BAB II

TINJAUAN PUSTAKA

2.1 Transportasi *Online*

Perkembangan teknologi secara pesat dalam bidang telekomunikasi di seluruh dunia menyebabkan setiap negara harus mampu bersaing dengan pemanfaatan teknologi serta mengaplikasikannya di dalam beraktivitas. Dalam bidang transportasi, beralihnya jasa transportasi konvensional ke jasa transportasi *online* merupakan bagian dari kemajuan teknologi. Diciptakannya teknologi bertujuan untuk memudahkan berbagai aktifitas manusia sehari-hari. Dengan kehadiran dari transportasi *online* menjawab kebutuhan masyarakat berkaitan dengan angkutan umum. Transportasi *online* sekarang banyak diminati masyarakat karena beragam keunggulannya yang mencakup : asuransi, diskon, keamanan, kenyamanan, kepraktisan, keterpercayaan, lahan kerja baru atau kerja sampingan, dan promo (Anwar 2017).

2.2 *Twitter*

Twitter merupakan salah satu situs jejaring sosial yang masih popular hingga saat ini. *Twitter* didirikan oleh Jack Dorsey pada bulan Maret 2006. *Twitter* membatasi kata yang akan di post sebanyak 140 karakter. Namun, tidak hanya tulisan yang dapat diunggah pada jejaring sosial tersebut, *twitter* juga bisa mengunggah foto, video, url, dan lain-lain. *Twitter* banyak digunakan untuk

berbagai informasi, menjalin relasi bisnis, menuangkan isi hati dan pikiran dalam bentuk tulisan.

Pada aplikasi *twitter* disediakan sebuah *search engine* yang dapat digunakan oleh pengguna *twitter*. Dengan *search engine* tersebut kita bisa mendapatkan informasi terkait *tweets* yang ada pada aplikasi *twitter*. Informasi tersebut banyak digunakan untuk mencari wawasan terkait permasalahan yang sedang dihadapi. Informasi yang diberikan dapat berupa *tweets*, pengguna yang menuliskan *tweets* tersebut, lokasi *tweets* tersebut, dan lain-lain. (Adiyana dan Hakim 2015 dalam Ghifari 2018).

2.3 *Text Mining*

Text Mining merupakan suatu proses penggalian informasi berdasarkan suatu sumber data dokumen yang berupa teks dalam suatu proses yang dilakukan dengan komputer (Feldman dan Sanger 2007 dalam Fatimah 2018). *Text mining* juga dikenal sebagai *data mining text* atau penemuan pengetahuan dari *database* tekstual. Menurut buku *The Text Mining Handbook*, *text mining* didefinisikan sebagai suatu proses menggali informasi dimana seorang *user* berinteraksi dengan sekumpulan dokumen menggunakan *tools* analisis yang merupakan komponen *data mining.*

Perbedaan antara *text mining* dengan *data mining* terletak pada sumber data yang digunakan. Dalam *text mining* pola-pola yang diekstrak dari data tekstual yang tidak terstruktur bukan berasal dari suatu database. Sumber data yang digunakan dalam *text mining* adalah sekumpulan teks yang memiliki format yang

tidak terstruktur atau minimal *semi* terstruktur. Adapun tugas khusus dari *text mining* antara lain yaitu pengkategorian teks dan pengelompokkan teks. (Nurhuda et al. 2013). Sedangkan dalam *data mining* data yang diolah adalah data yang terstruktur dari proses *warehousing* sehingga lebih mudah diproses oleh mesin/komputer. Persamaan dari *text mining* dan *data mining* adalah data yang digunakan merupakan data besar dan data berdimensi tinggi dengan struktur yang terus berubah.

