BAB II

TINJAUAN PUSTAKA

2.1 Aplikasi Halodoc

Halodoc adalah ekosistem layanan kesehatan terpadu dan terintegrasi hasil gabungan dua aplikasi kesehatan Labconx dan ApotixAntar, yang memungkinkan pengguna aplikasi mendapatkan akses terhadap layanan kesehatan secara mudah. Dengan memanfaatkan aplikasi Halodoc, pengguna dapat mengakses layanan kesehatan yang sebelumnya berbelit-belit menjadi dan juga menghubungkan praktisi kesehatan, seperti dokter, apotek, dan laboratorium langsung ke tangan pasien. Dengan aplikasi ini, seseorang bisa mencari dokter untuk berkonsultasi mengenai kondisi kesehatan yang dialami. Selanjutnya, setelah dokter diagnosis penyakit atau gejala yang diderita oleh pasien, akan diberikan resep obat yang sesuai.

Adapun pengguna bisa mendapatkan layanan sebagai berikut:

1. Layanan bicara dengan dokter

   Layanan ini memfasilitasi para dokter rekanan sebagai penyedia layanan dalam berinteraksi dengan pelanggan/pengguna melalui video call, voice call maupun chat yang dapat diakses melalui aplikasi dan website. Pelanggan/pengguna cukup memberikan informasi dan menjelaskan gejala atau keluhan fisik yang dialami secara jelas dan akurat ketika melakukan percakapan dengan dokter melalui fitur ini.

2. Layanan beli obat

   Layanan beli obat Halodoc terkoneksi dengan fitur GOMED pada aplikasi GOJEK sebagai pihak ketiga yang mengantarkan pesanan obat bebas (dot hijau), obat bebas terbatas (dot biru), obat keras (dot merah) dengan resep dokter, vitamin, dan alat kesehatan bukan obat, makanan dan minuman sehat

3. Layanan Lab

   Halodoc juga menyediakan layanan pengujian lab. Pengguna dapat dengan mudah memsan jasa tes labolatorium dari labolatorium resmi yang sudah bekerjasama.

4. Layanan Periksa Rumah Sakit

   Pengguna dengan membuat janji dengan dokter sehingga tanpa perlu antri di administrasi rumah sakit.

2.2 Data Mining

Secara sederhana *data mining* adalah penambangan atau penemuan informasi baru dengan mencari pola atau aturan tertentu dari sejumlah data yang sangat besar (Davies ddk, 2004). *Data mining* juga disebut sebagai serangkaian proses untuk menggali nilai tambah berupa pengetahuan yang selama ini tidak diketahui secara manual dari suatu kumpulan data. Menurut Macleman dkk (2009), fungsi *data mining* dibagi menjadi sebagai berikut:

1. *Classification*

   Classification untuk mencari model atau fungsi yang menggambarkan dan membedakan kelas-kelas atau konsep data untuk mengklasifikasikan target

class ke dalam kategori yang dipilih disebut *classification*. Dalam klasifikasi, terdapat target variabel kategori, misalnya pendapatan tinggi, sedang, dan rendah.

2. *Clustering*

   Clustering berguna untuk mencari pengelompokkan atribut ke dalam segmentasi-segmentasi berdasarkan similaritas. *Cluster* berbeda dengan *classification* karena tidak adanya variabel target dalam *cluster*. Algoritma pengklusteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok yang memiliki kemiripan *record* dalam satu kelompok akan bernilai maksimal, sedangkan kemiripan dengan *record* dalam kelompok lain akan bernilai maksimal.

3. *Association*

   Association berguna untuk mencari keterkaitan antara atribut atau item set, berdasarkan jumlah item yang muncul dari *association rule* yang ada. Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam satu waktu.

4. *Regression*

   Regression berguna untuk mencari prediksi dari suatu pola yang ada, fungsinya hamper menyerupai dengan *classification.*

5. *Forecasting*

   Forecasting berguna untuk peramalan waktu yang akan datang berdasarkan trend yang telah terjadi di waktu sebelumnya.

6. *Sequence Analysis*

   Sequence analysis berguna untuk mencari pola urutan dari ringkasan kejadian.

