

BAB 2 LANDASAN TEORI

Pada bab ini dijelaskan landasan teori dari pekerjaan dan metode yang digunakan dalam tugas akhir untuk melakukan klasifikasi topik. Pembahasan ini dimulai dengan penjelasan klasifikasi topik, lalu dilanjutkan mengenai metode-metode pada *machine learning* yang akan digunakan pada tugas akhir ini.

2.1 Klasifikasi Topik

Suatu dokumen dapat dikelompokkan ke dalam satu topik tertentu. Proses pengelompokkan dokumen ini dapat dilakukan dengan cara membaca beberapa kalimat atau kata yang ada pada dokumen tersebut. Beberapa kata atau kalimat khusus yang ada pada dokumen tersebut akan membuat seseorang dapat menentukan topik dari dokumen tersebut. Contoh beberapa kalimat yang dapat membuat seseorang secara cepat mengetahui topik dari dokumen :

1. Setelah kalah pada game pertama, Nova/Liliyana bahkan hampir tersingkir ketika tertinggal hingga empat angka menjelang akhir game kedua, 13-17.
2. Hepatitis C juga dapat ditularkan melalui hubungan suami-istri, tetapi risiko penularan tidaklah sebesar HIV.
3. Disebutkan, total nilai perdagangan bilateral kedua negara pada 2007 mencapai 1,8 miliar dolar AS dengan nilai ekspor Indonesia ke Turki mencapai 1,1 juta dollar AS, dan impor sebesar 700 juta dollar AS.

Kalimat pertama mengandung beberapa kata seperti kalah, game, dan 13-17. Dengan melihat kalimat (1) ini, dapat diketahui atau disimpulkan bahwa kalimat (1) ini sedang membahas topik olahraga. Kalimat kedua terdapat kata hepatitis c, penularan, dan HIV. Dengan kalimat (2) menggunakan beberapa kata tersebut, akan timbul suatu dugaan bahwa kalimat (2) ini sedang membahas tentang kesehatan. Pada kalimat (3), terdapat kata-kata khusus seperti perdagangan, ekspor, dan impor. Dari kata-kata yang terkandung pada kalimat tersebut dapat disimpulkan bahwa kalimat tersebut sedang membicarakan mengenai ekonomi ataupun perdagangan.

Apabila suatu kata seperti kalah dipakai pada kalimat yang lain, ada kemungkinan bahwa kalimat tersebut tidak akan termasuk pada topik olahraga. Akan tetapi, karena kata tersebut digunakan bersamaan dengan kata game dan 13-17, maka kalimat itu dapat disimpulkan termasuk ke dalam topik olahraga. Oleh karena itu, dalam menentukan suatu dokumen ataupun kalimat termasuk dalam topik yang mana, haruslah dilihat kata-kata yang terkait pada dokumen ataupun kalimat tersebut.

Seorang manusia dapat menyimpulkan suatu dokumen termasuk ke dalam topik yang mana. Hal ini dikarenakan manusia telah mengetahui dan mempelajari terlebih dahulu kata-kata yang digunakan pada suatu dokumen. Atau pun jika ia tidak mengetahui beberapa kata dari dokumen tersebut, maka dia akan menyimpulkan dokumen tersebut termasuk ke dalam topik yang mana berdasarkan kata-kata yang ia tahu. Manusia ini sebelumnya sudah mengerti bahwa kata-kata seperti kalah, game pertama, 13-17, dan kalah biasanya digunakan pada olahraga.

Kegunaan dari klasifikasi topik dapat dirasakan manusia secara dekat dalam kehidupan sehari-hari. Kegunaan yang paling terasa adalah pada saat seseorang mencari suatu dokumen. Dokumen tersebut akan dapat ditemukan lebih mudah apabila sudah dilakukan proses klasifikasi dokumen yang dimiliki terlebih dahulu dibandingkan mencarinya dengan melihat satu per satu dokumen yang dimiliki. Selain itu, kegunaan yang lain mungkin tidak menjadi peran utama tapi lebih ke mendukung kegunaan yang lain. Sebagai contoh, mesin penerjemah. Mesin ini akan lebih mudah menerjemahkan suatu dokumen apabila sebelumnya telah diketahui termasuk ke dalam bahasa manakah dokumen tersebut berada.