Adapun tahapan-tahapan dalam *text mining* secara umum adalah *text preprocessing* dan *feature selection* (Feldman dan Sanger 2007 dalam Fatimah 2018). Dimana penjelasan dari tahapan-tahapan diatas sebagai berikut:

2.3.1 Text Preprocessing

Tahap pertama dalam melakukan *text mining* yaitu *text preprocessing*, teks dokumen yang digunakan harus dipersiapkan terlebih dahulu, setelah itu dapat digunakan untuk proses utama. *Text preprocessing* 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:

a. *Spelling Normalization*

Spelling Normalization merupakan proses perbaikan atau subtitusi kata-kata yang salah eja atau disingkat dalam bentuk tertentu. Subtitusi 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.

b. *Case Folding*

Case Folding adalah proses penyamaan *case* dalam sebuah dokumen. Hal ini dilakukan untuk mempermudah pencarian. Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu peran *case holding* dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (dalam hal ini huruf kecil atau *lowercase*).

Tabel 2.1 Contoh tahapan dan luaran *Case Folding*

Tweet	Luaran
"Pertama kali coba pesan makanan di aplikasi @gojekindonesia, ternyata mudah bgt dan banyak promonya bgt/uhtg53/uhtg89/uhtg53/uhtg53	pertama kali coba pesan makanan di aplikasi gojekindonesia ternyata mudah bgt dan banyak promonya bgt/uhtg53/uhtg89/uhtg53/uhtg53

c. *Tokenizing*

Tokenizing adalah proses memecah kalimat menjadi kata-kata yang dilakukan untuk menjadikan sebuah kalimat menjadi lebih bermakna. Tahap pertama yang dilakukan adalah normalisasi kata dengan mengubah semua karakter huruf menjadi huruf kecil atau *to LowerCase.* Proses tokenisasi diawali dengan menghilangkan delimiter-delimiter yaitu *symbol* dan tanda baca yang ada pada

teks tersebut seperti @, $, &, tanda titik (.), koma (,), tanda Tanya (?), tanda seru (!). tahap tokenisasi selanjutnya yaitu proses penguraian teks yang semula berupa kalimat-kalimat yang berisi kata-kata. Proses pemotongan *string* berdasarkan tiap kata yang menyusunnya, umumnya setiap kata akan terpisahkan dengan karakter spasi, proses tokenisasi mengandalkan karakter spasi pada dokumen teks untuk melakukan pemisahan. Hasil dari proses ini adalah kumpulan kata saja (Putri, 2016).

Tabel 2.2 Contoh tahapan dan luaran *Tokenizing*

Tweet	Luaran
“pertama kali coba pesan makanan di aplikasi gojekindonesia, ternyata mudah bgt dan banyak promonya bgt/uhtg53/uhtg89/uhtg53/uhtg53”	Pertama ; kali ; coba ; pesan ; makanan ; di ; aplikasi ; gojekindonesia ; ternyata ; mudah ; bgt ; dan ; banyak ; promonya ; /uhtg53/uhtg89/uhtg53/uhtg53

d. *Filtering*

Tahap filtrasi adalah tahap mengambil kata-kata penting dari hasil token. Algoritma *stoplist* (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”, “di”, “dari”, dan seterusnya (Putri, 2016). Dalam filtrasi ini menggunakan

stoplist/stopword agar kata-kata yang kurang penting dan sering muncul dalam suatu dokumen dibuang sehingga hanya menyisakan kata-kata yang penting dan mempunyai arti yang diproses ke tahap selanjutnya.

Tabel 2.3 Contoh tahapan dan luaran *Fittering*

Tweet	Luaran
Pertama ; kali ; coba ; pesan ; makanan ; di ; gojekindonesia ; ternyata ; mudah ; bgt ; dan ; banyak ; promonya /uhtg53/uhtg89/uhtg53/uhtg53	Pertama ; kali ; coba ; pesan ; makanan ; di ; gojekindonesia ; ternyata ; mudah ; bgt ; dan ; banyak ; promonya

e. *Stemming*

Stemming bertujuan untuk mengurangi jumlah kata dan mendapatkan kata dasar yang benar-benar sesuai.