7. *Deviation Analysis*

   Deviation analysis berguna untuk mencari kejadian abnormal yang sangat berbeda dari keadaan umumnya.

2.3 *Text Mining*

Menurut Feldman & Sanger (2007), *text mining* merupakan proses penggalinan informasi secara insetif yang bekerja menggunakan alat dan metode tertentu untuk menganalisis suatu kumpulan dokumen. *Text mining* digunakan untuk mendeskripsikan sebuah kronologi yang mampu menganalisis data teks semi-terstruktur maupun tidak terstruktur, sedangkan *data mining* digunakan dalam pengolahan data yang bersifat terstruktur.

Adapun yang paling membedakan antara *text mining* dan *data mining* berada pada sumber daya yang digunakan. Kesamaan antara keduanya yakni menggunakan data besar dan data berdimensi tinggi dengan struktur yang terus berubah. Pada *text mining*, pola-pola yang diekstrak dari data tekstual yang tidak terstruktur. Sedangkan pada *data mining*, data yang diolah umumnya sudah terstruktur dari proses *warehousing*. Sehingga *text mining* biasanya lebih sulit dara *data mining* karena berkaitan langsung dengan masyarakat dimana memiliki struktur teks yang kompleks, struktur yang tidak lengkap, Bahasa yang berbeda, dan arti yang tidak standar. Maka dari itu digunakan *Natural Language Processing*

untuk analisis texks yang tidak berstruktur tersebut. Secara umum tahap-tahap pada *text mining* dapat dibagi atas *text preprocessing* dan *feature selection.*

2.3.1 Text Preprocessing

Dalam proses *text mining,* teks dokumen yang digunakan harus dipersiapkan terlebih dahulu sebelum dapat digunakan untuk proses utama. Proses mempersiapkan teks dokumen atau dataset mentah disebut juga dengan proses *text preprocessing. Text preprocrssing* berfungsi untuk mengubah data teks yang tidak terstruktur atau sembarang menjadi data yang terstruktur. Secara umum proses yang dilakukan dalam tahapan preprocessing adalah sebagai berikut

1. *Spelling Normalization*

   Spelling Normalization adalah proses substitusi atau perbaikan kata-kata singkatan atau salah ejaan. Substitusi kata dilakukan untuk menghindari jumlah perhitungan dimensi kata yang melebar. Perhitungan dimensi kata akan melebar jika kata yang salah eja atau disingkat tidak diubah karena kata tersebut sebenarnya mempunyai maksud dan arti yang sama tetapi akan dianggap sebagai entitas yang berbeda pada saat proses penyusunan matriks.

2. *Case Folding*

   Case Folding adalah proses pelayanan *case* dalam sebuah dokumen. Hal ini dilakukan untuk mempermudah pencarian. Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu peran *case folding*

dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (dalam hal ini huruf kecil atau *lowercase*).

3. *Tokenizing*

   Tokenizing adalah proses penguraian teks yang semula berupa kalimat-kalimat yang berisi kata-kata. Proses tokenisasi diawali dengan menghilangkan *deliminter-deliminter* yaitu symbol dan tanda baca yang ada pada teks tersebut sepert @, $, &, tanda titi (.), koma (,), tanda Tanya (?), tanda seru (!). Proses pemotongan *string* berdasarkan tiap kata yang menyusunnya, umumnya setiap kata akan terpisahkan dengan karakter spasi, proses terkoneksi mengandalkan karakter spasi pada dokumen teks untuk melakukan pemisahan. Hasil dari proses ini adalah kumpulan kata saja.

4. *Filtering*

   Filtering adalah proses mengambil kata-kata pentig dari hasil token. Algoritma *stoplist/stopword* (membuang kata yang kurang penting) atau *wordlist* (menyimpan kata yang penting) dapat digunakan pada tahap ini. *Stopword* adalah kata-kata yang tidak deskriptif dan bukan merupakan kata penting dari suatu dokumen sehingga dapat dibuang. Contoh *stopword* adalah “yang”, “dan”, “dari” dan sterusnya. Dalam filterasi ini menggunakan *stopword* agar kata-kata kurang penting dan sering muncul dalam suatu dokumen dibunag, sehingga hanya menyisakan kata-kata yang menyisakan kata-kata yang penting dan mempunyai arti yang diproses ke tahap selanjutnya.

