* mempelajari dari data dan membuat generalisasi sendiri tentang bagaimana mengatasi masalah, tanpa diberikan instruksi yang spesifik oleh pemrogram.

Perbedaan utama antara *Machine Learning* dan pemrograman konvensional adalah bahwa pemrograman konvensional memerlukan pemrogram untuk menentukan secara *eksplisit* bagaimana sistem harus bereaksi terhadap masukan, sementara *Machine Learning* memperbolehkan sistem mempelajari dan membuat generalisasi sendiri dari data." Ini berarti bahwa pemrogram harus menulis kode secara manual untuk setiap kemungkinan masukan dalam pemrograman konvensional, sementara sistem *Machine Learning* dapat mempelajari dan membuat generalisasi sendiri dari data yang diterimanya. Pemahaman terhadap suatu sistem harus dipahami dengan struktur sistem secara komprehensif. Bahwa hubungan suatu komponen sifatnya tidak harus linear namun juga bersifat *circular* dan saling mengunci (Alfaris et,al., 2022).

Kelebihan utama dari *Machine Learning* adalah kemampuannya untuk mengatasi tugas yang sangat kompleks dan memproses jumlah

Pengenalan *Machine Learning*

Machine Learning digunakan untuk menganalisis data dan mengidentifikasi ancaman keamanan yang mungkin terjadi.

Secara keseluruhan, *Machine Learning* dapat digunakan dalam berbagai bidang dan aplikasi untuk membuat keputusan yang lebih akurat dan efektif berdasarkan data yang tersedia. Dalam beberapa tahun ke depan, *Machine Learning* diharapkan dapat terus berkembang dan membantu dalam penyelesaian masalah-masalah yang lebih kompleks dan bermanfaat bagi masyarakat.

Teknik *Preprocessing* Data

Teknik *preprocessing* data adalah proses untuk mempersiapkan data sebelum dilakukan pemodelan *Machine Learning*. Tujuannya adalah untuk meningkatkan kualitas data, mengurangi *noise*, *outlier*, dan memperbaiki kekurangan dalam data yang mungkin mempengaruhi akurasi hasil pemodelan.

Dalam *Machine Learning*, teknik *preprocessing* data sangat penting untuk mempersiapkan data sebelum dilakukan pemodelan. Beberapa teknik *preprocessing* data yang umum digunakan antara lain *data cleaning*, *data transformation*, *feature scaling*, *feature selection*, *handling missing value*, *handling outlier*, *data integration*, dan *data discretization*. Dengan melakukan teknik *preprocessing* data ini, diharapkan akan meningkatkan kualitas data, mengurangi *noise* dan *outlier*, serta memperbaiki kekurangan dalam data yang mungkin mempengaruhi akurasi hasil pemodelan. Hal ini akan berdampak pada keberhasilan dan akurasi hasil pemodelan *Machine Learning* dan dapat membantu dalam pengambilan keputusan yang lebih tepat berdasarkan hasil pemodelan tersebut.

Semua teknik *preprocessing* data ini sangat penting untuk dipahami dan dilakukan sebelum melakukan pemodelan *Machine Learning*. Hal ini dapat mempengaruhi akurasi dan keberhasilan dari hasil pemodelan, serta meminimalkan kesalahan dalam pengambilan keputusan berdasarkan hasil pemodelan.

Daftar Pustaka

- Alfaris, L. et.al. (2022). *Riset Operasi*. Bandung: Indie Press.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (1st ed.)*. New York: Springer
- Grus, J. (2015). *Data Science from Scratch: First Principles with Python*. Shelter Island: O'Reilly Media.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.)*. Stanford: Stanford University.
- Jain, A. K., & Dubes, R. C. (1988). *Algorithms for Clustering Data*. New Jersey: Prentice-Hall, Inc.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. New York: Springer.
- Mitchell, T. M. (1997). *Machine Learning*. New York: McGraw-Hill. Mkitch
- Subakti, H. et.al. (2022). *Artificial Intelligence*. Tangerang : Media Sains Indonesia.
- Simon, P. (2013). *Too Big to Ignore: The Business Case for Big Data*. New Jersey: JohnWiley & Sons.
- Witten, I. H., Frank, E., & Mark, A. (2011). *Data Mining: Practical machine learning tools and techniques*. Elsevier.

PROFIL PENULIS



Aldi Cahya Muhammad, B.Sc. Engg., M.Sc. Engg.

Setelah lulus dari SMAN 1 Genteng Banyuwangi tahun 2013. Penulis lulus program *Bachelors* dan *Masters of Science in Electrical and Electronic Engineering* di salah satu Universitas Organisasi Kerjasama Islam (OKI), *Islamic University of Technology (IUT)*, Dhaka, Bangladesh di jurusan *Electrical and Electronic Engineering* lulus tahun 2019.

Karir pertama dimulai sebagai Bangladesh *Business Manager* di salah satu perusahaan Jepang, A-Wing Group. Setelah itu melanjutkan karir di perusahaan Radiant Group, Jakarta. Saat ini Penulis bekerja sebagai *Business Development* di Radiant Group Jakarta. Beberapa *research interest* penulis antara lain *Artificial Intelligence and Renewable energy*. Penulis pernah mendapat penghargaan sebagai *Best Paper* di The 7th *Indonesia International Geothermal Convention and Exhibition (IIGCE) 2019*.

Email Penulis: aldicahyamu@gmail.com

BAB 2 KERANGKA PEMBELAJARAN PAC (*THE PAC LEARNING FRAMEWORK*)

Ir. Anak Agung Gde Bagus Ariana, S.T., M.T.

Institut Bisnis dan Teknologi Indonesia

Pengantar

Saat ini, *Machine Learning* merupakan konsep yang sangat berkaitan dengan "belajar dari data". Dalam merancang dan menganalisis algoritma pembelajaran mesin, seringkali muncul beberapa pertanyaan, seperti berapa banyak data atau sampel yang diperlukan untuk mencapai hasil pembelajaran yang sukses (*sample complexity*). Selain itu, muncul juga pertanyaan tentang hal apa yang bisa dipelajari secara efisien dan masalah-masalah apa yang sulit dipelajari (*time and space complexity*). Ada pula pertanyaan apakah ada algoritma pembelajaran mesin yang mampu mempelajari segala hal (*general model of learning*). Oleh karena itu, dibutuhkan suatu kerangka formal seperti *Probably Approximately Correct (PAC) Learning* untuk menjawab pertanyaan-pertanyaan tersebut sehingga proses perancangan algoritma pembelajaran mesin dapat dilakukan dengan lebih efisien dan tidak hanya bergantung pada keahlian seseorang semata.

Definisi

Framework adalah sebuah kerangka kerja atau kerangka acuan yang digunakan untuk membuat dan mengembangkan suatu sistem atau aplikasi. *Framework* menyediakan struktur dan komponen yang

pengujian. Dalam contoh ini, kita akan menggunakan 1000 sampel pelatihan.

PAC Error: Probabilitas bahwa model yang dilatih tidak lebih buruk daripada toleransi kesalahan *epsilon*.

Contoh: kita ingin membangun model yang dapat memprediksi apakah seorang pelanggan akan membeli produk tertentu berdasarkan usia dan pendapatan. Kita ingin membangun model yang dapat memprediksi dengan keakuratan sekitar 95%. Dalam contoh ini, kita akan mengasumsikan bahwa model yang dibangun memenuhi kriteria *PAC*.

Generalization Bound: Menghitung tingkat kesalahan model pada data yang tidak digunakan untuk pelatihan.

Contoh: kita ingin membangun model yang dapat memprediksi apakah seorang pelanggan akan membeli produk tertentu berdasarkan usia dan pendapatan. Kita ingin membangun model yang dapat memprediksi dengan keakuratan sekitar 95%. Dalam contoh ini, kita akan menggunakan rumus *generalization bound* untuk menghitung batas atas kesalahan pada data yang tidak digunakan untuk pelatihan.

Rumus untuk menghitung *generalization bound* adalah:

$$P(|E_{out}(h) - E_{in}(h)| > \epsilon) \leq 2Me^{-2\epsilon^2 N}$$

Di mana $E_{out}(h)$ adalah *error* aktual pada data yang tidak digunakan untuk pelatihan, $E_{in}(h)$ adalah *error empiris* pada data pelatihan, M adalah jumlah *hipotesis* yang mungkin, ϵ adalah toleransi kesalahan, dan N adalah jumlah sampel pelatihan.

Dalam contoh kita, mari kita asumsikan bahwa kita telah menguji model pada 1000 sampel pengujian yang tidak digunakan untuk pelatihan, dan hasilnya menunjukkan bahwa model memiliki *error* aktual sebesar 0,08. Jumlah *hipotesis* yang mungkin adalah 100, dan toleransi kesalahan kita adalah 0,05.

Maka, tingkat kesalahan pada data yang tidak digunakan untuk pelatihan adalah:

$$P(|0.08 - E_{in}(h)| > 0.05) \leq 2 \times 100 \times e^{-2 \times 0.05^2 \times 1000}$$

$$P(|0.08 - E_{in}(h)| > 0.05) \leq 2 \times 10^{-2}$$

Daftar Pustaka

- Blumer, A., Ehrenfeucht, A., Haussler, D., & Warmuth, M. K. (1989). *Learnability and the Vapnik-Chervonenkis dimension*. Journal of the ACM (JACM), 36(4), 929-965.
- Gold, E. M. (1967). *Language identification in the limit*. Information and Control, 10(5), 447-474.
- Kearns, M. J., & Vazirani, U. V. (1994). *An introduction to computational learning theory*. MIT press.
- Kulkarni, T. D., & Boyd, S. (2016). *PAC-Bayesian theory meets convex optimization*. Foundations and Trends® in Optimization, 2(3-4), 105-191.
- Long, P. M. (1994). *PAC learning of neural networks*. IEEE Transactions on Neural Networks, 5(4), 594-603.
- Long, P. M. (1999). *On the PAC-Bayes model of learning*. Journal of Computer and System Sciences, 58(1), 137-163.
- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning*. MIT Press.
- Schapire, R. E. (1990). *The strength of weak learnability*. Machine Learning, 5(2), 197-227.
- Vapnik, V. (1995). *The nature of statistical learning theory*. Springer.
- Vapnik, V. N., & Chervonenkis, A. Y. (1971). *On the uniform convergence of relative frequencies of events to their probabilities*. Theory of Probability & Its Applications, 16(2), 264-280.