Tabel 2.4 Contoh tahapan dan luaran *Stemming*

Tweet	Luaran
Pertama ; kali ; coba ; pesan ; makanan ; di ; gojekindonesia ; ternyata ; mudah ; bgt ; dan ; banyak ; promonya	Pertama ; kali ; coba ; pesan ; makanan ; di ; gojekindonesia ; ternyata ; mudah ; bgt ; dan ; banyak ; promonya

2.3.2 *Feature Selection*

Feature Selection merupakan tahap lanjutan dari pengurangan dimensi. Walaupun di tahap sebelumnya sudah melakukan penghapusan kata-kata yang tidak deskriptif (*stopword*), tidak semua kata-kata didalam dokumen memiliki arti penting. Sehingga untuk mengurangi dimensi, pemilihan hanya dilakukan pada kata-kata yang relevan dan yang benar-benar mempresentasikan isi dari suatu dokumen. Kata-kata yang dinilai penting dilihat dari intensitas kemunculan dan yang paling informatif dari keseluruhan.

a. Pembobotan Kata (*Term Weighting*)

Hal yang perlu diperhatikan dalam pencarian informasi dari koleksi dokumen yang heterogen adalah pembobotan *term*. *Term* dapat berupa kata, *frase* atau unit hasil *indexing* lainnya dalam suatu dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut. Karena setiap kata memiliki tingkatan kepentingan yang berbeda dalam dokumen , maka untuk setiap kata tersebut diberikan sebuah indikator, yaitu *term weight* (Zafikri, 2008). Zafikri (2008) menyatakan *term weighting* atau pembobotan *term* sangat dipengaruhi oleh hal-hal sebagai berikut:

1. *Term Frequency* (TF)

Term Frequency (TF) adalah faktor yang menentukan bobot *term* pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen tersebut. Nilai jumlah kemunculan suatu kata (*term frequency*) diperhitungkan dalam pemberian bobot terhadap suatu kata. Semakin besar jumlah kemunculan suatu term (tf

tinggi) dalam dokumen, semakin besar pula bobotnya dalam dokumen atau akan memberikan nilai kesesuaian yang semakin besar.

2. *Inverse Document Frequency* (IDF)

Inverse Document Frequency (IDF) adalah pengurangan dominansi *term* yang sering muncul di berbagai dokumen. Hal ini diperlukan karena *term- term* yang banyak muncul di berbagai dokumen, dapat dianggap sebagai *term* umum (*common term*) sehingga tidak penting nilainya. Sebaliknya faktor jarang munculnya kata (*term scarcity*) dalam koleksi dokumen harus diperhatikan dalam pemberian bobot. Kata yang muncul pada sedikit dokumen harus dilihat sebagai kata yang lebih penting (*uncommon terms*) daripada kata yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang mengandung suatu kata (*Inverse Document Frequency*) (Putranti dan Winarko, 2014).

Metode TF-IDF merupakan metode pembobotan *term* yang banyak digunakan sebagai metode pembanding terhadap metode pembobotan baru. Pada metode ini, perhitungan bobot *term t* dalam sebuah dokumen dilakukan dengan mengalikan nilai *Term Frequency* dengan *Inverse Document Frequency*. Pada *Term Frequency* (TF), terdapat berbagai jenis formula yang dapat digunakan yaitu (Zafikri, 2008) :

1. TF biner *(binary TF*), hanya memperhatikan apakah suatu kata ada atau tidak dalam dokumen, jika ada diberi nilai satu, jika tidak diberi nilai nol.