2.3.2 *Feature Selection*

Feature Selection merupakan tahap selanjutnya dari proses pengurangan dimensi. Meskipun pada tahap sebelumnya telah dilakukan penghapusan terhadap kata-kata yang tidak deskriptif (*stopword*), tidak seluruh kata-kata yang terdapat dalam dokumen memiliki makna penting. Pada tahap ini dilakukan pemilihan terhadap kata-kata yang relevan dan benar-benar mempresentasikan dari dokumen. Pemilihan dilakuan dengan melihat kata-kata yang memiliki intensitas kemunculan tinggi pada dokumen, serta kata-kata yang interatif secara keseluruhan.

2.4 Pembobotan Kata (*Term Weigthing)*

Pembobotan kata (*term weighting)* merupakan salah satu tahapan yang perlu diperhatikan dalam mencari informasi dari koneksi dokumen yang heterogen. Dalam dokumen umumnya terdapat kata. Frase, atau unit indeks lainnya yang menunjukkan konteks dari dokumen tersebut, hal inilah yang disebut *term. Term weigthing* digunakan untuk memberiknan indikator dan setiap kata sesuai dengan tingkat kepentingan masing-masing kata dalam dokumen. Salah satu metode pembobot *term* terbaru yang paling banyak digunakan adalah metode *Term Frequency – Inverse Document Frequency* (TF-IDF). Dalam perhitungan TF-IDF, perhitungan bobot *term* dari sebuah doukmen dilakukan dengan menghitung masing-maing niali *Term Frequency* dan *Inverse Document Frequency*.

Menurut Zafikri (2008), perhitungan nilai TF-IDF dapat dilakukan dengan menggunakan rumus berikut:

1. *Term Frequency* (TF)

   Term Frequency merupakan faktor yang menentukan perhitungan bobot *term* berdasarkan jumlah dan bentuk kemunculan kata pada dokumen. Pada dasarnya dapat dikatakan bahwa semakin besar nilai jumlah kemunculan suatu term, maka semakin besar juga niali bobot term tersebut dalam dokumen. Adapun perhitungan nilai *term frequency* dapat dilakukan dengan beberapa cara sebagai berikut:

   1) TF biner, pemberian bobot *term* dilihat berdasarkan ada tidaknya suatu kata dalam dokumen. Jika terdapat kata tersebut maka diberi nilai satu, jika tidak diberi nilai nol.
   2) TF murni atau *raw TF*, pemberian bobot *term* dilihat berdasarkan jumlah kemunculan suatu kata dalam dokumen. Missal jika kata tersebut muncul tiga kali maka akan diberi bobot tiga.
   3) TF logaritmit, pemberian bobot *term* pada dokumen yang memiliki sedikit kata dalam *query*, namun mempunyai frekuensi yang tinggi.

$$tf = 1 + log\,(tf) \tag{2.1}$$

   4) TF normalisasi, pemberian bobot *term* diperoleh dengan membandingkan frekuensi sebuah kata dengan jumlah seluruh kata dalam dokumen

$$tf = 0{,}5 + 0{,}05\text{x}\,(\frac{tf}{\max tf}) \tag{2.2}$$

2. *Inverse Document Frequency* (IDF)

Inverse Document Frequency merupakan proses mengurangi dominasi *common term* yang sering muncul dalam dokumen, *common term* perlu dihilangkan umumnya kurang bernilai sehingga sering menyebabkan analisis kurang maksimal. Selain itu, IDF juga bertujuan untuk menjaga faktor dengan menghitung niali faktor kebalikan dari frekuensi dokumen yang mempunyai suatu kata. Adapun perhitungan nilai *inverse Document Frequency* dapat dilakukan dengan

$$idf_j = log\ (\frac{D}{df_j}) \quad (2.3)$$

dimana

D : jumlah keseluruhan dokumen

df_j : jumlah dokumen yang mempunyai *term* t_j.