PROFIL PENULIS



**Ir. Anak Agung Gde Bagus Ariana, S.T.,
M.T.**

Penulis kelahiran Gianyar-Bali tahun 1986 ini adalah dosen tetap Program Studi Sistem Komputer di Institut Bisnis dan Teknologi Indonesia sejak tahun 2010. Bidang riset yang ditekuni adalah *data mining* dan pengolahan citra digital. Penulis lulusan S1 dan S2 Teknik Elektro Universitas Udayana.

Email: gungariana@instiki.ac.id

BAB 3 *SUPPORT VECTOR MACHINE*

Indo Intan, S.T., M.T.
Universitas Dipa Makassar

Pendahuluan

Support Vector Machine (SVM) diajukan oleh Cortes, Boser, Guyon, Vapnik dimulai pada tahun 1992 hingga tahun 2000 sebagai suatu teknik klasifikasi. *Support Vector Machine (SVM)* dikenal sebagai teknik pembelajaran mesin (*machine learning*) paling mutakhir dan efisien setelah pembelajaran mesin sebelumnya, seperti *neural network*.

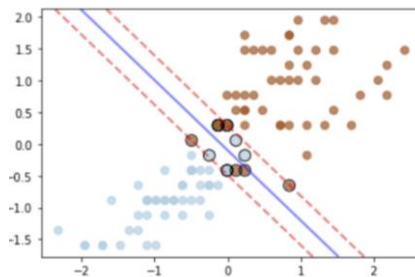
Kedua metode ini berhasil digunakan dalam pengenalan pola. Pengenalan pola bertujuan untuk mengklasifikasikan data berdasarkan pengetahuan *a priori* atau statistik informasi yang diambil dari data mentah, yaitu alat yang ampuh dalam pemisahan data di banyak disiplin ilmu.

SVM adalah jenis pembelajaran terawasi melalui proses pelatihan. Pembelajaran menggunakan pasangan data *input* dengan data *output* berupa target data yang diinginkan. Pada prosesnya, setiap kelas objek ditandai sebagai kelas yang berbeda sebanyak jumlah kelas yang ada sebagai label. Pada proses validasi dan testing, model akan diminta memprediksi kategori baru, apakah model memiliki kemampuan yang lebih besar untuk menggeneralisasi masalah (kemampuan mengenali masalah baru berdasarkan pengetahuan sebelumnya), yang merupakan tujuan secara statistik.

Secara sederhana, *SVM* merupakan usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah kelas pada ruang *input*. *SVM* mencoba menemukan fungsi pemisah (*hyperplane*) dengan cara memaksimalkan jarak antar kelas. Teknik ini akan memastikan

Support Vector Machine

```
le = preprocessing.LabelEncoder()
y = le.fit_transform(iris["species"])
X = iris.drop(["species"], axis=1)
if len(cols) > 0:
    X = X[cols]
return X.values, y
if __name__ == '__main__':
    # make sure the targets are (-1, +1)
    cols = ["petal_length", "petal_width"]
    X, y = load_data(cols)
    y[y == 0] = -1
    # scale the data
    scaler = StandardScaler()
    X = scaler.fit_transform(X)
    # now we'll use our custom implementation
    model = LinearSVMUsingSoftMargin(C=15.0)
    model.fit(X, y)
    print("train score:", model.score(X, y))
    model.plot_decision_boundary()
```



Gambar 3.15. Grafik Output Soft Margin Classifier

Sumber : <http://www.adeveloperdiary.com/data-science/machine-learning/support-vector-machines-for-beginners-linear-svm/>

Daftar Pustaka

Alpaydin, E. (2014). *Introduction to Machine Learning*. MIT Press.

Araque, A., Carmignoto, G., Bloomfield, S. M., Mckinney, J., Smith, L., Bogen, I. L., Risa, O., Haug, K. H., Sonnewald, U., Fonnum, F., Bonni, A., Sun, Y., Bhatt, A., Frank, D. A., Rozovsky, I., Stahl, N., & Yancopoulos, G. D. (2008). *Application Research of Support Vector Machine in Network Security Risk Evaluation* (Vol. 29, Issue 2001, pp. 2004–2010). 2008 International Symposium on Intelligent Information Technology Application Workshops.

Casey, R. G., & Nagy, G. (1971). *Advances in Pattern Recognition Support Vector Machines for Pattern Classification*. In *Scientific American* (Vol. 224, Issue 4). <https://doi.org/10.1038/scientificamerican0471-56>

Christopher J.C. Burges. (1998). *A Tutorial on Support Vector Machines for Pattern Recognition*. *Data Mining and Knowledge Discovery*, 2, 121–167.

Cristianini, N., & Shawe-Taylor, J. (2017). *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press.

Jakkula, V. (2011). *Tutorial on Support Vector Machine (SVM)*. School of EECS, Washington State University, 1–13. <http://www.ccs.neu.edu/course/cs5100f11/resources/jakkula.pdf>

Jana, A. (n.d.). *Support Vector Machines for Beginners – Linear SVM*. A Developer Diary. <http://www.adeveloperdiary.com/data-science/machine-learning/support-vector-machines-for-beginners-linear-svm>

Jordan, M., Kleinberg, J., & Schölkopf, B. (2006). *Support Vector Machines (Information Science and Statistics)*. www.springer.com/series/3816

Li, Z., Yuan, R., & Guan, X. (2007). *Accurate classification of the internet traffic based on the SVM method*. IEEE International Conference on

Support Vector Machine

Communications,1373–1378.

<https://doi.org/10.1109/ICC.2007.231>

Ma, Y., & Guo, G. (2014). *Support vector machines applications*. In *Support Vector Machines Applications* (Vol. 9783319023).

<https://doi.org/10.1007/978-3-319-02300-7>

Noble, W. S. (2006). *What is a support vector machine?* *Nature Biotechnology*, 24(12), 1565–1567.

<https://doi.org/10.1038/nbt1206-1565>

Pisner, D. A., & Schnyer, D. M. (2019). *Support vector machine*. *Machine Learning: Methods and Applications to Brain Disorders*, 101–121.

<https://doi.org/10.1016/B978-0-12-815739-8.00006-7>

Scholkopf, B., & Smola, A. J. (2002). *Learning with Kernels, Support Vector Regularization, Optimization, and Beyond*. In *Journal of Chemical Information and Modeling*. The MIT Press, Cambridge, Massachusetts.

Vapnik, V. N. (2000). *The Nature of Statistical Learning Theory*. Springer.

Vijayakumar, S., & Wu, S. (1999). *Sequential Support Vector Classifiers and Regression*. *Proceedings of International Conference on Soft Computing (SOCO '99)*, 619(February), 610–619.

Yuan, R., Li, Z., Guan, X., & Xu, L. (2010). *An SVM-based machine learning method for accurate Internet traffic classification*. *Information Systems Frontiers*, 12(2), 149–156. <https://doi.org/10.1007/s10796-008-9131-2>

PROFIL PENULIS



Indo Intan, S.T., M.T.

Lahir di Lajokka Kabupaten Wajo pada tanggal 29 Desember. Ia Lulus pada tahun 2010 hingga mendapat gelar Magister Teknik pada Jurusan Elektro di Universitas Hasanuddin. Saat ini ia tercatat sebagai dosen tetap untuk beberapa mata kuliah dalam kelompok sistem cerdas seperti Pengenalan Pola, *Deep Learning*, dan *Sistem Mikrokontroler* di Universitas Dipa Makassar.

Selain mengajar ia aktif dalam kegiatan tridarma lainnya di antaranya ialah penelitian dan pengabdian kepada masyarakat. Saat ini ia pun diberikan tanggung jawab sebagai *reviewer* pada *Jurnal Artificial Intelligence Review (Belanda)* dan beberapa Universitas yang terdapat di Indonesia, di antaranya pada *Jurnal Dipa Komsu-Komti* (Undiversitas Dipa Makassar), *Jurnal JIRE* (STMIK Lombok), *Jurnal Dharma Raflesia* (Universitas Bengkulu), dan *Jurnal Sinkron* (Insitut Penelitian Medan). Beberapa penelitian yang berhasil didanai oleh Ristekdikti dari tahun 2013 hingga sekarang berjudul: *Simulasi Sistem Penanganan Kebakaran Otomatis Berbasis Mikrokontroler (2012)*, *Identifikasi Pola Terhadap Citra Kanker Tiroid Menggunakan Metode Gaussian Markov Random Field Dengan Metode Klasifikasi Self Organizing Map Kohonen (2013)*, *Sistem Autentikasi Biometrik Multimodal Pada Presensi Perkuliahan (2015)*. Demikian juga pada pengabdian masyarakat, di antaranya: *Sistem Informasi Pengembangan Potensi Dan Hasil Pembangunan Pulau Barrang Lompo Berbasis Web (2015)* dan *IbM Pelatihan Animasi Visual Dalam Penyusunan Teknik Pembelajaran Berbasis Multimedia Sebagai Upaya Peningkatan Profesionalisme Guru SDN Barrang Lompo Dan SMPN 28 Di Kelurahan Barrang Lompo (2014)*.