Penggolongan dokumen ke dalam beberapa topik dapat dilakukan dengan dua cara yaitu dengan menggunakan klasifikasi dan dengan *clustering*. Klasifikasi dokumen dilakukan dengan cara pembelajaran terlebih dahulu. Mesin ataupun robot yang akan kita gunakan terlebih dahulu akan melalui masa *training* yaitu masa pembelajaran atau pengenalan topik yang akan dipilih (*supervised learning*). Setelah mesin itu belajar untuk mengenal ciri-ciri dari setiap topik yang ada,

mesin ataupun robot tersebut siap digunakan. Mesin tersebut akan memasukkan dokumen yang ada berdasarkan hasil dari pembelajarannya.

Cara lain yang dapat digunakan adalah *clustering*. *Clustering* merupakan cara pengelompokan dokumen dengan pendekatan *unsupervised learning*, yaitu semua dokumen akan dilihat dari kemiripan informasi yang terkandung. Cara ini akan mengelompokkan satu dokumen dengan dokumen lain yang dianggap paling mirip. Dengan cara inilah akan terbentuk kelompok-kelompok dokumen yang sejenis.

2.2 Machine learning untuk klasifikasi topik

Dalam bidang klasifikasi topik, *machine learning* dapat dilakukan dengan dua cara yaitu dengan menggunakan pendekatan *supervised learning* dan *unsupervised learning*. Pada *unsupervised learning*, pengelompokan kelompok tidak melalui proses pengenalan ciri-ciri suatu topik dokumen (Turney, 2002). Pendekatan ini akan dilakukan dengan cara menentukan dokumen mana yang akan menjadi pusat dari satu bidang topik dan selanjutnya dokumen lain akan dilihat kedekatannya dengan dokumen tersebut. Dokumen yang akan ditentukan topiknya dimasukkan ke dalam kelompok topik yang paling dekat dengan dokumen yang sudah dipilih sebagai pusat. Hal ini dapat dilakukan berulang kali hingga dokumen terbagi ke dalam beberapa kelompok dan pembagian dokumen tersebut sudah tidak berubah ataupun perubahan dari pembagian kelompok tersebut tidak signifikan. Mengenai hal signifikan atau tidaknya suatu perubahan tergantung pada batas yang telah ditetapkan sebelumnya oleh orang yang menggunakannya. Pendekatan ini biasanya dinamakan dengan teknik *clustering*.

Pendekatan lainnya dilakukan dengan cara mempelajari terlebih dahulu ciri-ciri yang dapat dimiliki oleh dokumen yang ada pada suatu bidang topik. Pendekatan ini biasanya dinamakan dengan teknik klasifikasi. Klasifikasi topik ini akan membagi dokumen yang ada menjadi dokumen *training* dan dokumen *testing*. Dengan menggunakan dokumen *training* ini, pendekatan ini akan membangun model analisis untuk dapat menentukan dokumen *testing* tersebut dimasukkan ke dalam topik yang mana. Pembangunan model tersebut dilakukan dengan terlebih dahulu melakukan *feature selection*, yaitu pemilihan kata atau bagian dari

dokumen yang nantinya akan digunakan sebagai tanda untuk menentukan topik dari suatu dokumen. Salah satu fitur yang dapat digunakan adalah *n-gram language model*. Model ini akan memakai fitur n-kata yang berurutan muncul pada suatu dokumen. Fitur ini akan dihitung untuk tiap dokumen pada data *training* dan kemudian akan dianalisis dengan menggunakan metode *machine learning*. Analisis klasifikasi topik dengan *machine learning* ini dapat menggunakan beberapa metode yang ada, seperti Naïve Bayes dan Maximum Entropy. Selanjutnya, analisis dokumen pada data *testing* juga menggunakan metode *machine learning* dengan fitur yang sama yang telah digunakan pada data *training*.

Metode *machine learning* dapat digunakan dalam melakukan klasifikasi topik. Hal ini telah terbukti pada penelitian yang telah dilakukan sebelumnya. Pada (Sebastiani,2002), salah satu metode yang digunakan untuk melakukan klasifikasi dokumen adalah Naïve Bayes. Metode ini mencapai nilai akurasi tertinggi yaitu 81,5% saat melakukan klasifikasi ke dalam 10 topik. Penelitian lain dilakukan pada (Nigam, Lavery,& McCallum, 1999), dua metode yang digunakan adalah Naïve Bayes dan Maximum Entropy. Dengan menggunakan metode Naïve Bayes, hasil akurasi tertinggi yang dihasilkan adalah 86,9%. Sementara dengan menggunakan Maximum Entropy nilai akurasi yang dihasilkan dapat mencapai 92.18%. Dengan melihat kedua percobaan yang pernah dilakukan, kedua metode ini telah terbukti dapat digunakan dalam melakukan klasifikasi topik.

