MODUL DATA MINING

FUTURE SELECTION
PERTEMUAN 11 (ONLINE)

Disusun Oleh
Syefira Salsabila

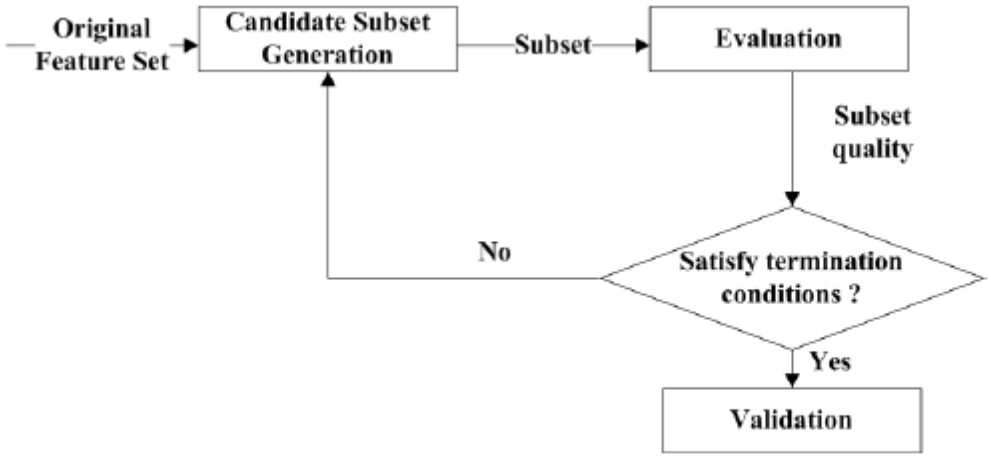
Informasi menjadi salah satu kebutuhan paling dasar manusia dan menjadi komoditi yang penting, dimana saaat ini tak dapat dipungkiri kita sudah berada pada era "*information-based society*". Informasi dapat diartikan suatu data atau objek yang diproses terlebih dahulu sedemikian rupa sehingga dapat tersusun dan terklasifikasi dengan baik, sehingga memiliki arti bagi penerimanya yang selanjutnya menjadi pengetahuan bagi penerima tentang suatu hal tertentu yang membantu pengambilan keputusan secara tepat. Informasi memiliki sifat *integrity, availability* (ketersediaan), dan *confidentiality* (kerahasiaan), dan informasi bagi sebuah perusahaan adalah modal sangat penting. Dari ketiga sifat itu jika ada yang terganggu maka keamanan sistem dan jaringan (*system and network security*) patut diperhatikan dengan seksama dan harus diperbaiki.

Menjadi hal penting yang harus diperhatikan dalam keamanan sistem informasi dan jaringan komputer:

a. Kehilangan data / *data loss*
b. Penyusup / *intruder*

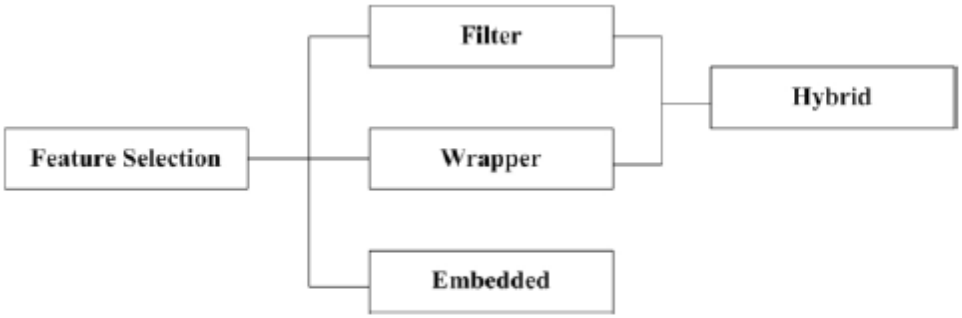
Menurut Bace dan Mell penyusupan/*intrusion* adalah kegiatan yang merusak atau menyalahgunakan sistem atau setiap usaha yang melakukan compromise integritas kepercayaan atau ketersediaan suatu sumber daya komputer dan tidak bergantung pada berhasil atau tidaknya aksi tersebut sehingga ini berkaitan dengan suatu serangan pada sistem komputer.

Seleksi fitur adalah satu dari istilah yang umum digunakan dalam data mining. Digunakan untuk mengurangi input sesuai ukuran yang akan dikelola pada processing dan analisis. Fitur atau atribut pada dataset KDD CUP'99 diselidiki untuk mengindentifikasi relevansi setiap fitur dalam metode induksi. Rule deteksi intrusi digunakan untuk menentukan fitur yang paling diskriminatif untuk masing-masing kelas. Sehingga relevansi dari 41 fitur yang berkaitan dengan label dataset dapat diselidiki.