Adapun karya buku yang telah ditulisnya yaitu: *Satistik, Teman Belajar dan Analisis Mahasiswa (2019)* dan *Jaringan Saraf Tiruan (2022)*. **Prinsipnya adalah mendedikasikan dan mengupgrade diri untuk ilmu pengetahuan dan keterampilan, berharap menuai kebaikan bagi semua.**

Email Penulis: indo.intan@undipa.ac.id

BAB 4 METODE *KERNEL*

Sumanto, M.Kom.

Universitas Bina Sarana Informatika

Apa itu *Kernel*?

Kernel adalah suatu ukuran *similarity* (kesamaan). Secara garis besar, algoritma *machine learning* yang hanya bergantung pada produk titik di antara instance "*kernelized*" dengan mengganti semua instance dari (x, x') oleh fungsi kernel $k(x, x')$. Dasar dari *kernel* dimisalkan x merupakan *input* dan $k = X \times X \rightarrow \mathbb{R}$ maka fungsi yang dihasilkan adalah $k(x, x') = (\varphi(x), \varphi(x'))$, dimana φ adalah fitur dari peta yang mana memetakan X kedalam beberapa *dot product space* H . Dengan kata lain *kernel* menyesuaikan *dot product* di beberapa *dot product space*. (A. Smola & Vishwanathan, 2008).

Keuntungan utama menggunakan *kernel* sebagai ukuran *similarity* adalah :

Pertama, Jika *fitur space* cukup maka *estimasi* sederhana seperti *hyperplane* dan *half-space* mungkin sudah cukup.

Kedua, *kernel* memungkinkan kita untuk membangun algoritma *machine learning* di *dot product space* tanpa menghitung nilai $\varphi(x)$.

Ketiga, kita tidak perlu membuat asumsi tentang *input space* X .

Kernel Dalam Machine Learning

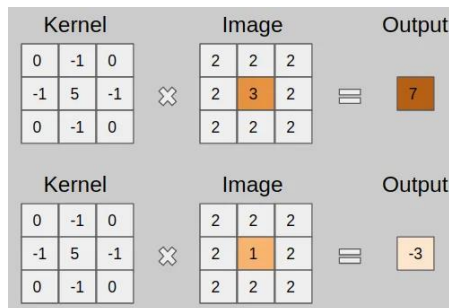
Kernel dalam *machine learning* adalah fungsi matematis yang memetakan data dari dimensi rendah ke dimensi yang lebih tinggi. *Kernel* ini biasanya digunakan dalam algoritma pembelajaran mesin yang bergantung pada jarak antara pasangan data, seperti *Support Vector Machines (SVMs)*. Dalam *SVM*, *kernel* mengubah data dari ruang dimensi rendah menjadi ruang dimensi yang lebih tinggi sehingga *SVM*

Metode Kernel

Menurut (Gonzalez & Woods, 2018), "Operasi konvolusi yang memanfaatkan kernel filter digunakan dalam berbagai aplikasi pemrosesan citra. Konvolusi menghasilkan gambar baru dengan menggabungkan sinyal asli dan kernel filter, yang menekankan beberapa aspek sinyal dan menekan yang lain."

Contoh kernel konvolusi yang umum digunakan dalam pengolahan citra adalah *kernel Gaussian*, yang digunakan untuk menghaluskan gambar atau untuk mengurangi *derau* pada gambar. Ada juga *kernel konvolusi* lainnya, seperti *kernel sobel* dan *kernel laplace*, yang digunakan untuk mendeteksi tepi pada gambar atau untuk mengidentifikasi fitur-fitur penting lainnya pada gambar.

Convolution kernel adalah matriks (biasanya berukuran ganjil) yang digunakan dalam operasi konvolusi pada pengolahan citra dan sinyal. Contoh *convolution kernel* yang paling sederhana adalah kernel identitas, yaitu:



Gambar 4.1: Contoh Convolution Kernel

Sumber : <https://towardsdatascience.com/types-of-convolution-kernels-simplified-f040cb307c37>

String Kernel

String kernel adalah salah satu jenis *kernel* yang digunakan dalam pembelajaran mesin dan pengolahan teks. *Kernel* ini digunakan untuk mengukur kesamaan antara dua buah *string* berdasarkan pada kesamaan *sub-string* (bagian dari *string*) yang terdapat pada keduanya.

Pada dasarnya, *string kernel* menggunakan fungsi basis basis (*basis function*) untuk menghitung produk titik antara dua buah vektor yang merepresentasikan kedua *string*. Fungsi basis tersebut dihitung dengan menggunakan kesamaan antara *sub-string* pada kedua *string*. Jumlah

PROFIL PENULIS



Sumanto, M.Kom.

Ketertarikan penulis terhadap ilmu komputer dimulai pada tahun 2002 silam. Hal tersebut membuat penulis memilih untuk masuk ke Universitas Bina Sarana Informatika dengan memilih Jurusan Manajemen Informatika (MI) dan berhasil lulus pada tahun 2006. Penulis kemudian melanjutkan pendidikan ke Perguruan Tinggi dan berhasil menyelesaikan studi S1 di prodi Teknik Informatika Universitas Nusa Mandiri pada tahun 2007. Dua tahun kemudian, penulis menyelesaikan studi S2 di prodi Ilmu Komputer Program Pasca Sarjana Universitas Nusa Mandiri, Saat ini Penulis sedang melanjutkan Pendidikan S3 pada IPB University jurusan Ilmu Komputer .

Penulis memiliki kepakaran dibidang *Image Processing*, *Signal Processing* dan *Data Mining*. Dan untuk mewujudkan karir sebagai dosen profesional, penulis pun aktif sebagai peneliti dibidang kepakarannya tersebut. Beberapa penelitian yang telah dilakukan didanai oleh internal perguruan tinggi dan juga Kemenristek DIKTI. Selain peneliti, penulis juga aktif menulis buku dengan harapan dapat memberikan kontribusi positif bagi bangsa dan negara yang sangat tercinta ini. Atas dedikasi dan kerja keras dalam menulis buku.

Email Penulis: sumanto@bsi.ac.id

BAB 5 PEMBELAJARAN *ONLINE*

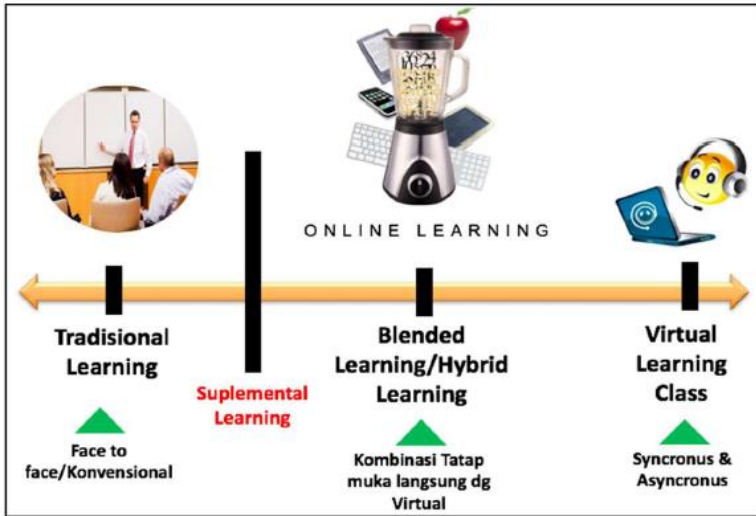
Pastima Simanjuntak, S.Kom., M.SI.
Universitas Putera Batam

Pengertian Pembelajaran *Online*

Pendidikan yang berorientasi pada siswa bisa dilakukan dengan membangun sistem pendidikan yang memperbolehkan siswa mempunyai keahlian guna belajar lebih menarik dan interaktif. Siswa wajib mempunyai kompetensi yang bermanfaat untuk masa depannya. Bersamaan dengan pertumbuhan teknologi dan infrastruktur penunjangnya, upaya untuk meningkatkan mutu pendidikan bisa dicoba lewat pemanfaatan teknologi tersebut dalam sesuatu sistem yang diketahui dengan *online learning*.

Pembelajaran secara *online* diketahui dari pertumbuhan pendidikan berbasis elektronik (*e-learning*) yang diperkenalkan oleh Universitas Illionis lewat sistem pendidikan berbasis PC. *Online learning* ialah sesuatu sistem yang bisa memfasilitasi siswa belajar lebih luas, lebih banyak, serta bermacam-macam. Lewat sarana yang disediakan oleh sistem tersebut, siswa bisa belajar kapan dan dimana saja tanpa terbatas oleh jarak, ruang serta waktu. Pembelajaran secara *online* sangat berbeda dengan pendidikan secara konvensional. Pembelajaran secara *online* lebih menekankan pada ketelitian serta kejelian siswa dalam menerima serta mencerna data yang disajikan secara *online*.

E-learning merupakan pendidikan yang membolehkan ataupun didukung oleh pemakaian perlengkapan serta *konten digital*. *E-learning* umumnya mengaitkan sebagian wujud *interaktivitas*, tercantum di dalamnya bisa berbentuk interaksi *online* antara partisipan didik serta pengajar ataupun kelompok mereka. *E-learning* umumnya diakses



Gambar 5.1 *Blended Learning/Hybrid Learning*

Sumber : Diolah Penulis

Online learning terus banyak digunakan sekarang ini, mungkin kita dapat melihat bahwa media ini sesuai dan mempunyai banyak kelebihan. Hal ini bisa dilihat dari *fitur* yang ada rata-rata cuma berbentuk *Next* dalam serangkaian persoalan yang dapat dijawab dengan berupaya terus tanpa wajib dipelajari secara sungguh- sungguh. Pada biasanya warga lebih suka meniru produk inovasi yang baru tanpa didasari oleh kreativitas, lebih cenderung memola dari yang telah ada.