2. TF murni (*raw TF*), nilai TF diberikan berdasarkan jumlah kemunculan suatu kata di dokumen. Contohnya, jika muncul lima kali maka kata tersebut akan bernilai lima.
3. TF logaritmik, hal ini untuk menghindari dominansi dokumen yang mengandung sedikit kata dalam *query*, namun mempunyai frekuensi yang tinggi.

$$1 + \log(tf) \quad (2.1)$$

4. TF normalisasi, menggunakan perbandingan antara frekuensi sebuah kata dengan jumlah keseluruhan kata pada dokumen. *Inverse Document Frequency* (IDF) dihitung dengan menggunakan formula:

$$idf_j = \log \frac{D}{df_j} \quad (2.2)$$

Dimana

idf_j : adalah jumlah semua dokumen dalam koleksi

df_j : adalah jumlah dokumen yang mengandung term tj

Dengan demikian rumus umum untuk TF-IDF adalah penggabungan dari formula perhitungan *raw* TF dan formula IDF dengan cara mengalikan nilai *Term Frequency* (TF) dengan nilai *Inverse Document Frequency* (IDF):

$$w_{ij}=tf_{ij}\times idf_j \quad (2.3)$$

Keterangan :

tf_{ij} : adalah bobot *term* t_j terhadap dokumen d_i

idf_j : adalah jumlah kemunculan *term* t_j dalam dokumen d_i

2.4 Klasifikasi

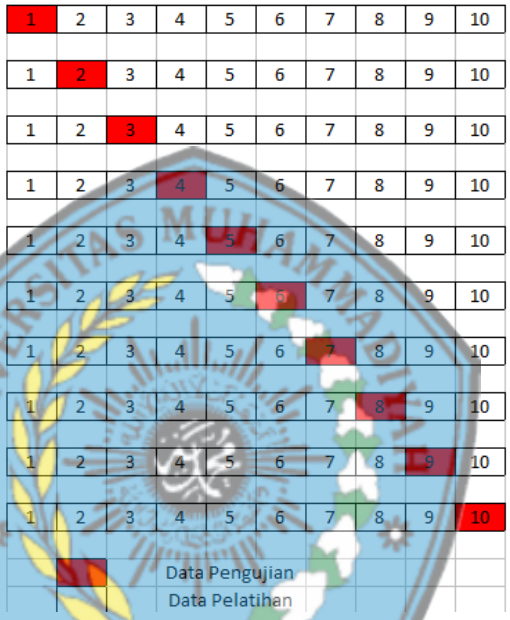
Teknik klasifikasi adalah salah satu dari teknik data mining yang termasuk ke dalam *supervised learning*. *Supervised learning* adalah proses pembentukan sebuah korespondensi menggunakan sebuah *training* dataset. Tujuannya adalah untuk memprediksi target dari beberapa atribut (Zaki dan Meira, 2014). Terdapat pada dua pekerjaan utama pada klasifikasi yaitu melakukan *training* untuk disimpan sebagai prediksi dan melakukan *testing* untuk proses klasifikasi agar diketahui di label mana objek data tersebut (Liu, Loh, dan Sun, 2009).

Model klasifikasi dapat dibangun berdasarkan pengetahuan seorang pakar (ahli). Namun, mengingat himpunan data yang sangat besar, model klasifikasi lebih sering dibangun menggunakan teknik pembelajaran dalam bidang *machine learning*. Proses pembelajaran secara otomatis terhadap suatu himpunan data mampu menghasilkan model klasifikasi (fungsi target) yang memetakan objek data (*input*) ke salah satu kelas y yang telah didefinisikan sebelumnya. Jadi, proses pembelajaran memerlukan masukan (*input*) berupa himpunan data latih (*training set*) yang berlabel (memiliki atribut kelas) dan mengeluarkan *output* yang berupa sebuah model klasifikasi (Suyanto, 2017 dalam Praptiwi, D, Y. 2018)