Gambar 7. Proses seleksi fitur [21]

Ada 4 model utama yang ditetapkan pada seleksi fitur yaitu: metode *wrapper*, metode *filter*, metode *hybrid* dan metode *embedded.*



Gambar 8. 4 metode seleksi fitur [21]

Feature selection adalah suatu metode penganalisaan data yang bertujuan untuk memilih fitur yang berpengaruh (fitur optimal) dan mengesampingkan fitur yang tidak berpengaruh. Ada beberapa algoritma *feature selection* yang dapat digunakan, salah satunya adalah *Relief*. *Relief* memanfaatkan teknik bobot (*weight*) untuk mengukur signifikansi fitur dalam konteks klasifikasi dan fitur yang memiliki nilai bobot di atas ambang batas (*threshold*) yang digunakan akan dipilih.

Suatu objek perlu diketahui fitur-fiturnya agar dapat dikenali dan dibedakan dari objek yang lain. Fitur-fitur optimal yang dapat diketahui dari suatu objek akan mempermudah dan mempercepat proses identifikasi objek tersebut. Fitur atau variabel di dalam penelitian merupakan suatu atribut dari sekelompok objek yang diteliti yang mempunyai variasi antara satu dengan yang lain dalam kelompok tersebut.

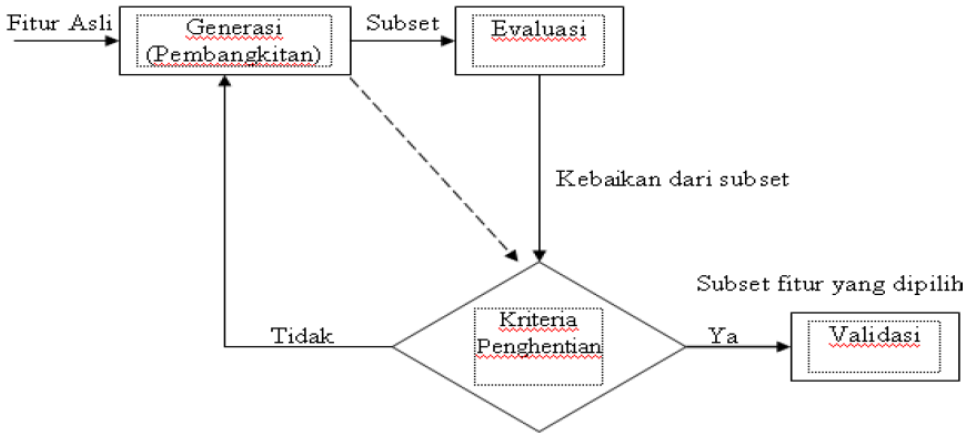
Feature Selection merupakan suatu kegiatan pemodelan atau penganalisaan data yang umumnya dapat dilakukan secara *preprocessing* dan bertujuan untuk memilih fitur yang berpengaruh (fitur optimal) dan mengesampingkan fitur yang tidak berpengaruh. Ada beberapa algoritma *Feature Selection* yang dapat digunakan. untuk menemukan fitur-fitur yang optimal dari sebuah himpunan fitur. Salah satu algoritma *Feature Selection* adalah algoritma *Relief*. *Relief* pertama kali diusulkan oleh Kira dan Rendell pada tahun 1992. *Relief* termasuk dalam metode *Feature Selection* tipe *Filter*, yang didasarkan pada estimasi fitur. *Relief* memberikan nilai yang relevan untuk setiap fitur, dan fitur yang memiliki nilai di atas ambang batas (*threshold)* yang diberikan oleh pengguna yang akan dipilih. Algoritma *Relief* memanfaatkan teknik bobot untuk mengukur signifikansi fitur dalam konteks klasifikasi. Bobot *Relief* adalah nilai-nilai yang kontinu dan memungkinkan fitur untuk digolongkan berdasarkan relevansi. *Relief* juga merupakan algoritma yang menarik dalam *Feature Selection* karena memiliki komputasi yang efisien.

Feature Selection digunakan untuk menemukan subhimpunan dari himpunan fitur yang tersedia untuk meningkatkan aplikasi dari suatu algoritma pembelajaran. *Feature Selection* digunakan dibanyak area aplikasi sebagai alat untuk menghilangkan fitur yang tidak relevan dan atau fitur berlebihan. Sebuah fitur dikatakan tidak relevan jika memberikan sedikit informasi, sedangkan sebuah fitur dikatakan berlebihan jika informasi yang diberikan adalah informasi yang terkandung dalam fitur lain (tidak memberikan informasi baru).

Ada empat langkah yang dilakukan dalam feature selection yaitu:

a. **Prosedur generasi (pembangkitan)**, untuk menghasilkan calon subhimpunan berikutnya dapat dilakukan dengan beberapa cara yaitu : lengkap, heuristik dan acak.
b. **Evaluasi fungsi**, untuk mengevaluasi subhimpunan, dengan cara mengukur jarak, informasi, konsistensi, ketergantungan, dan mengukur tingkat kesalahan klasifikasi.
c. **Kriteria penghentian,** untuk memutuskan kapan harus berhenti, dengan cara melihat nilai ambang batas *(threshold)*, diawali dengan sejumlah pengulangan dan sebuah ukuran subhimpunan fitur terbaik.
d. **Prosedur validasi**, untuk memeriksa apakah subhimpunan valid. (opsional).