Pendidikan *online* sudah menjajaki pola yang sama. Dengan merancang pendidikan *online* yang pada dasarnya tidak lebih dari novel bacaan yang dipindahkan pada layar komputer, pengembang kandas memakai keuntungan dari *website* untuk merancang *online learning*. Apa yang kita wajib jalani malah mengevaluasi ciri dari teknologi ini buat memikirkan apa yang jadi kebutuhan siswa. Tabel di bawah ini menguraikan sebagian yang pro serta kontra dari kelas tradisional serta pendidikan *online*. Sebagai contoh, teknologi jaringan (*networking*) membolehkan orang untuk berbicara serta berbagi, sehingga seluruh data bisa dikirim serta diterima lewat *e-mail*. Ini merupakan metode yang mudah serta bisa diandalkan berbagi pengetahuan. Kelebihannya kalau kita bisa dengan mudah menggunakan keahlian ini untuk bisa belajar.

Tabel 5.2 Potensi Pembelajaran dikelas yang setuju dan yang tidak setuju

Potensi Pembelajaran di Kelas (<i>Classroom Setting</i>) yang setuju	Potensi Pembelajaran di Kelas (<i>Classroom Setting</i>) yang tidak setuju
<ol style="list-style-type: none"> 1. Umpan balik (<i>feed back</i>) segera. 2. Isyarat visual. 3. Kemudahan interaksi sosial. 4. Perilaku guru dapat dengan mudah ditiru oleh siswanya sebagai bentuk pemodelan. 	<ol style="list-style-type: none"> 1. Orang yang pemalu atau kurang kritis tidak memberikan kontribusi dalam pembelajaran. 2. Pembelajaran dikendalikan oleh guru sepenuhnya. 3. Berbasis peristiwa (terjadi pada satu waktu tertentu).
Potensi <i>Online Learning</i> yang Pro	Potensi <i>Online Learning</i> yang Kontra
<ol style="list-style-type: none"> 1. Setiap orang dapat memberikan kontribusi. 2. Pembelajaran dikendalikan oleh siswa (<i>learner</i>). 3. Proses berbasis <i>online</i> (dapat terjadi kapan saja, dimana saja). 4. Hasil interaksi pembelajaran tercatat oleh sistem dengan baik. 	<ol style="list-style-type: none"> 1. Kurangnya isyarat <i>visual</i> menjadi penghambat terhadap teknologi dan akses belajar. 2. Komunikasi lebih terjadi hanya sebatas komunikasi tulis.

Sumber: diolah penulis

Implementasi pembelajaran yang digunakan memerlukan satu pemikiran untuk memutuskan mana yang sangat sesuai. Guru butuh memikirkan dengan mengacu pada ciri modul pelajaran, kompetensi yang hendak dicapai, dan ciri siswa. Pada dasarnya seluruh model tersebut mempunyai keunggulan. Yang berarti buat dicermati dalam keadaan siswa. Pembelajaran *online* memerlukan sebagian persyaratan untuk siswa, ialah:

1. ICT Literacy

Siswa wajib mempunyai keahlian dini berbentuk kemampuan *ICT* sebagai perlengkapan buat belajar, maksudnya bila siswa kelas rendah dimana keahlian membaca serta menulisnya belum baik,

staf pendidik yang lain. Pendidikan konektif didasarkan pada pendidikan sosial serta teori pendidikan konstruktivis, semacam yang dipaparkan oleh George Siemens. Baginya kalau belajar tidak wajib dilihat sebagai sesuatu peristiwa, namun suatu proses yang mengaitkan antara memori, kognisi, emosi, kepercayaan, serta anggapan. Tidak hanya itu belajar bisa dicoba dengan bermacam metode semacam mengirim *e-mail*, menelusuri *web*, melaksanakan obrolan *online* serta lain-lain. Lewat pembelajaran *online* siswa akan terkoneksi dengan dunia maya. Siswa hendak lebih banyak menciptakan banyak sumber belajar yang tidak terbatas. Dalam pembelajaran *online* tidak adanya batas ruang serta waktu sehingga siswa bisa belajar secara terkoneksi.

Daftar Pustaka

- Albashtawi, Abeer Hameed; Al Bataineh, Khaleel Bader. (2020). *The Effectiveness of Google Classroom Among EFL Students in Jordan: An Innovative Teaching and Learning Online Platform*. International Journal of Emerging Technologies in Learning (IJET) vol. 15 iss. 11. DOI: 10.3991/ijet.v15i11.12865
- Andrew D. Moffat. (2022). *Second Language Use Online and its Integration in Formal Language Learning: From Chatroom to Classroom*. Second Language Acquisition, 153. ISBN: 1800413629; 9781800413627. Publisher: Multilingual Matters
- Ann Marcus-Quinn, Triona Hourigan (eds.) (2021). *Handbook for Online Learning Contexts: Digital, Mobile and Open: Policy and Practice* Edition: 1st ed. DOI: 10.1007/978-3-030-67349-9. ISBN: 9783030673482. Publisher: Springer
- Gutierrez-Estevez, M. A. ;Kasparick, Martin ;Stanczak, Slawomir. (2021). *Online Learning of Any-to-Any Path Loss Maps*. IEEE Communications Letters. DOI: 10.1109/lcomm.2021.3050252
- Huang, Chao ;Peng, Zongju ;Xu, Yong ;Chen, Fen ;jiang, Qiuping ;Zhang, Yun ;jiang, Gangyi ;Ho, Yo-Sung. (2021). *Online Learning-Based Multi-*

Stage Complexity Control for Live Video Coding. IEEE Transactions on Image Processing vol. 30. DOI: 10.1109/tip.2020.3036766. Publisher: Institute of Electrical and Electronics Engineers (IEEE)

Huang, Jinchao ;Li, Guofu ;Tian, Jianwei ;Li, Shenghong. (2021). *Accurate interpretation of the online learning model for 6G-enabled Internet of Things*. IEEE Internet of Things Journal. DOI: 10.1109/jiot.2020.3048793. Publisher: Institute of Electrical and Electronics Engineers (IEEE)

Joos Korstanje. (2022). *Machine Learning for Streaming Data with Python: Rapidly build practical online machine learning solutions*. ISBN: 9781803248363; 180324836X. Publisher: Packt

Karaoglan Yilmaz, Fatma Gizem; Yilmaz, Ramazan. (2020). *Learning analytics as a metacognitive tool to influence learner transactional distance and motivation in online learning environments*. Innovations in Education & Teaching International. DOI: 10.1080/14703297.2020.1794928

Kim, Min Kyu; Lee, In Heok; Wang, Yinying. (2020). *How students emerge as learning leaders in small group online discussions*. Journal of Computer Assisted Learning. DOI: 10.1111/jcal.12431

Laurence Mann;Jieun Kiaer;Emine Çakır. (2022). *Online Language Learning: Tips for Teachers* Edition: 1. Publisher: Palgrave Macmillan. DOI: 10.1007/978-3-030-91418-9. ISBN: 9783030914189

Manegre, Marni; Sabiri, Kashif Ali. (2020). *Online language learning using virtual classrooms: an analysis of teacher perceptions*. Computer Assisted Language Learning. DOI: 10.1080/09588221.2020.1770290

Siregar, N; Manurung, S L. (2020). *Development of a blended learning model using the online learning system (SIPDA)*, Universitas Negeri Medan. Journal of Physics Conference Series vol. 1462. DOI: 10.1088/1742-6596/1462/1/012020

Susan Ko, Olena Zhadko. (2022). *Best Practices for Mentoring in Online Programs: Supporting Faculty and Students in Higher Education*. Best Practices in Online Teaching and Learning. ISBN: 9780429434754; 0429434758. Publisher: Routledge City: New York

Yang, Miao; Zhu, Hongbin; Wang, Haifeng; Koucheryavy, Yevgeni; Samouylov, Konstantin; Qian, Hua. (2021). *An Online Learning Approach to Computation Offloading in Dynamic Fog Networks*. IEEE Internet of Things Journal. DOI: 10.1109/JIOT.2020.3015522

Zalli, Mohd Muslim Md; Nordin, Hasniza; Awang Hashim, Rosna. (2020). *Online Self-Regulated Learning Strategies in MOOCs: A Measurement Model*. International Journal of Emerging Technologies in Learning (IJET) vol. 15 iss. 08. DOI: 10.3991/ijet.v15i08.12401

PROFIL PENULIS



Pastima Simanjuntak, S.Kom., M.SI.

Lulusan Sarjana Komputer dari Universitas Katolik Santo Thomas Sumatera Utara pada Tahun 2007 dan Lulusan Magister Sistem Informasi dari STMIK Putera Batam Pada Tahun 2012. Saat ini Dosen Aktif Pada Fakultas Teknik dan Komputer Universitas Putera Batam dari 2009 sampai sekarang.

BAB 6 RANKING

Satria, S.Kom., M.Kom.

Universitas Bani Saleh Bekasi

Definisi Ranking

Peringkat adalah jenis pembelajaran mesin yang mengurutkan data dalam urutan yang relevan. Perusahaan menggunakan peringkat untuk mengoptimalkan pencarian dan rekomendasi. *Learning to rank* atau *machine-learned ranking (MLR)* adalah penerapan pembelajaran mesin, biasanya pembelajaran yang diawasi, semi-diawasi atau penguatan, dalam pembangunan model peringkat untuk sistem temu kembali informasi (M. Mohri, 2012). Data pelatihan terdiri dari daftar item dengan beberapa urutan parsial yang ditentukan di antara item dalam setiap daftar. Urutan ini biasanya diinduksi dengan memberikan skor numerik atau *ordinal* atau penilaian *biner* (misalnya "relevan" atau "tidak relevan") untuk setiap item.