Adapun niali TF-IDF diperoleh dari perkalian nilai *Term Frequency* dengan nilai *Inverse Document Frequency*. Maka pada perhitungan TF-IDF untuk *raw TF* digunakan rumus sebagai berikut:

$$W_{ij} = tf_{ij} \text{ x } idf_j \quad (2.4)$$

$$W_{ij} = tf_{ij} \text{ x } log\ \frac{D}{df_j} \quad (2.5)$$

Dengan

W_{ij} : bobot *term* t_j terhadap dokumen d_i

tf_{ij} : jumalah kemunculan *term* t_j dalam dokumen d_i

2.5 Analisis Sentimen

Menurut Lee dan Pang (2008), analisis sentiment merupakan proses memperoleh informasi dengan cara memahami, mengekstrak, dan mengolah data tekstual secara otomatis. Analisis sentiment mulai terkenal pada tahun 2013 sebagai salah satu cabang *text mining* dan juga dikenal dengan *opinion mining*. Pada dasarnya, analisis sentimen digunakan untuk mengetahui tanggapan dan sikap dari suatu kelompok atau individu terhadap suatu topik bahasan kontekstual keseluruhan dokumen. Tanggapan dan sikap tersebut dapat berupa pendapat, penilaian atau evaluasi (teori appraisal), keadaan efektif (keadaan emosional penulis saat menulis) atau komunikasi emosional (efek emosional yang sampai pada pembaca).

Secara umum domain yang sering membutuhkan analisis sentiment antara lain produk konsumen, layanan dan jasa, peristiwa social dan politik yang memerlukan opini publik. Analisis sentiment lebih memiliki kecenderungan terhadap penelitian mengenai peenyataan suatu pendapat yang mempunyai suatu sentiment baik positif atau negatif. Pendapat memiliki pengaruh yang tinggi kepada perilaku seseorang, maka dapat dikatakan semua aktivitas seseorang terwakili dari pendapat tersebut. Hal ini terlihat dalam proses pengambilan

keputusan dimana umumnya diambil dari pendapat orang-orang. Pada sektor bisnis dan organisasi, pendapat dan opini publik menjadi sangat penting terhadap penilaian suatu produk dan jasa.

Bagi sektor bisnis, analisis sentiment dapat berguna dalam proses pelacakan produk, jasa, merek, dan target konsumen di pasar. Selain itu analisis sentimen juga dapat menilai keunggulan dan kelemahan suatu produk dan jasa. Secara umum analisis sentimen digunakan untuk mendeteksi keluhan, persepsi produk atau layanan baru, dan persepsi dari suatu merek tertentu.

2.6 Klasifikasi

Menurut Prasetyo (2012), klasifikasi adalah proses pengelompokkan teramati dari suatu objek data ke dalam suatu kelas tertentu berdasarkan kelas-kelas yang ada. Teknik klasifikasi lebih efektif digunakan dalam proses prediksi dan penggambaran suatu kumpulan data untuk jenis kategori biner atau nominal dibandingkan dengan kategori ordinal. Contohnya klasifikasi lebih cocok dalam mengklasifikasi seseorang dengan penghasilan dan tidak berpenghasilan dibandingkan mengklasifikasikan seseorang berpenghasilan rendah, menengah, dan tinggi.

Menurut Ham dan Kamber (2006), data klasifikasi strategi menjadi dua proses tahapan. Tahap pertama merupakan *learned model* dimana dilakukan pembangunan model hasil analisis *record database* dari serangkaian kelas data yang ada. Masing-masing *record* diasumsikan mempunyai *predefined class* yang didasarkan pada atribut kelas label, karena masing-masing *record* memiliki kelas

label maka klasifikasi termasuk ke dalam *supervied learning*. Hal inilah yang membedakan antara klasifikasi dan *clustering learning* yang tidak memerlukan kelas label (*unsupervised learning*). Tahap ini juga sering disebut dengan tahap pembelajaran atau pelatihan. Pelatihan dilakukan dengan menganalisis data latih hingga diperoleh informasi yang dibutuhkan untuk membangun suatu model algoritma klasifikasi. Proses pembangunan tersebut dapat dilihat sebagai proses pembentukan dan penataan fungsi $y = f(x)$ dengan *y* yaitu kelas label hasil prediksi dan x yaitu *record* yang akan diprediksikan.