Proses dalam feature selection tersebut dapat dituangkan dalam skema berikut:



Gambar 1. Proses *Feature Selection* dengan validasi, (Dash dan Liu, 1997)

Prosedur Generasi

Prosedur generasi merupakan prosedur pencarian yang pada dasarnya menghasilkan *subset* (subhimpunan) dari fitur-fitur untuk dievaluasi. Jika himpunan fitur asli berisi *N* jumlah fitur, maka jumlah calon bersaing untuk menjadi subhimpunan yang dihasilkan adalah *2N*. Ini merupakan jumlah besar bahkan untuk setengah dari jumlah

N. Ada berbagai pendekatan untuk menyelesaikan masalah ini, yaitu: lengkap, heuristik, dan acak.

a. Lengkap
   Urutan ruang pencarian prosedur generasi ini adalah *O*(*2N*), sebuah subhimpunan yang sedikit untuk dievaluasi. Subhimpunan fitur yang optimal sesuai dengan evaluasi fungsi, karena prosedur ini dapat dilakukan dengan cara mundur. Mundur dapat dilakukan dengan menggunakan berbagai teknik, seperti: *branch and bound*, pencarian pertama terbaik, dan balok pencarian.
b. Heuristik
   Dalam setiap pengulangan prosedur generasi ini, semua sisa fitur yang belum dipilih (ditolak) masih dipertimbangkan untuk pemilihan (penolakan). Ada banyak variasi untuk proses sederhana ini, tapi generasi subhimpunan pada dasarnya meningkat atau menurun. Urutan ruang pencarian adalah *O*(*N2*) atau kurang. Prosedur ini sangat sederhana untuk diterapkan dan sangat cepat dalam memperoleh hasil, karena ruang pencarian hanya kuadrat dari jumlah fitur.
c. Acak
   Prosedur generasi ini masih baru dalam penggunaannya dalam metode *Feature Selection* dibandingkan dengan dua kategori lainnya. Meskipun ruang pencarian adalah *O*(*2N*), tetapi metode ini biasanya mencari lebih sedikit jumlah subhimpunan daripada *2N* dengan menetapkan jumlah maksimum pengulangan yang mungkin. Optimalitas subhimpunan yang dipilih tergantung pada sumber daya yang tersedia. Setiap prosedur generasi acak akan memerlukan nilai-nilai dari beberapa parameter.

Evaluasi Fungsi

Evaluasi fungsi mengukur kebaikan subhimpunan yang dihasilkan oleh beberapa prosedur generasi, dan nilai ini dibandingkan dengan yang terbaik sebelumnya. Jika ditemukan yang lebih baik, maka subhimpunan terbaik sebelumnya digantikan. Ada beberapa cara dalam melakukan evalusi fungsi, salah satunya yaitu ukuran Jarak**.**

Juga dikenal sebagai keterpisahan, perbedaan, atau diskriminasi ukuran. Untuk dua kelas, fitur *X* adalah fitur yang lebih disukai dari fitur *Y* apabila *X* menginduksi perbedaan yang lebih besar antara kedua kelas probabilitas kondisional dari *Y* dan jika perbedaan adalah nol, maka *X* dan *Y* tidak dapat dibedakan (sama). Sebagai contoh adalah jarak *Euclidea*. *Euclidean* merupakan metode pengukuran jarak di antara dua objek berdasarkan akar jumlah kuadrat jarak kedua objek. Rumus umum untuk menghitung jarak *Euclidean* yaitu, jika *X* memiliki koordinat (x1, x2,…xn) dan objek *Y* memiliki koordinat (y1, y2, …yn), maka jarak *Euclidean* kedua objek tersebut adalah,

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)}$$

Kriteria Penghentian

Prosedur generasi dan evaluasi fungsi dapat mempengaruhi pilihan untuk kriteria penghentian.

Prosedur Validasi

Proses validasi bukan merupakan bagian dari proses *Feature Selection* itu sendiri, namun sebuah *Feature Selection* harus divalidasi dengan cara melakukan pengulangan terhadap evaluasi fungsi subhimpunan dari fitur sampai kriteria penghentian terpenuhi. Dengan terus bertambahnya jumlah dan keanekaragaman dokumen, penggolongan secara manual tentu saja akan menjadi suatu masalah baru untuk user. Hal tersebut akan memakan banyak waktu dan menimbulkan kejenuhan. Dokumen yang tersebar dan tidak terkoordinasi dengan baik akan menyulitkan user dalam mendapatkan informasi yang diinginkan. Dengan dokumen yang telah terklasifikasi, pengguna informasi (*user*) dapat dengan mudah menemukan dokumen yang dibutuhkan karena dokumen tersebut telah dikelompokkan berdasarkan kategori yang mencerminkan isi dari suatu dokumen. Sekumpulan data biasanya diklasifikasikan dalam single label dengan jumlah atribut yang terbatas.