Tujuan membangun model peringkat adalah untuk memberi peringkat pada daftar baru yang tidak terlihat dengan cara yang mirip dengan peringkat dalam data pelatihan. Sebagai contoh data pelatihan terdiri dari *kueri* dan dokumen yang mencocokkannya bersama dengan tingkat relevansi dari setiap kecocokan. Ini dapat disiapkan secara manual oleh penilai manusia, yang memeriksa hasil untuk beberapa *kueri* dan menentukan relevansi setiap hasil. Tidak mungkin untuk memeriksa relevansi semua dokumen, dan biasanya teknik yang disebut penyatuan digunakan-hanya beberapa dokumen teratas, yang diambil oleh beberapa model peringkat yang ada yang diperiksa. Teknik ini dapat menimbulkan bias seleksi. Alternatifnya, data pelatihan dapat diperoleh secara otomatis dengan menganalisis (Joachim,2002) rantai *kueri*, atau *fitur* mesin dapat dibiarkan oleh kecenderungan pengguna untuk mengklik hasil pencarian teratas dengan asumsi bahwa mereka sudah berperingkat baik (Joachim, 2005).



Gambar 6.3 Contoh DCG
 Sumber : Francesco Casalegno, 2022

Untuk *kueri* yang diberikan q dan dokumen terkait $D = \{d_1, \dots, d_n\}$, kami mempertimbangkan dokumen yang diambil ke- k teratas. Perolehan $G_k = 2^{y_k} - 1$ mengukur seberapa berguna dokumen ini (kita ingin dokumen dengan relevansi tinggi), sedangkan diskon $D_k = 1/\log_2(k+1)$ menghukum dokumen yang diambil dengan peringkat lebih rendah (kita ingin dokumen yang relevan di peringkat teratas).

Jumlah istilah keuntungan yang didiskon $G_k D_k$ untuk $k = 1 \dots n$ adalah Keuntungan Kumulatif yang Didiskontokan (DCG). Untuk memastikan bahwa skor ini terikat antara 0 dan 1, kita dapat membagi DCG terukur dengan skor ideal IDCG yang diperoleh jika kita memeringkat dokumen dengan nilai sebenarnya y_k . Ini memberi kita Keuntungan Kumulatif Diskon Normalisasi (NDCG), di mana $NDCG = DCG/IDCG$.

Terakhir, untuk MAP, biasanya menghitung rata-rata nilai DCG atau NDCG untuk sekumpulan m kueri untuk mendapatkan nilai rata-rata.

Model Pembelajaran Mesin Untuk Peringkat/Ranking

Untuk membangun model *Machine Learning* untuk peringkatan, kita perlu mendefinisikan fungsi *input*, *output*, dan *loss*.

1. Masukan – Untuk *kueri* q kami memiliki n dokumen $D = \{d_1, \dots, d_n\}$ untuk diurutkan berdasarkan relevansi. Elemen $x_i = (q, d_i)$ adalah input untuk model kita.

Ranking

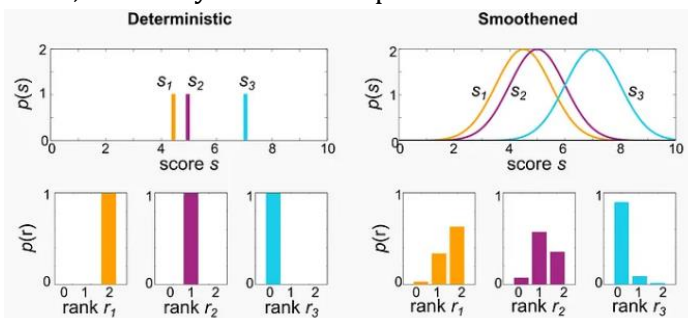
menjalankan optimalisasi *Gradient Descent* kita tidak memerlukan fungsi *loss*, kita hanya membutuhkan gradiennya saja, *LambdaRank* (wang, 2018) mendefinisikan *gradien* dari fungsi kerugian *implisit* sehingga dokumen dengan peringkat tinggi memiliki *gradien* yang jauh lebih besar:

$$\lambda_j := \frac{\partial L}{\partial s_j} = \frac{1}{G_{max}} \sum_{i \neq j} \sigma(s_i - s_j) |G_i - G_j| |D_i - D_j|$$

Memiliki *gradien* juga cukup untuk membuat model *Gradient Boosting*. Ini adalah ide yang digunakan *LambdaMART*, menghasilkan hasil yang lebih baik dibandingkan dengan *LambdaRank* (Burges, 2010)

6. Metode *Listwise*

Pendekatan *pointwise* dan *pairwise* mengubah masalah peringkat menjadi tugas *regresi* atau klasifikasi pengganti. Metode *Listwise*, sebaliknya, memecahkan masalah secara lebih langsung dengan memaksimalkan *metrik* evaluasi. Secara intuitif, pendekatan ini akan memberikan hasil terbaik, karena informasi tentang peringkat dieksploitasi sepenuhnya dan *NDCG* dioptimalkan secara langsung. Tapi masalah yang jelas dengan pengaturan $Loss = 1 - NDCG$ adalah informasi peringkat yang diperlukan untuk menghitung diskon D_k hanya tersedia setelah menyortir dokumen berdasarkan skor yang diprediksi, dan menyortir tidak dapat dibedakan.



Gambar 6.5 Ketidakpastian dalam Skor Menghasilkan Kerugian yang Mulus di *Softrank*

Sumber : Francesco Casalegno, 2022

Ranking

2006	<i>IR-SVM</i>	<i>pairwise</i>
2006	<i>LambdaRank</i>	<i>pairwise/listwise</i>
2007	<i>AdaRank</i>	<i>listwise</i>
2007	<i>FRank</i>	<i>pairwise</i>
2007	<i>RankRLS</i>	<i>pairwise</i>
2007	<i>SVMmap</i>	<i>listwise</i>
2008	<i>LambdaSMART/LambdaMART</i>	<i>pairwise/listwise</i>
2008	<i>ListMLE</i>	<i>listwise</i>
2008	<i>Ranking Refinement</i>	<i>pairwise</i>
2008	<i>SSRankBoost</i>	<i>pairwise</i>
2008	<i>SortNet</i>	<i>pairwise</i>
2009	<i>MPBoost</i>	<i>pairwise</i>
2009	<i>BoltzRank</i>	<i>listwise</i>
2009	<i>BayesRank</i>	<i>listwise</i>
2010	<i>NDCG Boost</i>	<i>listwise</i>
2010	<i>GBlend</i>	<i>pairwise</i>
2010	<i>IntervalRank</i>	<i>pairwise & listwise</i>
2010	<i>CRR</i>	<i>pointwise & pairwise</i>
2014	<i>LCR</i>	<i>pairwise</i>
2015	<i>FaceNet</i>	<i>pairwise</i>
2016	<i>XGBoost</i>	<i>pairwise</i>
2017	<i>ES-Rank</i>	<i>listwise</i>
2018	<i>DLCM</i>	<i>listwise</i>
2018	<i>PolyRank</i>	<i>pairwise</i>
2018	<i>FATE-Net/FETA-Net</i>	<i>listwise</i>

2019	<i>FastAP</i>	<i>listwise</i>
2019	<i>Mulberry</i>	<i>listwise & hybrid</i>
2019	<i>DirectRanker</i>	<i>pairwise</i>
2019	<i>GSF</i>	<i>listwise</i>
2020	<i>RaMBO</i>	<i>listwise</i>
2020	<i>PRM</i>	<i>pairwise</i>
2021	<i>PiRank</i>	<i>listwise</i>
2022	<i>SAS-Rank</i>	<i>listwise</i>
2022	<i>VNS-Rank</i>	<i>listwise</i>
2022	<i>VNA-Rank</i>	<i>listwise</i>

Sumber : https://en.wikipedia.org/wiki/Learning_to_rank

Kesimpulan

Masalah peringkat/ranking sering ditemukan di mana-mana, mulai dari pencarian informasi hingga sistem pemberi rekomendasi dan pemesanan perjalanan. *Metrik* evaluasi seperti *MAP* dan *NDCG* memperhitungkan peringkat dan relevansi dokumen yang diambil, dan karenanya sulit untuk dioptimalkan secara langsung.

Metode *Learning to Rank* menggunakan model *Machine Learning* untuk memprediksi skor relevansi suatu dokumen, dan dibagi menjadi 3 kelas: *pointwise*, *pairwise*, *listwise*. Pada sebagian besar masalah peringkatan, metode *listwise* seperti *LambdaRank* dan kerangka umum *LambdaLoss* mencapai yang paling canggih.

Cara tercepat untuk membangun model peringkat / ranking untuk perusahaan, seluruh tim ilmuwan data dan *AI* dibangun untuk membuat dan mempertahankan sistem peringkat. Biaya untuk membangun tim ini tidak praktis untuk sebagian besar bisnis. Baru-baru ini, ada alat hebat yang muncul yang memungkinkan pembuatan dan penyebaran model peringkat dengan mudah—ini dengan sedikit atau tanpa pengalaman pemrograman. Kemungkinan dalam pembangunan dan penyebaran model peringkat tanpa pengetahuan pemrograman *ML* (*Machine Learning*).