Menurut Ham dan Kamber (2006), untuk memperoleh hasil klasifikasi yang baik diperlukan beberapa persiapan sebagai berikut:

1. Pembersihan data

   Pembersihan data digunkan untuk mengurangi kecacatan data terutama dalam proses pembangunan model. Pembersihan yang bisa dilakukan adalah menghilangkan data noise, melengkapi data yang hilang, dan seterusnya.

2. Analisa relevansi

   Dalam proses klasifikasi terdapat atribut-atribut yang memiliki tingkat kemampuan tinggi dan sering kali saling berhubungan kuat satu dengan lainnya. Sehingga atribut-atribut tersebut perlu dihilangkan agar tidak mempengaruhi tingkat keoptimalan klasifikasi.

2.6.1 Ukuran Evaluasi Model Klasifikasi

Proses evaluasi dilakukan dengan menghitung suatu ukuran tertentu terhadap himpunan data uji, yakni data yang tidak digunakan dalam proses pembuatan model klasifikasi tersebut. *Confusion matrix* merupakan matrix yang berisi informasi mengenai klasifikasi actual yang akan diprediksi oleh sistem klasifikasi (Kohavi & Provost, 1998). Sistem klasifikasi dibentuk dari pemetaan suatu baris data dan *output* suatu hasil prediksi kelas dari data tersebut. Pada suatu klasifikasi baris data dapat menghasilkan empat kemungkinan yang digunakan untuk menilai dan mengevaluasi proses klasifikasi. Apabila data positif dan tepat di prediksi positif maka disebut *true positive*, namun jika salah dan terprediksi negatif maka disebut *false negative*. Apabila data negative dan tepat diprediksi negatif maka disebut *true negative*, namun jika salah dan terprediksi positif maka disebut *false positive*

Tabel 2.1 *Confision Matrix*

Prediksi	Kelas	Positif	Negatif
	Positif	*True Positive (TF)*	*False Negative (FN)*
	Negatif	*False Positive (FP)*	*True Negative (TN)*

Adapun ukuran yang umumnya digunakan dalam penelitian dan evaluasi model klasifikasi sebagai berikut:

1. Accuracy

   Akurasi adalah jumlah proporsi prediksi yang benar. Akurasi digunakan sebagai tingkat ketepatan antara nilai actual dengan nilai prediksi. Adapun rumus perhitungan akurasi dapat dilihat pada persamaan berikut:

$$Accuracy = \frac{TP+FN}{TP+FP+TN+FN} \quad (2.6)$$

2. *Precision*

   Precision adalah proporsi jumlah dokumen teks yang relevan terkenali diantara semua dokumen teks yang terpilih oleh sistem. *Precision* digunakan sebagai tingkat ketepatan antara informasi yang diminta dengan jawaban yang diberikan oleh sistem. Rumus *precision* dapat dilihat pada perumusan sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \quad (2.7)$$

3. *Recall*

   Recall adalah proporsi jumlah dokumen teks yang relevan terkenali diantara semua dokumen teks relevan yang ada pada koleksi. Nilai *recall* untuk kelas positif disebut juga dengan nilai *sensitivity*, sedangkan untuk kelas negatif disebut dengan *specificity*. *Recall* digunakan sebagai ukuran keberhasilan sistem dalam menemukan kembali informasi. Rumus *recall* dapat dilihat pada perumusan sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \quad (3.8)$$

4. *F-Measure*

F-Measure merupakan rata-rata harmonis dari nilai *recall* dan nilai *precision* sehingga dapat memberikan penilaian kinerja yang lebih seimbang. *F-measure* digunakan untuk mengatur kinerja sistem secara menyeluruh dalam pengklasifian. Rumus *F-measure* dapat dilihat pada persamaan berikut:

$$F-measure = \frac{2\,(recall\ x\ precision)}{recall+precision} \quad (3.9)$$

2.6.2 *K-FOLD Cross Validations*

Cross validation adalah teknik validasi tingkat keakuratan sebuah model dari suatu dataset tertentu. Dataset terbagi menjadi data latih sebagai data yang digunakan untuk membangun model dan data uji sebagai data yang digunakan untuk memvalidasi model tersebut. Model klasifikasi digunakan dalam proses klasifikasi atau prediksi suatu data baru yang tidak termasuk dalam data pembangunan model.