Untuk meningkatkan efisiensi dan keakuratan dalam klasifikasi dokumen diperlukan teknik *feature selection.* Teknik ini mengurangi jumlah fitur yang ada pada *feature space* dan juga merupakan salah satu pemecahan dalam menangani data *imbalance*. Berawal dari masalah tersebut, maka dalam tugas akhir ini akan dilakukan analisis perbandingan metode *feature selection* untuk menangani data *imbalance* pada suatu klasifikasi dokumen. Diantaranya adalah metode *Odds Ratio* (OR) dan *GSS Coefficient* yang termasuk kedalam *feature selection* menggunakan *onesided metric* dimana *one-sided metric* hanya memilih fitur positif yang berpengaruh pada kelas. Kemudian *Information Gain* (IG) yang termasuk kedalam *two-sided metrics* dimana *two-sided metric* mengkombinasikan secara implisit fitur positif dan fitur negatif. Selain itu metode *improved OR* (iOR) dan *improved SIG* (iSIG) yang termasuk kedalam kombinasi antara fitur positif dan fitur negatif secara eksplisit.

Pendekatan *feature selection* yang akan dilakukan terdiri atas *filtering feature selection* dan *wrapper feature selection*. Pada penerapan *filtering feature selection,* selain menggunakan metode *multinomial naive bayes* juga akan dilakukan proses pengklasifikasian dokumen menggunakan metode yang. Metode *multinomial naive bayes* merupakan algoritme yang *naive* karena mengasumsikan indepedensi di antara kemunculan kata-kata dalam dokumen, tanpa memperhitungkan urutan kata dan informasi konteks dalam kalimat atau dokumen secara umum. Selain itu metode tersebut memperhitungkan jumlah kemunculan kata dalam dokumen.

Kemajuan teknologi dalam penyimpanan data telah mendorong banyak orang untuk berlomba-lomba menyimpan data mereka, baik berupa data pribadi sampai data perusahaan yang besar sekalipun. Hal ini dapat mengakibatkan jumlah data yang semakin meningkat dari hari ke hari. Hal ini dapat menimbulkan masalah. Apabila data

hanya dikumpulkan dan ditumpuk begitu saja, maka data tersebut tidak lebih hanyalah sebuah tumpukan data-data setelah digunakan untuk kepentingan operasional tertentu.

Oleh sebab itu, muncullah apa yang disebut dengan *Data Mining*, yaitu proses menemukan ilmu atau sesuatu yang berguna dari suatu basis data atau kumpulan data yang besar. Pengertian sesuatu yang berguna memiliki kepentingan dan pengertian yang berbeda bagi tiap orang. *Data Mining* memiliki beberapa fungsi atau tugas seperti klasifikasi, klasterisasi, asosiasi, deteksi anomali, dan prediksi.Dalam melaksanakan tugasnya, tentunya setiap fungsi dari *Data Mining* tersebut dihadapkan dengan berbagai tantangan, seperti skalabilitas, dimensionalitas, heterogenitas, *robustness,* kepemilikan dan distribusi data.

Skalabilitas merupakan permasalahan yang berkaitan dengan jumlah data yang sangat besar, dimensionalitas berkaitan dengan jumlah dimensi data yang banyak, heterogenitas mengenai tipe data inputan yang beraneka ragam, *robustness* berkaitan dengan masalah *noise,* data yang tidak lengkap, dll, kepemilikan dan distribusi data berkaitan dengan pendistribusian data berukuran sangat besar sehingga harus dimiliki oleh beberapa basis data maupun media penyimpanan yang berbeda. Beberapa masalah yang dihadapi oleh berbagai fungsi *Data Mining* tersebut banyak berhubungan dengan jumlah data, dimensi data, dan jumlah atribut data yang sangat besar.

Diperlukan suatu mekanisme untuk meningkatkan kinerja fungsi *Data Mining* agar lebih optimal. Fungsi tersebut dibutuhkan sebelum proses *Data Mining* tersebut dijalankan, yaitu proses *preprocessing*. Pada proses *preprocessing* ada banyak cara yang dilakukan agar inputan data yang diterima diolah sedemikian rupa agar fungsi *Data Mining* menjadi lebih optimal, diantaranya: *Feature Selection, Dimension Reduction, Normalization, Data subsetting.* Salah satu langkah yang akan diambil pada Tugas Akhir ini adalah mengenai *Feature Selection. Feature Selection* dapat membantu mengolah data dengan jumlah fitur/atribut yang sangat banyak. Jumlah waktu dan memori yang dibutuhkan oleh *Data Mining* juga akan berkurang dengan pengurangan dimensionalitas.

Feature Selection dapat membantu fungsionalitas dari tugas *Data Mining* seperti Klasifikasi, Klasterisasi, dll. Kerja dari fungsi *Data Mining* akan tidak optimal apabila didalamnya terdapat fitur/atribut yang tidak relevan dan redundan. *Feature Selection* dapat membuang fitur/atribut yang tidak relevan dan yang redundan serta meningkatkan performansi dari tugas *Data Mining*. *Feature Selection* juga banyak dilakukan pada bidang yang lainnya seperti stastika, *image retrieval, pattern recognition,* dll. Metode yang banyak dilakukan pada *Feature Selection* secara umum terbagi menjadi 2, yaitu metode *filter* dan metode *wrapper*. Pada metode *filter* seleksi fitur dilakukan sebelum algoritma *Data Mining* (pada saat ini klasterisasi) dijalankan.