2.3 Naïve Bayes

Naïve Bayes merupakan salah satu metode *machine learning* yang menggunakan konsep probabilitas. Metode ini melakukan klasifikasi dengan menghitung nilai probabilitas $p(a|b)$, yaitu probabilitas kelas a jika diketahui suatu b , berdasarkan teorema Bayes. Klasifikasi dapat dilakukan untuk menentukan suatu kelas $a \in A$ dari suatu dokumen $b \in B$ dengan $A = \{a_1, a_2, \dots, a_p\}$ dan $B = \{b_1, b_2, \dots, b_q\}$. Penentuan kelas dalam klasifikasi dokumen tersebut dilakukan dengan cara memilih nilai maksimum dari $p(a|b)$ dari distribusi probabilitas $P = \{p(a|b) \mid a \in A \text{ dan } b \in B\}$. Suatu dokumen b ke i dapat direpresentasikan sebagai vektor dari nilai-nilai fitur yang ada pada dokumen tersebut sehingga $b = [f_{i1}, f_{i2}, \dots, f_{im}]$.

Nilai dari elemen tiap vektor merupakan nilai untuk fitur f_j pada himpunan fitur $F = \{f_1, f_2, \dots, f_n\}$ dengan f_{ij} adalah nilai dari fitur ke j pada dokumen b ke i . Berdasarkan teorema Bayes penghitungan nilai probabilitas $p(a|b)$ dapat dilakukan dengan menggunakan persamaan (Mitchell, 2005):

$$p(a|b) = \frac{p(b|a) \times p(a)}{p(b)}$$

Dengan $p(b|a)$ merupakan nilai probabilitas dari kemunculan dokumen b pada kelas a , $p(a)$ merupakan nilai probabilitas kemunculan kelas a dan $p(b)$ merupakan nilai probabilitas kemunculan dokumen b .

2.3.1 Model Naïve Bayes

Teorema Bayes menganggap suatu dokumen sebagai kumpulan fitur yang membentuknya. Dengan menggunakan pemikiran seperti ini, proses klasifikasi akan membutuhkan data *training* yang cukup banyak. Hal ini diperlukan untuk dapat melakukan estimasi semua kemungkinan kombinasi fitur yang dapat merepresentasikan dokumen tersebut. Apabila nilai fitur yang digunakan adalah bilangan biner, yaitu bernilai 0 jika fitur tersebut tidak muncul pada dokumen dan bernilai 1 jika fitur tersebut ada, maka akan terdapat $2^{|F|} \times |A|$ kemungkinan dari b yang perlu untuk diestimasi. $|F|$ adalah jumlah *token* yang digunakan dalam melakukan klasifikasi dan $|A|$ adalah jumlah kelas yang akan dilakukan klasifikasi. Oleh karena itu, apabila terdapat 100 fitur dan 5 kelas yang akan digunakan, maka estimasi perlu dilakukan pada $2^{100} \times 5$ kemungkinan.

Berdasarkan teorema Bayes, Naïve Bayes ini merupakan metode yang dikembangkan sebagai pendekatan dalam melakukan estimasi kelas untuk suatu dokumen. Apabila pada teorema bayes fitur-fitur yang ada pada dokumen itu terkait satu dengan yang lain, maka lain halnya dengan Naïve Bayes ini. Naïve Bayes memiliki asumsi bahwa setiap fitur yang ada pada dokumen tersebut tidak berkaitan satu dengan yang lain (*conditional independence*), walau mungkin dalam kenyataannya ada kemungkinan bahwa fitur-fitur ini saling berkaitan. Berikut ini merupakan gambaran dari model yang digunakan Naïve Bayes untuk melakukan klasifikasi:

$$p(a|b) = \frac{\pi_j p(f_j|a) \times p(a)}{p(b)}$$

Probabilitas $p(b|a)$ digantikan dengan perkalian probabilitas $p(f_j|a)$ dari $|F|$ buah fitur independen yang merepresentasikan b . Apabila menggunakan nilai fitur biner, maka estimasi yang perlu diperhitungkan adalah $2 \times |F| \times |A|$ kemungkinan.