Menurut Refaeilzadeh dkk (2009), salah satu metode dari *cross validation* yang umumnya digunakan dalam perhitungan akurasi prediksi suatu sistem adalah *K-fold cross validation.* Proses dalam *K-fold cross validation* dilakukan dengan membagi dataset menjadi *K* segmen yang hamper sama ukuran proporsinya. Kemudian salah satu segmen *K* diambil sebagai data uji sedangkan *K-I* segmen

lainnya digunakan sebagai data latih dari pembentukan model baru. Proses pelatihan dan penilaian ini dilakukan sebanyak *K* kali iterasi. Nilai *K-fold cross validation* diperoleh dari rata-rata hasil iterasi yang dilakukan. Jumlah *K* yang umumnya digunakan dalam *K-fold cross validation* yakni 5. 7. 10 dan 15. Adapun simulasi untuk *K-fold cross validation* untuk *K=5* sebagai berikut:

"Dataset = K1,K2,K3,K4,K5"

Iterasi	Data latih	Data uji
1	K2,K3,K4,K5	K1
2	K1,K3,K4,K5	K2
3	K1,K2,K4,K5	K3
4	K1,K2,K3,K5	K4
5	K1,K2,K3,K4	K5

Berdasarkan tabel *K-fold cross validation* untuk *K =5* sehingga dilakukan iterasi sebanyak 5 kali. Iterasi dilakukan dengan mengambil data untuk segmen uji (1 segmen) dan segmen lainnya (4 segmen) pada tiap iterasinya (5 kali iterasi). Apabila terdapat 1000 data, untuk *K=5* maka per-segmen berjumlah masing-masing 200 data. Apabila *K=10* dengan dataset sebanyak 1000 data maka data latih berjumlah 900 data sedangkan untuk data uji sejumlah 100 data.

2.7 Maximum Entropy

Menurut Nigam (1999), Maximum Entropy merupakan salah satu *machine learning* yang menggunakan proses pengestimasian probabilitas distribusi dalam pengklasifikasian data. Dalam metode *Maximum Entropy* dinyatakan bahwa untuk dataset yang tidak diketahui informasi mengenai distribusinya, maka data tersebut akan diasumsikan untuk mengestimasi berbagai *natural language* taks seperti *language modelling* pelabelan *part of speech* dan segmentasi pada teks lainnya.

Menurut Anggreini (2008), *Maximum Entropy* adalah metode klasifikasi berbasiskan probabilitas yang termasuk dalam kelas model eksponensial. Prinsip dari *Maximum Entropy* didasarkan pada distribusi *p(a|b)* yang akan memberikan nilai *entropy* maksimum. *Maximum Entropy* didefinisikan sebagai rata-rata nilai informasi yang maksimum untuk suatu himpunan kejadian X dengan distribusi nilai probabilitas yang seragam. Distribusi nilai probabilitas seragam dimaksud adalah distribusi yang menggunakan faktor ketidakpastian yang minimum atau dapat disebut sebagai distribusi yang memakai asumsi seminimal mungkin. Dengan menggunakan asumsi yang minimal, maksimal distribusi yang didapatkan merupakan distribusi yang paling mendekati kenyataan. Pencarian disribusi probabilitas yang paling memberikan nilai *entropy* yang maksimum dilakukan dengan tujuan mencari probabilitas terbaik.

2.7.1 Definisi *Entropy*

Entropy merupakan rata-rata dari himpunan informasi yang terdapat dalam suatu kumpulan kejadian $x = (x1, x2 \dots xn)$. Himpunan informasi yang terdapat pada suatu kejadian dapat dinyatakan dengan

$$h(x) = log \frac{1}{p(x)} \quad (3.10)$$

dimana $h(x)$ adalah himpunan informasi dari suatu kejadian x yang dinyatakan dengan ukuran *bit.* Sedangkan $p(x)$ adalah probabilitas dari kemunculan kejadian merepresntasikan himpunan informasi suatu kejadian. Semakin besar nilai $h(x)$, maka semakin besar pula informasi yang dapat dinyatakan sebagai berikut

$$H(p) = -\Sigma_{x \epsilon \epsilon} \, p(x) \, log \, p(x) \quad (3.11)$$

Dengan

$\alpha \, \epsilon \, A \, dan \, b \, \epsilon \, B$

$x = (ab)$

$\varepsilon = A \, x \, B$

p(x) yakni peluang kelas a terdapat pada dokumen b.