Pendekatan yang dilakukan pada *Feature Selection* ini adalah dengan menggunakan metode *wrapper.* Metode *wrapper* merupakan metode yang menggunakan algoritma dari *Data Mining* target (pada saat ini klasterisasi) sebagai sebuah *black box* untuk menemukan subset atribut terbaik. Ide dasar dari metode ini

adalah melakukan pencarian melalui subset ruang fitur dengan input seluruh fitur dari dataset. Lalu evaluasi setiap subset fitur kandidat dengan pertama mengklaster dengan algoritma klasterisasi dan kemudian mengevaluasi klaster hasil subset fitur dengan menggunakan kriteria *Feature Selection* yang terpilih. Proses ini akan diulang hingga ditemukan subset fitur terbaik dengan klaster yang sesuai berdasarkan kriteria evaluasi fitur. Pendekatan *wrapper* membagi tugas kedalam tiga tahap: pencarian fitur, algoritma klasterisasi, dan evaluasi subset fitur. Pendekatan *wrapper* menjadi menarik karena dilakukan dengan cara menggabungkan algoritme klasterisasi pada pencarian dan pemilihan fitur.

Jumlah informasi pada artikel berbahasa Indonesia berbasiskan web saat ini semakin besar. Besarnya jumlah ini menyebabkan diperlukannya suatu kategorisasi terhadap artikel tersebut untuk memudahkan pembaca dalam mencari topik berita yang mereka inginkan. Salah satu cara yang dapat dilakukan sebagai solusi untuk masalah ini adalah dengan menggunakan proses kategorisasi teks dalam *data mining* yang dapat menggali informasi yang tersembunyi dari informasi-informasi mentah yang ada.

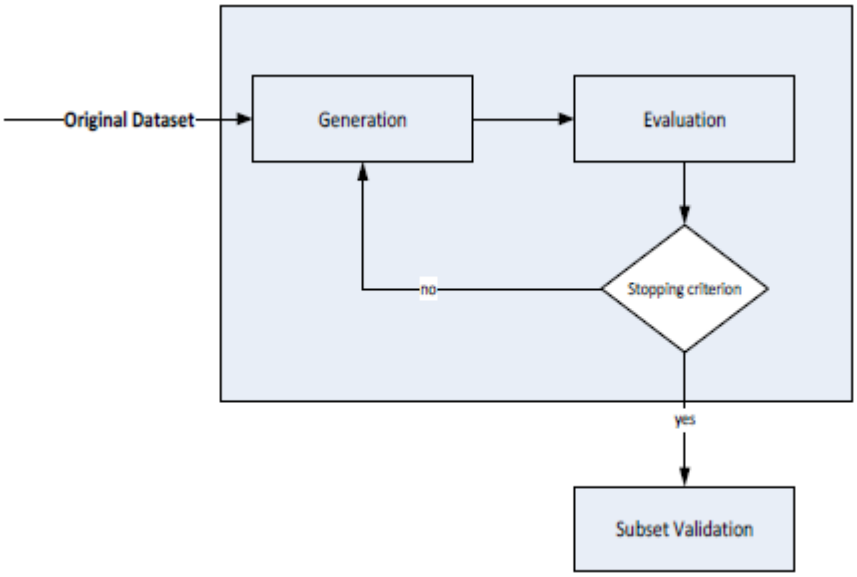
Akan tetapi, tingginya dimensi dari *feature space* data dan adanya data *noise* menjadi masalah utama dalam kategorisasi teks. Hal ini dapat mengganggu efektifitas dari hasil kategorisasinya itu sendiri. Oleh karena itu, harus dilakukan pemilihan terhadap beberapa atribut yang dapat berpengaruh besar terhadap hasil kategorisasi, yaitu *feature selection*, untuk mengurangi tingginya dimensi data berupa manipulasi feature sehingga dapat meningkatkan efektifitas dari *classifier*.

Saat ini, ada banyak *measurement function* dalam proses *feature selection* yang dapat digunakan untuk kategorisasi teks. Beberapa diantaranya yaitu *CHI*, *Information Gain*, *Expected Cross Entropy*, dan *Weight of evidence*. Selain itu, ada pula modifikasi dari *Gini Index* agar dapat digunakan langsung sebagai fungsi pada *text feature selection*.

Dari segi implementasi *feature selection*, ada beberapa pendekatan yang dapat digunakan. Salah satunya adalah *filter-based feature selection*. Ini merupakan teknik pemilihan atribut yang tidak bergantung terhadap *classifier* sehingga hasilnya dapat digunakan oleh algoritma *classifier* manapun bahkan oleh algoritma *classifier* yang kompleks seperti *Neural Network*. Selain itu, komputasi dari pendekatan ini relatif rendah sehingga tidak memakan *cost* yang banyak.