Proses pembelajaran untuk menentukan topik didapatkan dari hasil pemodelan permasalahan, yaitu dengan menghitung nilai $p(f_j|a)$ yang didapat dari data *training*. Untuk f_{ij} diskrit dengan $f_{ij} \in V = \{v_1, v_2, \dots, v_m\}$ maka nilai dari $p(f_j|a)$ akan dicari untuk seluruh kemungkinan nilai f_{ij} dan didapatkan dengan menghitung persamaan (Mitchell, 2005)

$$p(f_j = f_{ij}|a) = \frac{D_b(f_j = f_{ij}, a)}{D_b(a)}$$

dan

$$p(a) = \frac{D_b(a)}{|D|}$$

Dengan $D_b(f_j = f_{ij}, a)$ adalah fungsi yang mengembalikan jumlah dokumen b di kelas a yang memiliki nilai fitur $f_j = f_{ij}$, $D_b(a)$ adalah fungsi yang mengembalikan jumlah dokumen b dengan kelas a , dan $|D|$ merupakan jumlah seluruh dokumen pada data *training*. Persamaan $p(f_j = f_{ij}|a)$ seringkali digabungkan dengan proses *smoothing* untuk menghindari hasil berupa nilai 0, yang dapat mengacaukan hasil perhitungan proses klasifikasi topik. Berikut merupakan persamaan $p(f_j = f_{ij}|a)$ yang telah digabungkan dengan proses *Laplacian smoothing* (Mitchell, 2005):

$$p(f_j = f_{ij}|a) = \frac{D_b(f_j = f_{ij}, a) + 1}{D_b(a) + |V|}$$

dengan $|V|$ merupakan jumlah kemungkinan nilai dari f_{ij} .

Proses klasifikasi dokumen dilakukan dengan memilih nilai a yang akan memberikan nilai $p(a|b)$ paling besar, dan hal ini dapat dinyatakan sebagai berikut:

$$a^* = \arg \max_{a \in A} p(a|b) = \arg \max_{a \in A} \prod_j p(f_j|b) \times p(a)$$

kelas a^* merupakan kelas yang memiliki nilai $p(a|b)$ terbesar. Nilai $p(b)$ dapat diabaikan karena nilai $p(b)$ akan bernilai sama untuk semua kelas sehingga tidak akan memberikan pengaruh apapun dalam proses perbandingan nilai $p(a|b)$.

2.3.2 Naïve Bayes Multinomial

Metode Naïve Bayes Multinomial merupakan variasi lain dari naïve bayes. Metode ini memandang frekuensi kemunculan fitur ke- j pada dokumen b_i (f_{ij}) yang ingin diklasifikasi sebagai jumlah kemunculan *event* ataupun fitur independen. Kemunculan fitur f_{ij} pada dokumen b akan dianggap tidak berhubungan ataupun independen terhadap kemunculan fitur yang lain.

Berikut merupakan persamaan untuk menghitung nilai $p(f_j|a)$ dengan menggunakan metode Naïve Bayes Multinomial dan digabungkan dengan *Laplacian smoothing* (Schneider, 2004):

$$p(f_j|a) = \frac{\sum_{i=1}^{|D|} f_{ij} \delta(a^i = a) + 1}{\sum_{j=1}^{|F|} \sum_{i=1}^{|D|} f_{ij} \delta(a^i = a) + |F|}$$

, dengan $|F|$ adalah jumlah *token* yang digunakan.

Metode Naïve Bayes Multinomial melakukan proses klasifikasi dengan memasukkan informasi f_{ij} dari dokumen b yang ingin dilakukan klasifikasi pada perhitungan. Penentuan kelas a^* ditentukan dengan (Schneider, 2004):

$$a^* = \arg \max_{a \in A} \prod_j p(f_j|b)^{f_{ij}} \times p(a)$$

2. 4 Maximum Entropy

Metode Maximum Entropy ini mencari distribusi $p(a|b)$ yang akan memberikan nilai *entropy* maksimum. Pada (MacKay, 2003), Maximum Entropy didefinisikan sebagai rata-rata nilai informasi yang maksimum untuk suatu himpunan kejadian X dengan distribusi nilai probabilitas yang seragam. Yang dimaksud dengan distribusi nilai probabilitas seragam adalah distribusi yang menggunakan faktor ketidakpastian yang minimum atau dapat disebut sebagai distribusi yang memakai

asumsi sesedikit mungkin. Dengan menggunakan asumsi yang minimal, maka distribusi yang didapatkan merupakan distribusi yang paling mendekati kenyataan. Pencarian distribusi probabilitas yang paling memberikan nilai *entropy* yang maksimum dilakukan dengan tujuan mendapatkan distribusi probabilitas terbaik yang mendekati kenyataan. Selain itu, dengan menggunakan metode ini diharapkan juga dapat memodelkan permasalahan yang ada pada dunia nyata dan model yang dihasilkan tersebut merupakan gambaran yang paling mirip dengan kehidupan nyata. Pendekatan metode ini akan dijelaskan lebih lanjut dengan menjelaskan terlebih dahulu pengertian akan *entropy* dan model dari metode Maximum Entropy dari suatu permasalahan.