2.5 *K-Fold Cross Validation*

K-Fold Cross Validation adalah salah satu metode yang digunakan untuk mempartisipasi data menjadi data *training* dan data *testing*. Pada penelitian ini digunakan metode tersebut karena dapat mengurangi bias yang terjadi dalam pengambilan sampel. Metode ini digunakan secara berulang-ulang membagi data

menjadi dua yaitu data *training* dan *testing*, setiap data memperoleh kesempatan menjadi data *testing* (Ibrahim, N. dan Wibowo, A. 2014). K disini merupakan besar angka partisi data yang akan digunakan untuk pembagian antara *training* dan *testing*. Berikut ilustrasi pembagian data menggunakan *k-fold cross validation.*



Gambar 2.1 Ilustrasi Pembagian Data

2.6 Analisis Sentimen

Analisis sentimen atau biasa disebut *opinion mining* merupakan studi komputasi pendapat, sentimen, dan ekspresi emosi yang diungkapkan dalam teks. Secara umum, *opinion mining* diperlukan untuk mengetahui sikap seorang pembicara atau penulis sehubungan dengan topik atau polaritas kontekstual keseluruhan dokumen. Sikap yang diambil mungkin menjadi pendapat atau penilaian atau evaluasi (teori appraisal), keadaan afektif (keadaan emosional

penulis saat menulis) atau komunikasi emosional (efek emosional penulis yang ingin disampaikan pada pembaca) (Saraswati, 2011).

Analisis sentimen bertujuan untuk mengelompokkan teks yang mengandung opini sebagai kelas positif dan negatif. (Liu 2010 dalam Saputra 2018). Analisis sentimen media sosial dapat menjadi sumber informasi dan dapat memberi wawasan yang berguna seperti dalam menentukan strategi pemasaran, meningkatkan penjualan produk, meningkatkan layanan pelanggan dan lain-lainnya.

2.7 *Naïve Bayes Classifier*

Naïve Bayes Classifier merupakan salah satu metode yang popular digunakan pada *data mining* karena kemudahan penggunaannya (Hall, 2006 dalam Syakuro 2017) serta waktu pemrosesannya yang cepat, mudah diimplementasikan dengan struktur yang cukup sederhana dan tingkat efektifitas yang tinggi (Taheri dan Mammadov 2013 dalam Syakuro 2017). Metode ini menggunakan aturan *Bayesian. Bayesian Classification* merupakan klasifikasi secara statistik dengan memprediksi suatu data terprediksi ke dalam kelas tertentu (Han et al. 2012 dalam Putra 2017).

Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas *Bayes*, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa

sebelumnya sehingga dikenal sebagai teori *Bayes*. Secara umum teorema *Bayes* dapat dinotasikan sebagai berikut:

$$P(C|E) = \frac{P(E|C)P(C)}{P(E)} \quad (2.4)$$

Keterangan:

$P(C|E)$: Probabilitas akhir bersyarat (*conditional probability*) suatu hipotesis C terjadi jika diberikan bukti (*evidence*) E terjadi.

$P(E|C)$: Probabilitas sebuah bukti E terjadi akan mempengaruhi hipotesis C.

P(C) : Probabilitas awal (priori) hipotesis C terjadi tanpa memandang bukti apapun.

P(E) : Probabilitas awal (priori) hipotesis E terjadi tanpa memandang hipotesis bukti yang lain.

Metode *Naïve Bayes Classifier* merupakan penyederhanaan dari algoritma *Naïve Bayes* dan cocok digunakan dalam pengklasifikasian teks atau dokumen.

Persamaannya adalah:

$$V_{MAP} = avgmax P(V_j|a_1, a_2, \dots, a_n) \quad (2.5)$$

Berdasarkan persamaan tersebut, maka rumus *bayes* dapat di tulis menjadi:

$$V_{MAP} = \frac{\underset{v_j \epsilon V}{avgmax} P(a_1,a_2,\dots,a_n|v_j)P(v_j)}{P(a_1,a_2,\dots,a_n)} \quad (2.6)$$