Feature Selection atau seleksi fitur adalah sebuah proses yang biasa digunakan pada Machine Learning dimana sekumpulan dari fitur yang dimiliki oleh data digunakan untuk pembelajaran algoritma. Feature selection telah menjadi bidang penelitian aktif dalam pengenalan pola, statistik, dan Data Mining. Seleksi fitur adalah salah satu faktor yang paling penting yang dapat mempengaruhi tingkat akurasi klasifikasi karena jika dataset berisi sejumlah fitur, dimensi dataset akan menjadi besar hal ini membuat rendahnya nilai akurasi klasifikasi. Masalah dalam seleksi fitur adalah pengurangan dimensi, dimana awalanya semua atribut diperlukan untuk memperoleh akurasi yang maksimal.

Ide utama dari Feature Selection adalah memilih subset dari fitur yang ada tanpa transformasi karena tidak semua fitur/atribut relevan dengan masalah. Bahkan beberapa dari fitur atau atribut tersebut mengganggu dan mengurangi akurasi. Noisy Features atau fitur yang tidak terpakai tersebut harus dihapus untuk meningkatkan akurasi. Selain itu dengan fitur atau atribut yang sangatbanyak akan memperlambat proses komputasi. Berikut gambar tahapan Feature Selection.



Gambar 2.1 Feature Selection

Feature Reduction adalah suatu kegiatan yang umumnya bisa dilakukan secara preprocessing dan bertujuan untuk memilih feature yang berpengaruh dan mengesampingkan feature yang tidak berpengaruh dalam suatu kegiatan pemodelan atau penganalisaan data. Ada banyak alternatif yang bisa digunakan dan harus dicoba-coba untuk mencari yang cocok. Secara garis besar ada dua kelompok besar dalam pelaksanaan feature selection: Ranking Selection dan Subset Selection.

Ranking selection secara khusus memberikan ranking pada setiap feature yang ada dan mengesampingkan feature yang tidak memenuhi standar tertentu. Ranking selection menentukan tingkat ranking secara independent antara satu feature dengan feature yang lainnya. Feature yang mempunyai ranking tinggi akan digunakan dan yang rendah akan dikesampingkan. Ranking selection ini biasanya menggunakan beberapa cara dalam memberikan nilai ranking pada setiap feature misalnya regression, correlation, mutual information dan lain-lain.

Subset selection adalah metode selection yang mencari suatu set dari features yang dianggap sebagai optimal feature. Ada tiga jenis metode yang bisa digunakan yaitu selection dengan tipe wrapper, selection dengan tipe filter dan selection dengan tipe embedded.

Feature Selection Tipe Wrapper: feature selection tipe wrapper ini melakukan feature selection dengan melakukan pemilihan bersamaan dengan pelaksanaan pemodelan. Selection tipe ini menggunakan suatu criterion yang memanfaatkan classification rate dari metode pengklasifikasian/pemodelan yang digunakan. Untuk mengurangi computational cost, proses pemilihan umumnya dilakukan dengan memanfaatkan classification rate dari metode pengklasifikasian/pemodelan untuk pemodelan dengan nilai terendah (misalnya dalam kNN, menggunakan nilai k terendah). Untuk tipe wrapper, perlu untuk terlebih dahulu melakukan feature subset selection sebelum menentukan subset mana yang merupakan subset dengan ranking terbaik. Feature subset selection bisa dilakukan dengan memanfaatkan metode sequential forward selection (dari satu menjadi banyak feature), sequential backward selection (dari banyak menjadi satu), sequential floating selection (bisa dari mana saja), GA, Greedy Search, Hill Climbing, Simulated Annealing, among others.

Feature Selection Tipe Filter: feature selection dengan tipe filter hampir sama dengan selection tipe wrapper dengan menggunakan intrinsic statistical properties dari data. Tipe filter berbeda dari tipe wrapper dalam hal pengkajian feature yang tidak dilakukan bersamaan dengan pemodelan yang dilakukan. Selection ini dilakukan dengan memanfaatkan salah satu dari beberapa jenis filter yang ada. Contohnya: Individual Merit-Base Feature Selection dengan selection criterion: Fisher Criterion, Bhattacharyya, Mahalanobis Distance atau Divergence, Kullback-Leibler Distance, Entropy dan lain-lain. Metode filter ini memilih umumnya dilakukan pada tahapan preprocessing dan mempunyai computational cost yang rendah.

Feature Selection Tipe Embedded: feature selection jenis ini memanfaatkan suatu learning machine dalam proses feature selection. Dalam sistem selection ini, feature secara natural dihilangkan, apabila learning machine menganggap feature tersebut tidak begitu berpengaruh. Beberapa learning machine yang bisa digunakan antara lain: Decision Trees, Random Forests dan lain-lain.

Why Do Feature Selection?

Feature selection is critical to building a good model for several reasons. One is that feature selection implies some degree of *cardinality reduction*, to impose a cutoff on the number of attributes that can be considered when building a model. Data almost always contains more information than is needed to build the model, or the wrong kind of information. For example, you might have a dataset with 500 columns that describe the characteristics of customers; however, if the data in some of the columns is very sparse you would gain very little benefit from adding them to the model, and if some of the columns duplicate each other, using both columns could affect the model.