2.4.1. Entropy

Entropy merupakan rata-rata dari himpunan informasi yang terkandung dalam suatu kumpulan kejadian $X = \{x_1, x_2, \dots, x_n\}$. Himpunan informasi yang terkandung pada suatu kejadian dapat dinyatakan sebagai:

$$h(x) = \log_2 \frac{1}{p(x)}$$

, dengan $h(x)$ merupakan himpunan informasi dari suatu kejadian x dan $p(x)$ merupakan probabilitas dari kemunculan kejadian x . Nilai dari $h(x)$ dinyatakan dalam ukuran bit dan jumlah bit pada $h(x)$ merupakan banyaknya bit yang diperlukan untuk merepresentasikan himpunan informasi dari suatu kejadian x . Semakin besar nilai $h(x)$ maka semakin banyak pula informasi yang dimiliki oleh $h(x)$. *Entropy* dari kumpulan kejadian X dapat dinyatakan sebagai:

$$H(x) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

Dengan metode Maximum Entropy, hasil yang diinginkan adalah nilai $H(p)$ yang maksimum. Nilai *entropy* yang maksimum akan tercapai apabila nilai dari X seragam sehingga mengakibatkan $p(x)=1/|X|$, dengan $|X|$ merupakan kardinalitas dari X . Dalam proses klasifikasi, untuk mendapatkan nilai maksimum yang seragam tidaklah sesederhana dengan membagi nilai 1 dengan nilai kardinalitas X . Pencarian distribusi probabilitas tersebut juga harus memenuhi batasan-batasan

yang ada dengan mengetahui fakta ataupun data yang dimiliki. Hal ini akan dibahas lebih lanjut pada model Maximum Entropy (subbab 2.4.2).

2.4.2 Model Maximum Entropy

Pemodelan dengan menggunakan Maximum Entropy digunakan untuk mencari distribusi yang seragam dari suatu kumpulan probabilitas. Dalam melakukan proses klasifikasi, penggunaan Maximum Entropy mirip dengan penggunaan Naïve Bayes, dimana dengan menggunakan metode ini akan dicari nilai *conditional probability* $p(a|b)$ dari suatu kelas a jika diketahui dokumen b , untuk suatu himpunan kelas $A = \{a_1, a_2, \dots, a_p\}$ dan $B = \{b_1, b_2, \dots, b_q\}$. Penentuan kelas a dari dokumen b akan dilihat dengan mencari nilai probabilitas $p(a|b)$ yang maksimum dari distribusi probabilitas dengan *entropy* maksimum. Dalam menentukan distribusi yang seragam untuk setiap pasangan $a \in A$ dan $b \in B$, pencarian ini haruslah memenuhi batasan-batasan yang timbul dari fakta yang ada. Fakta dari data *training* dapat dinyatakan sebagai fungsi fitur $f_j: (a,b) \rightarrow \{0,1\}$ yang dipelajari dari kumpulan dokumen B , dengan ketentuan:

$$f_j(a, b) = \begin{cases} 1, & \text{jika } f_j \text{ muncul di dokumen } b \text{ pada kelas } a \\ 0, & \text{jika } f_j \text{ tidak muncul di dokumen } b \text{ pada kelas } a \end{cases}$$

Sebagai contoh, misalkan B adalah kumpulan artikel berita olahraga $\{b_1, b_2, \dots, b_q\}$ dan A adalah himpunan cabang olahraga $\{\text{basket, sepak bola, tenis}\}$. Probabilitas $p(\text{basket}|b_1)$ adalah probabilitas kemungkinan cabang olahraga basket dibahas pada artikel berita b_1 . Fitur $f_j(a|b)$ dapat juga dilihat sebagai probabilitas kemunculan sebuah kata pada dokumen b untuk cabang olahraga a , sebagai contoh:

$$f_j(a, b) = \begin{cases} 1, & \text{jika kata } f_j \text{ muncul di artikel berita } b \text{ pada cabang olahraga } a \\ 0, & \text{jika kata } f_j \text{ tidak muncul di artikel berita } b \text{ pada cabang olahraga } a \end{cases}$$