Adapun hasil yang diharapkan dari metode algoritma *Maximum Entropy* yakni mendapatkan nilai (p) yang paling maksimal. Nilai *entropy* yang maksimal akan terpenuhi pada distribusi seragam sehingga mengakibatkan $(x) = \frac{1}{|x|}$, dengan

nilai $|X|$ merupakan kardinilitas dari X. kardinilitas suatu himpunan merupakan nilai ukuran banyaknya elemen yang terdapat dalam suatu himpunan. Sehingga proses untuk mendapatkan nilai maksimal yang seragam dalam klasifikasi tidak semudah membagi nilai 1dengan nilai kardinalitas X. Selain itu pencarian distribusi probabilitas tentunya harus memenuhi batasan-batasan sesuai dengan data yang diteliti.

2.7.2 Prinsip Maximum Entropy

Pada metode *Maximum Entropy* dinyatakan bahwa untuk memperoleh nilai *entropy* maksimal maka seluruh distribusi akan diusahakan untuk *inform*, apabila tidak terdapat informasi data yang lengkap. Pada klasifikasi teks dengan *Maximum Entropy* dilakukan pengestimasian distribusi fitur kemunculan kata. Perhitungan fitur didasari oleh penggunaan $fi \epsilon \{0,1\}$ dalam pencarian informasi kemunculan suatu fitur dalam suatu dokumen. Sehingga dasarnya metode algoritma *Maximum Entropy* digunakan untuk mencari distribusi probabilitas yang paling seragam.

2.7.3 Algoritma Klasifikasi dengan *Maximum Entropy*

Adapun proses algoritma klasifikasi teks menggunakan metode *Maximum Entropy* sebagai berikut;

1. Mengidentifikasi kata-kata spesifik yang ada di dalam dokumen (kalimat)
2. Membentuk matriks yang berisi nilai kemunculan kata-kata spesifik tersebut dengan indeks berikut:

$$f_j(a,b)\begin{cases} 1 : jika\ f\ j\ muncul\ di\ dokumen\ b\ pada\ kelas\ a \\ 0 : jika\ f\ j\ tidak\ muncul\ di\ dokumen\ b\ pada\ kelas\ a \end{cases} \quad (3.12)$$

3. Membangun data latih untuk membuat algoritma *Maximum Entropy* dengan menghitung nilai untuk setiap kelas *aj*.

$$a_J^{(0)} = 1 \tag{3.13}$$

$$a_J^{(n+1)} = a_J^{(n)} \left(\frac{E_p f_j}{E^{(n)} f_j}\right)^{\frac{1}{c}} \tag{3.14}$$

Dimana

$$E_\beta f_j = \Sigma_{x \epsilon \epsilon} \; p(x)\, f_j(x) \tag{3.15}$$

$$E_\beta^{(n)}(x) = \Sigma_{x \epsilon \epsilon} \; p^{(n)}(x) f_j(x) \tag{3.16}$$

$$\forall\, x \in \Sigma_{j=1}^{k} f_j(x) = c \tag{3.17}$$

4. Menghitung *joint probability* $p(a, b)$ untuk perhitungan data uji

$$a = \{positif, negatif\} \tag{3.18}$$

$$p^*(a, b) = \pi \Pi_{j=1}^{k} a_j^{f_{j(ab)}} \tag{3.19}$$

5. Menetukan kelas dokumen data uji dengan melihat nilai a^* tertinggi dari masing-masing kelas.

$$a^* = argmax\; p(a, b) \tag{3.20}$$

$$dengan\; a \in (positif, negatif)$$

2.8 Wordcloud

Wordcloud merupakan salah satu metode untuk menampilkan kata-kata popular yang berkaitan dengan kata kunci internet dari data teks, khususnya pada analisis *text mining. Wordcloud* dapat digunakan dalam menyoroti *trend* ataupun