Not only does feature selection improve the quality of the model, it also makes the process of modeling more efficient. If you use unneeded columns while building a model, more CPU and memory are required during the training process, and more

storage space is required for the completed model. Even if resources were not an issue, you would still want to perform feature selection and identify the best columns, because unneeded columns can degrade the quality of the model in several ways:

a. Noisy or redundant data makes it more difficult to discover meaningful patterns.
b. If the data set is high-dimensional, most data mining algorithms require a much larger training data set.

During the process of feature selection, either the analyst or the modeling tool or algorithm actively selects or discards attributes based on their usefulness for analysis. The analyst might perform feature engineering to add features, and remove or modify existing data, while the machine learning algorithm typically scores columns and validates their usefulness in the model.

In short, feature selection helps solve two problems: having too much data that is of little value, or having too little data that is of high value. Your goal in feature selection should be to identify the minimum number of columns from the data source that are significant in building a model.

How Feature Selection Works in SQL Server Data Mining

Feature selection is always performed before the model is trained. With some algorithms, feature selection techniques are "built-in" so that irrelevant columns are excluded and the best features are automatically discovered. Each algorithm has its own set of default techniques for intelligently applying feature reduction. However, you can also manually set parameters to influence feature selection behavior.

During automatic feature selection, a score is calculated for each attribute, and only the attributes that have the best scores are selected for the model. You can also adjust the threshold for the top scores. SQL Server Data Mining provides multiple methods for calculating these scores, and the exact method that is applied in any model depends on these factors:

a. The algorithm used in your model
b. The data type of the attribute
c. Any parameters that you may have set on your model

Feature selection is applied to inputs, predictable attributes, or to states in a column. When scoring for feature selection is complete, only the attributes and states that the algorithm selects are included in the model-building process and can be used for prediction. If you choose a predictable attribute that does not meet the threshold for feature selection the attribute can still be used for prediction, but the predictions will be based solely on the global statistics that exist in the model.

Feature selection affects only the columns that are used in the model, and has no effect on storage of the mining structure. The columns that you leave out of the mining model are still available in the structure, and data in the mining structure columns will be cached.

Feature Selection Scores

SQL Server Data Mining supports these popular and well-established methods for scoring attributes. The specific method used in any particular algorithm or data set depends on the data types, and the column usage.

a. The *interestingness* score is used to rank and sort attributes in columns that contain nonbinary continuous numeric data.
b. *Shannon's entropy* and two *Bayesian* scores are available for columns that contain discrete and discretized data. However, if the model contains any continuous columns, the interestingness score will be used to assess all input columns, to ensure consistency.

Interestingness score

A feature is interesting if it tells you some useful piece of information. However, *interestingness* can be measured in many ways. *Novelty* might be valuable for outlier detection, but the ability to discriminate between closely related items, or *discriminating weight*, might be more interesting for classification. The measure of interestingness that is used in SQL Server Data Mining is *entropy-based*, meaning that attributes with random distributions have higher entropy and lower information gain; therefore, such attributes are less interesting. The entropy for any particular attribute is compared to the entropy of all other attributes, as follows:

Interestingness(Attribute) = - (m - Entropy(Attribute)) * (m - Entropy(Attribute))

Central entropy, or m, means the entropy of the entire feature set. By subtracting the entropy of the target attribute from the central entropy, you can assess how much information the attribute provides. This score is used by default whenever the column contains nonbinary continuous numeric data.

Shannon's Entropy

Shannon's entropy measures the uncertainty of a random variable for a particular outcome. For example, the entropy of a coin toss can be represented as a function of the probability of it coming up heads. Analysis Services uses the following formula to calculate Shannon's entropy:

$$H(X) = -\sum P(xi) \log(P(xi))$$

This scoring method is available for discrete and discretized attributes.

Bayesian with K2 Prior

SQL Server Data Mining provides two feature selection scores that are based on Bayesian networks. A Bayesian network is a *directed* or *acyclic* graph of states and transitions between states, meaning that some states are always prior to the current state, some states are posterior, and the graph does not repeat or loop. By definition, Bayesian networks allow the use of prior knowledge. However, the question of which prior states to use in calculating probabilities of later states is important for algorithm design, performance, and accuracy.

The K2 algorithm for learning from a Bayesian network was developed by Cooper and Herskovits and is often used in data mining. It is scalable and can analyze multiple variables, but requires ordering on variables used as input. For more information, see Learning Bayesian Networks by Chickering, Geiger, and Heckerman.

This scoring method is available for discrete and discretized attributes.

Bayesian Dirichlet Equivalent with Uniform Prior

The Bayesian Dirichlet Equivalent (BDE) score also uses Bayesian analysis to evaluate a network given a dataset. The BDE scoring method was developed by Heckerman and is based on the BD metric developed by Cooper and Herskovits. The Dirichlet distribution is a multinomial distribution that describes the conditional probability of each variable in the network, and has many properties that are useful for learning.