Daftar Pustaka

- Broder, A. Z., Carmel, D., Herscovici, M., Soffer, A., & Zien, J. (2003, November). *Efficient query evaluation using a two-level retrieval process*. In Proceedings of the twelfth international conference on Information and knowledge management (pp. 426-434).
- Burges, C. J. (2010). *From ranknet to lambdarank to lambdamart: An overview*. Learning, 11(23-581), 81.
- Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., & Hullender, G. (2005, August). *Learning to rank using gradient descent*. In Proceedings of the 22nd international conference on Machine learning (pp. 89-96).
- Cambazoglu, B. B., Zaragoza, H., Chapelle, O., Chen, J., Liao, C., Zheng, Z., & Degenhardt, J. (2010, February). *Early exit optimizations for additive machine learned ranking systems*. In Proceedings of the third ACM international conference on Web search and data mining (pp. 411-420).
- Duh, K., & Kirchhoff, K. (2008, July). *Learning to rank with partially-labeled data*. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (pp. 251-258).<https://medium.com/@francesco.casalegno>
- Joachims, T. (2002, July). *Optimizing search engines using clickthrough data*. In Proceedings of the eighth ACM SIGKDD International Conference On Knowledge Discovery And Data Mining (pp. 133-142).
- Li, H. (2011). *A short introduction to learning to rank*. IEICE Transactions on Information and Systems, 94(10), 1854-1862.
- Liu, T. Y. (2009). *Learning to rank for information retrieval*. Foundations and Trends® in Information Retrieval, 3(3), 225-331.

- Lv, Y., Moon, T., Kolari, P., Zheng, Z., Wang, X., & Chang, Y. (2011, March). *Learning to model relatedness for news recommendation*. In Proceedings of the 20th international conference on World wide web (pp. 57-66).
- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning*. MIT press.
- Radlinski, F., & Joachims, T. (2005, August). *Query chains: learning to rank from implicit feedback*. In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining (pp. 239-248).
- Rahangdale, A., & Raut, S. (2019). *Machine learning methods for ranking*. International Journal of Software Engineering and Knowledge Engineering, 29(06), 729-761.
- Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39, pp. 234-265). Cambridge: Cambridge University Press.
- Taylor, M., Guiver, J., Robertson, S., & Minka, T. (2008, February). *Sofrank: optimizing non-smooth rank metrics*. In Proceedings of the 2008 International Conference on Web Search and Data Mining (pp. 77-86).
- Wang, X., Li, C., Golbandi, N., Bendersky, M., & Najork, M. (2018, October). *The lambda loss framework for ranking metric optimization*. In Proceedings of the 27th ACM international conference on information and knowledge management (pp. 1313-1322).

PROFIL PENULIS



Satria, S.Kom., M.Kom.

Penulis lahir di Bekasi-Jawa Barat, pada tanggal 19 Mei 1986 dan anak kedua dari tiga bersaudara. Pendidikan formal penulis, Pendidikan S-1 Sistem Informasi 2010 STMIK Bani Saleh Bekasi, S-2 Teknik Informatika (*Software Engineering*) 2013 STMIK Eresha Jakarta. Saat ini, saya bekerja sebagai salah satu dosen tetap jurusan Sistem Informasi di Universitas Bani Saleh sejak 2011 sampai sekarang dan Guru bidang studi Komputer di SMK Al Muhadjirin 2 di Bekasi sejak 2017 sampai sekarang. Penulis sangat gemar tentang Matematika dan mengolah data, serta untuk mengembangkan ilmu yang sudah diperoleh penulis lebih mendalami ilmu secara mandiri dan membaca buku-buku literatur. Matakuliah yang diampuh selama mengajar di Universitas Bani Saleh: *Perancangan Basis data, Aplikasi Basis data, Sistem Terintegrasi, Analisa Perancangan Sistem Informasi, Enterprise Resource Planning (ERP), Supply Chain Management (SCM) dan Customers Relations Management (CRM)*. **Hibah Penelitian:** Penelitian Nasional PDP-Dikti 2019. **Sertifikasi Kompetensi** yaitu *Oracle Academy, SAP 01, dan ToT Data Science* di Kominfo Jakarta. **Focus area penulisan penelitian** : Sistem informasi, *Data Mining, Data Science, Sistem Pengambilan Keputusan*.

Email Penulis: satria1905@gmail.com

BAB 7 REGRESI

M. Lutfi MA, S.T., M.Kom.
STMIK Bina Patria Magelang

Pengantar Pembelajaran Mesin

Pembelajaran mesin adalah salah satu bidang teknis yang paling berkembang pesat saat ini dan merupakan salah satu bidang ilmu komputer yang tumbuh paling cepat. Pembelajaran mesin adalah bidang studi yang membuat dan memasukkan algoritma kedalam komputer sehingga komputer mempunyai kemampuan untuk belajar tanpa diprogram secara *eksplisit*. (Saleh and Layouts, 2022).

Secara definisi, *machine learning* atau pembelajaran mesin adalah ilmu atau studi yang mempelajari tentang algoritma dan model statistik yang digunakan oleh sistem komputer untuk melakukan *task* tertentu tanpa instruksi *eksplisit*. *Machine Learning* adalah subbagian dari kecerdasan buatan dimana mesin dilatih untuk belajar dari pengalaman masa lalu. Pengalaman masa lalu dikembangkan melalui data yang dikumpulkan, kemudian digabungkan dengan algoritma.

Proses pada *Machine Learning* secara sederhana dapat dibagi menjadi 3 bagian:

1. **Decision process**—pada proses ini *machine learning* dapat digunakan untuk membuat prediksi atau klasifikasi melalui pembentukan algoritma tertentu sehingga dapat membuat perkiraan atau keputusan tentang pola suatu data.
2. **Error function**—sistem ini berfungsi untuk mengevaluasi prediksi model data dan membuat perbandingan untuk menilai keakuratan model tersebut pada langkah pertama.
3. **Model optimization process**—algoritma ini akan mengulangi proses evaluasi, pengoptimalan, dan memperbarui bobot secara mandiri sampai akurasi telah terpenuhi.

Linier dapat digunakan untuk menentukan rentang jumlah pada obat mana yang paling banyak keluar dan yang paling sedikit keluar pada bulan tertentu sehingga dapat menghindari kerugian. Berdasarkan hasil pengujian yang dilakukan, terdapat total Obat Masuk 438.614, Obat Keluar 348.001 dan hasil Prediksi 273.355 untuk tahun 2019, dari data obat antibiotik dalam kurun waktu setahun obat *pyrazinamide* Tab 511 adalah obat yang paling banyak memiliki Obat Masuk, Obat Keluar dan Prediksi paling banyak yaitu Obat Masuk sebanyak 9.886, Obat Keluar sebanyak 9.730 dan hasil Prediksi sebanyak 7.754. (Shidiq, Ermawati and Santoni, 2020).

Daftar Pustaka

- Disa, S. (2012), *Implementasi Metode Regresi Linear Dalam Pembuatan Aplikasi Simulasi Sistem Penentuan Kelayakan Lokasi Pembangunan Perumahan*, Jurnal Aspiration, pp. 15–24.
- Ginting, F., Buulolo, E. and Siagian, E.R. (2019), *Implementasi Algoritma Regresi Linear Sederhana Dalam Memprediksi Besaran Pendapatan Daerah (Studi Kasus: Dinas Pendapatan Kab. Deli Serdang, KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, 3(1), pp. 274–279. Available at: <https://doi.org/10.30865/komik.v3i1.1602>.
- Herdiana, Y. (2022), *Penerapan Machine Learning Dengan Model Linear Regression Terhadap Analisis Kualitas Hasil Petik The Di Pt. Perkebunan Nusantara Viii Kebun Sedep*, Jurnal Computing Volume 9, (01), pp. 1-9.
<https://garuda.kemdikbud.go.id/documents/detail/3297578>
- Kurniyawati, et al,(2019), *Analisis Regresi*, Jurusan Matematika Fakultas Sains Dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau pp. 1–57.
- Puteri, K. and Silvanie, A. (2020), *Machine Learning Untuk Model Prediksi Harga Sembako Dengan Metode Regresi Linier Berganda*, Jurnal Nasional Informatika, 1(2), pp. 82–94.
- Saleh, H. and Layous, J.A. (2022) '*Machine Learning – Regression By* ', *Research Gate*, (January), p. 25. Available at: <https://doi.org/10.13140/RG.2.2.35768.67842>.

Shidiq, A.R., Ermawati, I. and Santoni, M.M. (2020), *Penerapan Algoritma Regresi Linier Untuk Prediksi Pada Data Obat Antibiotik*, Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA), Jakarta-Indonesia, 28 Januari 2020, pp. 102-111.`

Tuah, Y.A.E. and Anyan, A. (2020) ,*Implementasi Model Regresi Linear Sederhana Untuk Prediksi Gaji Berdasarkan Pengalaman Lama Bekerja*, JUTECH : Journal Education and Technology, 1(2), pp. 56-70. Available at: <https://doi.org/10.31932/jutech.v1i2.1289>.

PROFIL PENULIS



M.Lutfi MA, ST., M.Kom.

Penulis adalah Dosen Prodi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer Bina Patria, Magelang, Jawa Tengah. Lahir di Magelang, pada tanggal 17 Mei 1973. Penulis menamatkan pendidikan program Sarjana (S1) di Universitas Diponegoro Semarang dan menyelesaikan program Pasca Sarjana (S2) di Universitas AKI Semarang prodi Sistem Informasi. Penulis memiliki ketertarikan dibidang Kecerdasan Buatan, *Data Science* dan UI/UX. Beberapa penelitian yang telah dilakukan baik yang didanai oleh internal perguruan tinggi maupun Kemenristek DIKTI. Selain penelitian, penulis juga aktif menulis buku dengan harapan dapat memberikan kontribusi positif bagi perkembangan ilmu pengetahuan di Indonesia.