istilah popular dikalangan pengguna. *Wordcloud* dibentuk berdasarkan frekuensi kemunculan kata, dimana kata yang paling sering muncul di dalam teks akan memiliki ukuran paling besar, begitu pula sebaliknya. Pendekatan menggunakan *wordcloud* dapat memberikan penjelasan terhadap pertanyaan penelitian dengan sangat cepat dan mudah serta dapat pula dilakukan analisis yang koprehensif.

2.9 Asosiasi Teks

Asosiasi teks diperoleh dengan melakukan pendekatan pada perhitungan nilai korelasi. Pada umumnya, nilai korelasi digunakan dalam menyatakan hubungan dua atau lebih variabel kuantitatif, namun pada asosiasi teks nilai korelasi dimaknai sebagai keeratan hubungan antar dua atau lebih variabel kualitatif. Korelasi bertujuan untuk menemukan tingkat hubungan antara variabel (X) dan variabel (Y), dalam ketentuan data memiliki syarat-syarat tertentu.

$$r = \frac{1}{\sqrt{\{n\Sigma xi^2 (\Sigma xi)^2\}\{n\Sigma yi^2 (\Sigma yi)^2\}}} \quad (3.21)$$

dengan

r = nilai korelasi antara variabel x dan variabel y

n = banyaknya pasangan data x dan y

Σxi = jumlah nilai pada variabel x i = 1,2,3, …, n

Σyi = jumlah nilai variabel y

Σxi^2 = kuadrat dari total nilai variabel x

Σxy^2 = kuadrat dari total nilai variabel y

$\Sigma xi\ \Sigma yi$ = jumlah dari hasil perkalian antara nilai variabel x dan variabel y

Dalam perhitungan asosiasi teks, pertama data teks ditransformasikan ke dalam *document term matrix* (dtm). Adapun simulasi perhitungan dilakukan pada enam data berikut:

Kata 1

Kata 1 Kata 2

Kata 1 Kata 2 Kata 3

Kata 1 Kata 2 Kata 3 Kata 4

Kata 1 Kata 2 Kata 3 Kata 4 Kata 5

Kata 1 Kata 2 Kata 3 Kata 4 Kata 5 Kata 6

Kemudian ke-6 kata tersebut diubah dalam *document term matrix*

Doc	Kata 1	Kata 2	Kata 3	Kata 4	Kata 5	Kata 6
1	1	0	0	0	0	0
2	1	1	0	0	0	0
3	1	1	1	0	0	0
4	1	1	1	1	0	0
5	1	1	1	1	1	0
6	1	1	1	1	1	1

Setelah diperoleh nilai *document term matrix,* selanjutnya dilakukan perhitungan nilai asosiasi. Nilai asosiasi diperoleh dengan menghitung rumus korelasi seperti pada simulasi kata 2 dan kata 4 berikut:

Doc	Kata 2	Kata 4	Kata 2^2	Kata 4^2	Kata 2^4
1	0	0	0	0	0
2	1	0	1	0	0
3	1	0	1	1	1
4	1	1	1	1	1
5	1	1	1	1	1
6	1	1	1	1	1
Total	5	3	5	4	4

$$r = \frac{n\Sigma xiyi - (\Sigma xi)(\Sigma yi)}{\sqrt{\{n\Sigma xi^2 (\Sigma xi)^2\}\{n\Sigma yi^2 (\Sigma yi)^2\}}}$$

$$r = \frac{(6x3) - (5x3)}{\sqrt{\{(6x5) - 5^2\}\{(\Sigma xi)^2\}\{(6x3) - 3^2\}}}$$

$$r = \frac{3}{\sqrt{45}} = 0{,}0447$$

jadi, nilai korelasi kata 2 dan kata 4 sebesar 0,447. Hal ini menunjukkan bahwa besarnya asosiasi atau hubungan antara kata 2 dan kata 4 sebesar 0,447 atau 44,7%.