The Bayesian Dirichlet Equivalent with Uniform Prior (BDEU) method assumes a special case of the Dirichlet distribution, in which a mathematical constant is used to create a fixed or uniform distribution of prior states. The BDE score also assumes likelihood equivalence, which means that the data cannot be expected to discriminate equivalent structures. In other words, if the score for If A Then B is the same as the score for If B Then A, the structures cannot be distinguished based on the data, and causation cannot be inferred.

For more information about Bayesian networks and the implementation of these scoring methods, see Learning Bayesian Networks.

Feature Selection Methods per Algorithm

The following table lists the algorithms that support feature selection, the feature selection methods used by the algorithm, and the parameters that you set to control feature selection behavior:

Algorithm	Method of analysis	Comments
Naive Bayes	Shannon's Entropy Bayesian with K2 Prior Bayesian Dirichlet with uniform prior (default)	The Microsoft Naïve Bayes algorithm accepts only discrete or discretized attributes; therefore, it cannot use the interestingness score. For more information about this algorithm, see Microsoft Naive Bayes Algorithm Technical Reference.
Decision trees	Interestingness score Shannon's Entropy Bayesian with K2 Prior Bayesian Dirichlet with uniform prior (default)	If any columns contain non-binary continuous values, the interestingness score is used for all columns, to ensure consistency. Otherwise, the default feature selection method is used, or the method that you specified when you created the model. For more information about this algorithm, see Microsoft Decision Trees Algorithm Technical Reference.
Neural network	Interestingness score Shannon's Entropy Bayesian with K2 Prior Bayesian Dirichlet with uniform prior (default)	The Microsoft Neural Networks algorithm can use both Bayesian and entropy-based methods, as long as the data contains continuous columns. For more information about this algorithm, see Microsoft Neural Network Algorithm Technical Reference.
Logistic	Interestingness	Although the Microsoft Logistic Regression algorithm is

Algorithm	Method of analysis	Comments
regression	score Shannon's Entropy Bayesian with K2 Prior Bayesian Dirichlet with uniform prior (default)	based on the Microsoft Neural Network algorithm, you cannot customize logistic regression models to control feature selection behavior; therefore, feature selection always default to the method that is most appropriate for the attribute. If all attributes are discrete or discretized, the default is BDEU. For more information about this algorithm, see Microsoft Logistic Regression Algorithm Technical Reference.
Clustering	Interestingness score	The Microsoft Clustering algorithm can use discrete or discretized data. However, because the score of each attribute is calculated as a distance and is represented as a continuous number, the interestingness score must be used. For more information about this algorithm, see Microsoft Clustering Algorithm Technical Reference.
Linear regression	Interestingness score	The Microsoft Linear Regression algorithm can only use the interestingness score, because it only supports continuous columns. For more information about this algorithm, see Microsoft Linear Regression Algorithm Technical Reference.
Association rules Sequence clustering	Not used	Feature selection is not invoked with these algorithms. However, you can control the behavior of the algorithm and reduce the size of input data if necessary by setting the value of the parameters MINIMUM_SUPPORT and MINIMUM_PROBABILIITY. For more information, see Microsoft Association Algorithm Technical Reference and Microsoft Sequence Clustering Algorithm Technical Reference.
Time series	Not used	Feature selection does not apply to time series models. For more information about this algorithm, see Microsoft Time Series Algorithm Technical Reference.

Feature selection merupakan bagian penting untuk mengoptimalkan kinerja dari *classifier. Feature selection* dapat didasarkan pada pengurangan ruang fitur yang

besar, misalnya dengan mengeliminasi atribut yang kurang relevan. Penggunaan algoritma *feature selection* yang tepat dapat meningkatkan *accuracy*. Algoritma *feature selection* dapat dibedakan menjadi dua tipe, yaitu *filter* dan *wrapper*. Contoh dari tipe *filter* adalah *information gain* (IG), *chi-square*, dan *log likelihood ratio*. Contoh dari tipe *wrapper* adalah *forward selection* dan *backward elimination.* Hasil *precision* dari tipe *wrapper* lebih tinggi daripada tipe *filter*, tetapi hasil ini tercapai dengan tingkat kompleksitas yang besar. Masalah kompleksitas yang tinggi juga dapat menimbulkan masalah.

DAFTAR PUSTAKA

J.Kittler, “Feature Selection & Extraction”, in Handbook of Pattern Recognition and Image Processing, Tzay Y. Young, King Sun Fu Ed. Academic Press, 1986.

Koncz, P., & Paralic, J. (2011). An approach to feature selection for sentiment analysis. In *2011 15th IEEE International Conference on Intelligent Engineering Systems* (pp. 357–362). IEEE. doi:10.1109/INES.2011.5954773

Wang, S., Li, D., Song, X., Wei, Y., & Li, H. (2011). A feature selection method based on improved fisher’s discriminant ratio for text sentiment classification. *Expert Systems with Applications*, *38*(7), 8696–8702. doi:10.1016/j.eswa.2011.01.077