Email Penulis: hmlutfima@gmail.com

BAB 8 MODEL *ENTROPI* MAKSIMUM BERSYARAT

Nono Heryana, M.Kom.

Universitas Singaperbangsa Karawang

Pengantar

Model *entropi* maksimum bersyarat adalah salah satu metode yang digunakan dalam *machine learning* untuk memprediksi dan mengklasifikasikan data. Model ini berdasarkan pada prinsip *entropi* dan pembatasan syarat untuk meminimalkan *entropi* dan memaksimalkan informasi dari data. Fokus dari model ini adalah memaksimalkan *entropi* (ketidakpastian) sistem sesuai dengan kondisi bersyarat yang ditentukan. Dalam bidang *machine learning*, model ini sering digunakan untuk memecahkan berbagai jenis masalah seperti klasifikasi, *clustering*, dan lain-lain (Japkowicz & Stephen, 2002).

Latar belakang dari model *entropi* maksimum bersyarat berasal dari prinsip termodinamika yang membahas tentang *entropi* sebagai ukuran ketidakpastian sistem (Shannon, 1948). Dalam bidang *machine learning*, *entropi* digunakan sebagai *metrik* untuk mengukur tingkat ketidakpastian dalam data. Kondisi bersyarat dalam model ini digunakan untuk membatasi *entropi* yang diperoleh sehingga memenuhi kriteria yang ditentukan (Cover & Thomas, 2006).

Menurut (Chen, 2018) Model *entropi* maksimum bersyarat didefinisikan sebagai "Suatu metode *machine learning* yang menggunakan teori *entropi* untuk menentukan distribusi *probabilitas* yang optimal untuk data yang dianalisis, dengan memperhitungkan pembatasan syarat". Sedangkan, Menurut (Liu, 2016), model *entropi* maksimum bersyarat adalah "suatu metode yang menggabungkan teori *entropi* dan pembatasan syarat untuk menentukan distribusi *probabilitas* yang paling baik untuk menjelaskan data yang ada".

```
# melakukan pemodelan
model = MultinomialNB()
model.fit(X_train, y_train)

# melakukan prediksi
y_pred = model.predict(X_test)

# mengevaluasi model
akurasi = accuracy_score(y_test, y_pred)
print("Akurasi:", akurasi)
```

3. Contoh: *Clustering*

Dalam implementasi penerapan model *entropi* maksimum bersyarat untuk *clustering*, beberapa langkah yang perlu dilakukan adalah membaca data, menentukan batasan-batasan yang akan digunakan, dan membuat model dengan memperhatikan batasan-batasan tersebut. Setelah itu, data dapat dikelompokkan dengan memanfaatkan model yang sudah dibuat.

Berikut ini adalah implementasi model *entropi* maksimum bersyarat untuk *clustering* sumber daya manusia menggunakan *Python*:

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans

# membaca data sumber daya manusia
df = pd.read_csv("data_sdm.csv")

# membuat array untuk model clustering
data = df[['usia', 'pendidikan',
'pengalaman_kerja']].values

# membuat objek KMeans
kmeans = KMeans(n_clusters=3)

# melakukan clustering
kmeans.fit(data)

# memasukkan hasil clustering ke dalam data sumber
daya manusia
df['kelompok'] = kmeans.labels_
```

Model Entropi Maksimum Bersyarat