Karena kategori $P(a_1, a_2, \dots, a_n)$ merupakan bilangan konstanta, maka persamaannya dapat ditulis sebagai berikut:

$$V_{MAP} = \underset{v_j \in V}{avgmax} P(a_1, a_2, \dots, a_n | v_j) P(v_j) \quad (2.7)$$

Dari persamaan diatas dapat disederhanakan menjadi:

$$V_{MAP} = \underset{v_j \in V}{avgmax} \prod P(a_n | v_j) P(v_j) \quad (2.8)$$

Dimana untuk:

V_j = Kategori opini dari j = 1, 2, 3, …, n. Dimana dalam penelitian ini j_1 adalah kategori opini puas, j_2 adalah kategori opini tidak puas, j_3 adalah kategori opini netral.

$P(a_n|v_j)$ = Probabilitas a_n pada kategori v_j.

$P(v_j)$ = Probabilitas dari v_j.

Untuk menghitung $P(V_j)$ dan $P(a_n|v_j)$ pada saat pelatihan dimana persamaannya sebagai berikut:

$$P(V_j) = \frac{|docs_j|}{|contoh|} \quad (2.9)$$

$$P(a_n|v_j) = \frac{|n_k+1|}{n+|konstanta|} \quad (2.10)$$

Dimana untuk:

$|docs|$ adalah frekuensi dokumen setiap kategori.

$|contoh|$ adalah jumlah dokumen yang ada.

n_k adalah jumlah frekuensi kemunculan setiap kata.

n adalah jumlah frekuensi kemunculan dari setiap kategori.

$|kosakata|$ adalah jumlah semua kata dari semua kategori.

2.8 Ukuran Evaluasi Model Klasifikasi

Evaluasi pada suatu klasifikasi pada umumnya dilakukan dengan menggunakan sebuah himpunan data yang diuji, tidak digunakan dalam pelatihan klasifikasi tersebut. Pada tahap ini terdapat sejumlah ukuran yang dapat digunakan untuk menilai kembali atau mengevaluasi model klasifikasi, yaitu *accuracy* atau tingkat pengenalan, tingkat kesalahan atau kekeliruan klasifikasi, *recall* atau *sensitivity* atau *true positif, specificity* atau *true negative* dan *precision.*

Model klasifikasi yang telah dibuat yaitu pemetaan dari suatu baris data dengan keluaran sebuah hasil prediksi kelas atau target dari data tersebut. Pada klasifikasi ini terdapat dua kelas sebagai luarannya yang disebut klasifikasi biner. Kedua kelas tersebut biasa diinterpretasikan dalam {0,1}, {+1,-1} atau {positif,negatif}.

Pada proses evaluasi klasifikasi terdapat empat kemungkinan yang terjadi yaitu proses pengklasifikasian pada suatu baris data. Jadi, jika data positif dan diprediksi positif maka akan dihitung sebagai *true positif*, bahkan jika data itu diprediksi negatif maka akan dihitung sebagai *false negative.* Jika data negatif dan diprediksi negative maka akan dihitung sebagai *true negative*, tetapi jika data tersebut diprediksi positif maka akan dihitung sebagai *false positif.* Hasil

klasifikasi biner pada suatu dataset yang dipresentasikan dalam bentuk matriks 2×2 yaitu dinamakan *confusion matrix.* Berikut merupakan contoh dari matriks

Tabel 2.5 Matriks *Contengency* Prediksi dan Aktual

		Aktual	
	Class	**Positive**	**Negative**
Prediksi	*Positive*	*True Positive (TP)*	*False Negative (FN)*
	Negative	*False Positive (FP)*	*True Negative (TN)*

Confussion Matrix bermanfaat untuk menganalisis kualitas *classifier* dalam mengenali tuple-tuple dari kelas yang ada. TP dan TN menyatakan pada *classifier* mengenali tuple dengan benar , artinya tuple positif dikenali sebagai positif dan tuple negatif dikenali sebagai negatif. Sedangkan, FP dan FN menyatakan bahwa *classifier* salah dalam mengenali tuple, tuple negatif dikenali sebagai positif dan tuple negatif dikenali sebagai positif. Ada beberapa dalam formula perhitungan performa klasifikasi yaitu nilai akurasi, presisi, dan *recall* biasa ditampilkan dalam presentase.

a. *Accurasy*

Akurasi adalah nilai ketepatan dimana pengguna memprediksi suatu kata sesuai dengan jawaban suatu sistem. Berikut perhitungan nilai akurasi

$$Accurasy = \frac{TP+FN}{TP+FP+TN+FN}$$

b. *Precission*

Precission adalah proporsi jumlah dokumen teks yang relevan yang dikenali diantara semua dokumen teks yang dipilih oleh system atau kesesuaian terhadap system.

$$Precission = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

c. *Recall*

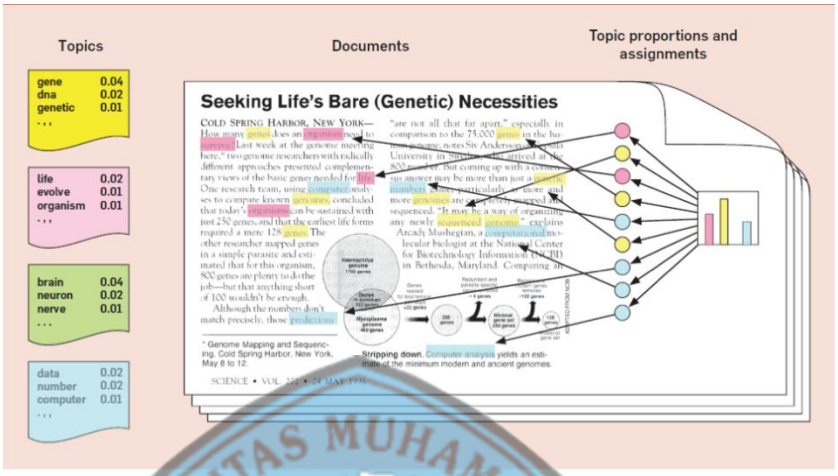
Recall adalah proporsi jumlah dokumen teks yang relevan dikenal diantara semua dokumen teks relevan yang ada pada koleksi.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

2.9 *Topic Modeling*

Topic Modelling atau pemodelan topik merupakan kumpulan algoritma yang digunakan untuk menemukan struktur tersembunyi dari tema yang terdapat dalam setiap dokumen. Algoritma ini, dapat digunakan untuk pengembangan dalam pencarian, ataupun meringkas teks yang ada dalam dokumen (Blei 2003 dalam Agustina 2017). *Topic Modelling* dapat mengatur kumpulan kata berdasarkan tema yang ditemukan. Selain itu, *topic modelling* juga dapat diaplikasikan untuk berbagai jenis data, seperti saat ini dilakukan adalah untuk mencari pola pada data genetik, gambar, ataupun pada sosial media. *Topic Modelling* merupakan salah satu bentuk dari *text mining* yang merupakan sebuah metode untuk menemukan

dan melacak kelompok kata dalam dokumen (Brett 2012 dalam Agustina 2017). Konsep *topic modelling* menurut Blei, ditunjukkan pada gambar

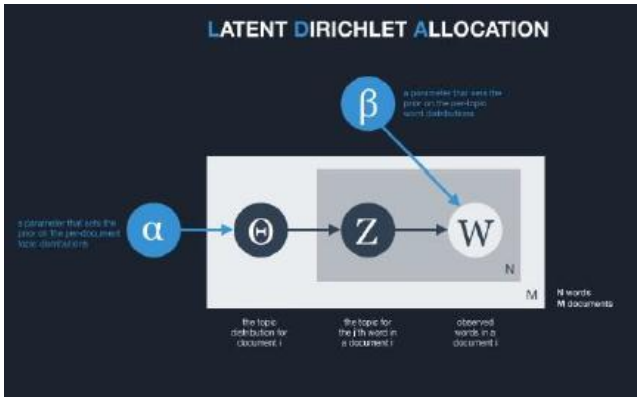


Gambar 2.2 Konsep *Topic Modelling*

Menurut Blei, dokumen mempunyai proporsi tersendiri dari topik-topik yang dibahas pada sebuah dokumen.