```
# menampilkan hasil clustering
print(df.head())
```

Pada kode di atas, data sumber daya manusia dibaca dari file CSV “data_sdm.csv” menggunakan *library pandas*. Kemudian, data dikelompokkan menjadi tiga kelompok menggunakan objek *KMeans* dari *library scikit-learn*. Hasil *clustering* kemudian ditampilkan dengan menambahkan kolom kelompok pada data sumber daya manusia.

Catatan: Kode di atas hanya sebagai contoh penerapan model entropi maksimum bersyarat masih membutuhkan perbaikan dan perubahan sesuai dengan data dan problem yang dihadapi.

Daftar Pustaka

- Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media, Inc.
- Catlett, M. (2017). *Entropy-based algorithms for best basis selection*. In *Data Compression Conference* (pp. 140-149). Springer, Cham.
- Chen, J. (2018). *Machine learning techniques and applications*. Springer.
- Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). John Wiley & Sons.
- Gan, G., & Klassner, F. (2007). *Entropy-based data clustering: A survey*. *ACM Computing Surveys*, 39(3), 6. <https://doi.org/10.1145/1235167.1235169>
- Japkowicz, N., & Stephen, S. (2002). *The class imbalance problem: A systematic study*. *Intelligent Data Analysis*, 6(5), 429-449. <https://doi.org/10.3233/IDA-2002-6505>
- Kapur, J. N. (1989). *Maximum entropy econometrics: robust estimation with limited data*. John Wiley & Sons.
- Kashyap, R. (2013). *Machine learning algorithms*. In *Data clustering: algorithms and applications* (pp. 1-30). CRC Press.

- Li, X., Lv, Z., Zhang, C., & Liu, T. Y. (2014). *Sentiment classification via maximum entropy models*. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 190-201). Springer, Cham.
- Liu, J. (2016). *Data mining with big data*. Morgan & Claypool Publishers.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. Cambridge, MA: MIT Press.
- Ni, W., Jiang, B., & Ma, Y. (2017). *Maximum Conditional Entropy Model and its Applications in Text Classification*. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 31-42). Springer, Cham.
- Shannon, C. E. (1948). *A mathematical theory of communication*. Bell system technical journal, 27(3), 379-423.
- Sun, X., Shi, Y., & Wei, B. (2016). *A portfolio performance evaluation method based on maximum entropy model*. Mathematical Problems in Engineering, 2016.
- Wu, X. (2005). *Maximum entropy clustering*. Proceedings of the 22nd international conference on Machine learning - ICML '05, 623-630.
- Yang, X., Zhang, C., & Li, Y. (2020). *Image classification using Maximum Conditional Entropy Model based on Convolutional Neural Network*. Journal of Ambient Intelligence and Humanized Computing, 11(6), 4447-4459.

PROFIL PENULIS

Nono Heryana, M.Kom.



Penulis lahir di Lampung Barat tanggal 03 Maret 1989. Menyelesaikan pendidikan S1 pada Jurusan Teknik Informatika di Universitas Singaperbangsa Karawang lulus tahun 2012 dan melanjutkan S2 pada Jurusan Ilmu Komputer di Universitas Budi Luhur lulus tahun 2016. Selain menempuh pendidikan formal penulis juga memiliki beberapa sertifikasi profesi, baik lokal maupun internasional. Seperti: *Google Certified Educator*, *Certified International of Internal Quality Audit (CIIQA)*, *Scrum Master Accredited Certification™ (SMAC)*, *Scrum Foundation Professional Certificate (SFPC)*, *Certified of Editor (C. Ed)*, *Certified Business Operations Associate (CBOA®)* dan *Remote Worker Professional Certificate (RWPC)*. Penulis adalah dosen tetap pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang, sejak tahun 2016 sampai dengan sekarang.

Email Penulis: nonoheryana@gmail.com

BAB 9 REINFORCEMENT LEARNING

Harrizki Arie Pradana, S.Kom., M.T., MTA.
Institut Sains dan Bisnis Atma Luhur

Pendahuluan

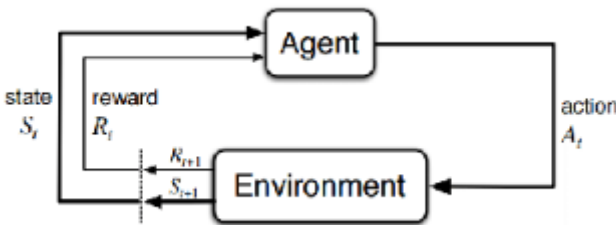
Reinforcement learning merupakan suatu metode yang diterapkan pada *machine learning* agar bisa menentukan aksi yang tepat. Sehingga, sebuah program atau aplikasi pun dapat saling bekerja secara otomatis dan memberikan hasil yang benar (Heuillet et al., 2021). Jika belum paham, simak analogi ini; Misalkan Anda memiliki seekor anjing dan ingin melatihnya dengan cara memberikan suatu instruksi. Jika anjing melakukan perintahnya dengan benar, maka Anda berhak memberikan *rewards* kepada anjing tersebut, begitu juga sebaliknya, demikian juga dengan cara kerja dari *reinforcement learning*.

Cara kerja *reinforcement learning* tentunya berbeda dari *supervised learning* yang pasti memiliki jawaban pasti. Pada metode ini, mesin akan melakukan perintah atau instruksi berdasarkan situasi yang dihadapinya. Karena tidak adanya jawaban yang pasti benar, maka mesin akan belajar dari pengalaman-pengalaman sebelumnya untuk menghindari kesalahan dan mendapatkan *punishment* atasnya (Charpentier et al., 2021).

Machine Learning sendiri merupakan suatu bidang penelitian dari salah satu cabang ilmu pengetahuan yang menggabungkan gagasan dari beberapa cabang ilmu pengetahuan, misalkan seperti kecerdasan buatan, statistik, teori informasi, matematika, dan lainnya (Liu et al., 2021). *Machine learning* secara umum berfokus pada teori, kinerja, sifat sistem pembelajaran dan algoritma (Hernandez-Leal et al., 2019a). *Machine learning* sendiri sering digunakan untuk

4. **State** : adalah suatu situasi yang dikembalikan oleh lingkungan setelah setiap tindakan yang telah dilakukan agen.
5. **Reward** : Merupakan umpan balik yang dikembalikan ke agen dari lingkungan dalam mengevaluasi tindakan suatu agen.
6. **Policy** : adalah strategi yang akan diterapkan oleh agen untuk tindakan selanjutnya.
7. **Value** : Imbalan jangka panjang yang diharapkan dengan diskon dan kebalikan imbalan dari jangka pendek.
8. **Q-value** : sebagian besar mirip dengan *value*, tetapi dibutuhkan satu parameter tambahan sebagai tindakan saat ini.

Adapun lebih jelasnya untuk pemodelan *Reinforcement Learning* bisa dilihat pada gambar 9.1.



Gambar 9.1: Model Reinforcement Learning

Sumber : (Ding et al., 2020)

Karakteristik Reinforcement Learning

Berikut ini adalah karakteristik dari *reinforcement learning* menurut (Ding et al., 2020):

1. Pada *reinforcement learning*, agen tidak diinstruksikan tentang lingkungan dan tindakan apa yang perlu dilakukan.
2. *Reinforcement learning* didasarkan pada proses *hit and trial*.
3. Agen mengambil tindakan berikutnya dan mengubah status sesuai dengan umpan balik dari tindakan sebelumnya.
4. Agen bisa saja mendapatkan imbalan yang tertunda.
5. Lingkungan adalah *stokastik*, dan agen perlu menjelajah untuk mencapainya dan untuk mendapatkan imbalan positif yang maksimal.

yaitu dapat menyampaikan suatu pesan ke agen secara berurutan, dengan menggunakan saluran dua arah (menyediakan pemesanan dengan sedikit kehilangan sinyal), dan saluran *all-to-all*.

MARL sendiri merupakan suatu bidang penelitian yang baru, namun aktif dan sangat berkembang pesat. Hal ini dapat mengintegrasikan hasil dari penguatan penguatan beberapa agen tunggal, teori permainan, dan pencarian langsung di ruang perilaku. Dalam pandangan secara khususnya, kemajuan signifikan dalam bidang *multiagent learning* dapat dicapai dengan cara menggunakan *fertilisasi* silang yang lebih intensif antara bidang pembelajaran mesin, teori permainan, dan teori kontrol (da Silva & Reali Costa, 2019).

Daftar Pustaka

- Buşoniu, L., Babuška, R., & de Schutter, B. (2008). *A comprehensive survey of multiagent reinforcement learning*. In IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews (Vol. 38, Issue 2). <https://doi.org/10.1109/TSMCC.2007.913919>
- Charpentier, A., Élie, R., & Remlinger, C. (2021). *Reinforcement Learning in Economics and Finance*. Computational Economics. <https://doi.org/10.1007/s10614-021-10119-4>
- da Silva, F. L., & Reali Costa, A. H. (2019). *A survey on transfer learning for multiagent reinforcement learning systems*. Journal of Artificial Intelligence Research, 64. <https://doi.org/10.1613/jair.1.11396>
- Ding, Z., Huang, Y., Yuan, H., & Dong, H. (2020). *Introduction to reinforcement learning*. In *Deep Reinforcement Learning: Fundamentals, Research and Applications*. https://doi.org/10.1007/978-981-15-4095-0_2
- Gronauer, S., & Diepold, K. (2022). *Multi-agent deep reinforcement learning: a survey*. Artificial Intelligence Review, 55(2). <https://doi.org/10.1007/s10462-021-09996-w>
- Hernandez-Leal, P., Kartal, B., & Taylor, M. E. (2019a). *A survey and critique of multiagent deep reinforcement learning*. Autonomous

- Agents and Multi-Agent Systems, 33(6).
<https://doi.org/10.1007/s10458-019-09421-1>
- Hernandez-Leal, P., Kartal, B., & Taylor, M. E. (2019b). *Is multiagent deep reinforcement learning the answer or the question?* A brief survey. *Autonomous Agents and Multi-Agent Systems*, 33(6).
- Heuillet, A., Couthouis, F., & Díaz-Rodríguez, N. (2021). *Explainability in deep reinforcement learning*. *Knowledge-Based Systems*, 214.
<https://doi.org/10.1016/j.knosys.2020.106685>
- Levorsen, M., Ito, A., Suzuki, S., & Izuma, K. (2021). *Testing the reinforcement learning hypothesis of social conformity*. *Human Brain Mapping*, 42(5). <https://doi.org/10.1002/hbm.25296>
- Liu, R., Nageotte, F., Zanne, P., de Mathelin, M., & Dresch-Langley, B. (2021). *Deep reinforcement learning for the control of robotic manipulation: A focussed mini-review*. *Robotics*, 10(1).
<https://doi.org/10.3390/robotics10010022>
- Maeda, R., & Mimura, M. (2021). *Automating post-exploitation with deep reinforcement learning*. *Computers and Security*, 100.
<https://doi.org/10.1016/j.cose.2020.102108>
- Mousavi, S. S., Schukat, M., & Howley, E. (2018). *Deep Reinforcement Learning: An Overview*. In *Lecture Notes in Networks and Systems* (Vol. 16). https://doi.org/10.1007/978-3-319-56991-8_32
- Pateria, S., Subagdja, B., Tan, A. H., & Quek, C. (2021). *Hierarchical Reinforcement Learning: A Comprehensive Survey*. In *ACM Computing Surveys* (Vol. 54, Issue 5).
<https://doi.org/10.1145/3453160>
- Recht, B. (2019). *A Tour of Reinforcement Learning: The View from Continuous Control*. In *Annual Review of Control, Robotics, and Autonomous Systems* (Vol. 2). <https://doi.org/10.1146/annurev-control-053018-023825>

- Samieiyeganeh, M., Rahmat, R. W. B. O. K., Khalid, F. B., & bin Kasmiran, K. A. (2022). *Deep Reinforcement Learning To Multi-Agent Deep Reinforcement Learning*. *Journal of Theoretical and Applied Information Technology*, 100(4).
- Varghese, N. V., & Mahmoud, Q. H. (2020). *A survey of multi-task deep reinforcement learning*. In *Electronics (Switzerland)* (Vol. 9, Issue 9). <https://doi.org/10.3390/electronics9091363>
- Zhang, Z., Ong, Y. S., Wang, D., & Xue, B. (2021). *A Collaborative Multiagent Reinforcement Learning Method Based on Policy Gradient Potential*. *IEEE Transactions on Cybernetics*, 51(2). <https://doi.org/10.1109/TCYB.2019.2932203>
- Zhang, Z., Wang, D., & Gao, J. (2021). *Learning Automata-Based Multiagent Reinforcement Learning for Optimization of Cooperative Tasks*. *IEEE Transactions on Neural Networks and Learning Systems*, 32(10). <https://doi.org/10.1109/TNNLS.2020.3025711>

PROFIL PENULIS



Harrizki Arie Pradana, S.Kom., M.T., MTA.

Penulis merupakan kelahiran Kota Pangkalpinang, Provinsi Kepulauan Bangka Belitung. Ketertarikan penulis terhadap ilmu komputer dimulai pada tahun 1995 silam, dimana beliau pertama kali mengenal komputer pada saat masih duduk di bangku SD kelas 3. Hal tersebut membuat penulis memilih untuk masuk ke Sekolah Menengah Umum di SMU Negeri 1 Koba, Kabupaten Bangka Tengah, Provinsi Kepulauan Bangka Belitung. Penulis kemudian melanjutkan pendidikan ke Perguruan Tinggi dan berhasil menyelesaikan studi S1 di prodi Teknik Informatika Universitas Islam Indonesia pada tahun 2012. Dua tahun kemudian, penulis menyelesaikan studi S2 di prodi *Enterprise Information Systems* Program Pasca Sarjana Universitas Atma Jaya Yogyakarta.

Penulis memiliki kepakaran dibidang Kecerdasan Tiruan atau *Artificial Intelligence dan e-Government*. Dan untuk mewujudkan karir sebagai dosen profesional, penulis pun aktif sebagai peneliti dibidang kepakarannya tersebut. Beberapa penelitian yang telah dilakukan didanai oleh internal perguruan tinggi dan juga Kemenristek DIKTI, hingga didanai oleh Luar Negeri. Selain peneliti, penulis juga aktif mengajar sebagai dosen dengan harapan dapat memberikan kontribusi positif bagi bangsa dan negara yang sangat tercinta ini. Atas dedikasi dan kerja keras dalam mengajar, penulis pernah mendapatkan hibah dari pemerintah Jepang dari tahun 2018 hingga saat ini. Penulis saat ini lagi melanjutkan tugas belajar di program S3 jurusan *Computer Science and Information Technology, Universiti Tun Hussein Onn, Malaysia*.

Email Penulis: harrizkiariep@yahoo.com

Dasar-Dasar Pembelajaran Mesin (Foundations of Machine Learning)

Istilah *Machine Learning* mengacu pada deteksi otomatis pola yang bermakna dalam data. Beberapa dekade terakhir ini telah menjadi suatu hal yang umum di hampir semua lini kehidupan yang membutuhkan ekstraksi informasi dari kumpulan data besar. *Machine Learning* banyak digunakan dalam aplikasi ilmiah seperti bioinformatika, kedokteran, dan astronomi. Buku Dasar - Dasar Pembelajaran Mesin (*Foundations of Machine Learning*) ini mencakup perihal terkait pengenalan *Machine Learning*, Kerangka Pembelajaran PAC (*The PAC Learning Framework*), *Support Vector Machine*, Metode Kernel, Pembelajaran *Online*, Ranking, Regresi, Model Entropi Maksimum Bersyarat, dan *Reinforcement Learning*. Semoga kehadiran buku ini mampu untuk menambah wawasan dan ilmu pengetahuan terkait *Machine Learning* serta berkontribusi terhadap ilmu pengetahuan.