2.10 *Latend Dirichlet Allocation (LDA)*

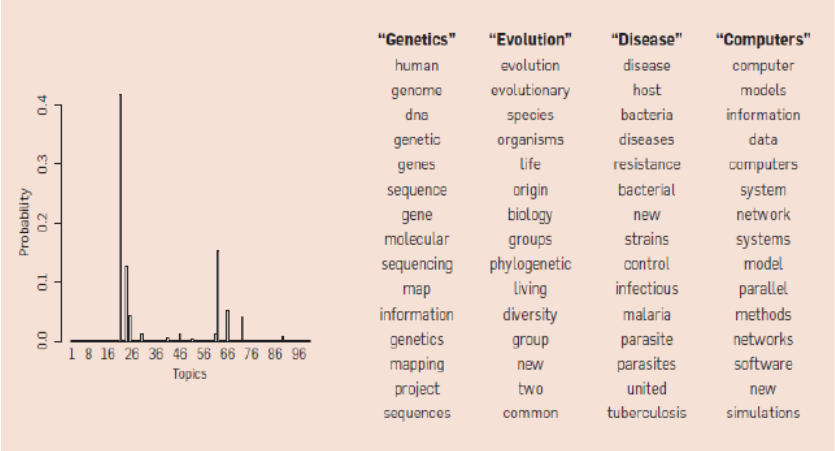
Latend Dirichlet Allocation (LDA) adalah metode statistika yang digunakan sebagai model untuk menganalisis suatu dokumen. LDA berusaha untuk melihat dokumen dengan cara mundur untuk menemukan satu set topik yang mungkin telah dikoleksi. LDA mempresentasikan dokumen dengan berbagai topik yang dibuat berdasarkan probabilitas tertentu (Chen 2011 dalam Agustina 2017). Probabilitas topik, merepresentasikan kejelasan dari suatu dokumen. Menurut Blei bahwa dalam sebuah dokumen terdapat beberapa komponen, yang ditunjukkan pada gambar 2.2



Gambar 2.3 Visualisasi *Topic Modeling dengan* Metode LDA

Alpha (α) menggambarkan sebuah parameter yang digunakan untuk menghitung bagaimana distribusi topik dalam dokumen. Semakin besar nilai α yang dimiliki pada suatu dokumen, menandakan semakin banyak campuran topik yang dibahas dalam dokumen. Semakin rendah nilai α yang dimiliki menunjukkan dokumen hanya membahas sedikit topik tertentu. Z merepresentasikan topik dari kata tertentu pada sebuah dokumen. Sedangkan Beta (β) adalah parameter yang digunakan untuk menghitung distribusi kata dalam topik. Semakin tinggi nilai β, maka semakin banyak kata-kata yang ada di dalam topik. Semakin kecil nilai β, maka semakin sedikit kata-kata yang ada pada topik sehingga lebih spesifik.

Ide dasar dari LDA adalah bahwa dalam dokumen, merepresentasikan campuran topik secara acak, dimana setiap topik digolongkan berdasarkan distribusi antar kata. LDA tidak hanya digunakan untuk melakukan pendeteksian topik saja, namun LDA juga digunakan sebagai salah satu *tools* untuk berbagai penelitian seperti pada konten percakapan. Sebagai salah satu contoh dari Blei, distribusi topik yang ditampilkan dengan kumpulan kata-kata pada dokumen ditunjukkan pada gambar 2.4



Gambar 2.4 Distribusi Topik LDA