Batasan-batasan atau fakta-fakta yang telah diketahui dalam proses pembelajaran dengan data *training*, dimasukkan dalam penghitungan sebagai nilai ekspektasi dari suatu nilai fitur, sebagai berikut;

$$E_{\tilde{p}} f_j = \sum_{(a,b) \in (A,B)} \tilde{p}(a, b) f_j(a, b)$$

, dengan $\tilde{p}(a, b)$ merupakan probabilitas kemunculan bersama (*joint probability*) pasangan a dan b . Probabilitas ini dapat dihitung dengan persamaan sebagai berikut:

$$\tilde{p}(a, b) = \frac{1}{|D|} xD(a, b)$$

, dengan $|D|$ merupakan jumlah data *training* dan $D(a, b)$ merupakan fungsi yang akan mengembalikan jumlah dokumen dimana pasangan a dan b ini muncul bersama. Nilai ekspektasi yang dihasilkan dari fakta dibatasi dengan nilai ekspektasi yang ada dari model Maximum Entropy berupa:

$$E_p f_j = \sum_{(a,b) \in (A,B)} p(a, b) f_j(a, b)$$

Dan memenuhi batasan sebagai berikut:

$$E_p f_j = E_{\tilde{p}} f_j$$

Dengan menggunakan batasan yang ada, distribusi probabilitas pada model Maximum Entropy dibuat menyerupai dengan distribusi yang terbentuk dengan menggunakan data *training*. Model Maximum Entropy mencari distribusi probabilitas yang memiliki *entropy* paling maksimum dari suatu himpunan P yang memenuhi fakta-fakta sebagai berikut.

$$P = \{p | E_{\tilde{p}} f_j = E_p f_j, j = \{1 \dots n\}\}$$

$$p^* = \arg \max_{p \in P} H(p)$$

Dengan persamaan $H(p)$:

$$H(p) = - \sum_{(a,b) \in (A,B)} p(a, b) \log(a, b)$$

P adalah himpunan yang berisikan distribusi probabilitas p yang memenuhi batasan $E_p f_j = E_{\tilde{p}} f_j$ dan p^* merupakan anggota dari himpunan nilai probabilitas dengan *entropy* paling maksimum yang memenuhi batasan tersebut.

2.4.3 Model Parametrik Maximum Entropy

Model Maximum Entropy mencari distribusi probabilitas yang memiliki entropy paling maksimum dengan juga menyelesaikan fungsi Lagrangian (λ_j) yang ditambahkan pada batasan yang ada sebelumnya. Berikut merupakan persamaan Lagrangian yang perlu diselesaikan (Berger, Pietra, & Pietra, 1996):

$$\Lambda(p, \lambda) = H(p) + \sum_j \lambda_j (E_{\tilde{p}} f_j - E_p f_j, j)$$

Dengan menggunakan batasan yang seperti ini, distribusi probabilitas dengan *entropy* paling maksimum adalah sebagai berikut:

$$p^* = \arg \max_{p \in P} \Lambda(p, \lambda)$$

Pada metode Maximum Entropy nilai dari probabilitas $p^*(a,b)$ yaitu nilai probabilitas kelas a dengan dokumen b dapat dicari dengan menyelesaikan persamaan *Langragian* $\Lambda(p, \lambda) = 0$ sehingga akan didapatkan :

$$-\nabla H(p) = \nabla \sum_j \lambda_j (E_{\tilde{p}} f_j - E_p f_j, j)$$

Solusi dari persamaan sebelumnya akan menjadi:

$$-H_{p(a_1, b_1)} = \sum_j \lambda_j f_j(a_1, b_1) \quad -H_{p(a_2, b_2)} = \sum_j \lambda_j f_j(a_2, b_2) \quad \dots \quad -H_{p(a_j, b_k)} = \sum_j \lambda_j f_j(a_i, b_k)$$

, dengan nilai dari $-H_{p(a_i, b_k)}$ merupakan $H(p)$ yang diturunkan terhadap variabel $p(a_i, b_k)$ dan $\sum_j \lambda_j f_j$ merupakan $\sum_j \lambda_j (E_{\tilde{p}} f_j - E_p f_j, j)$ yang diturunkan terhadap variabel $p(a_i, b_k)$. Pencarian nilai untuk tiap $p(a,b)$ dengan $(a,b) \in (A, B)$ dilakukan dengan menyelesaikan persamaan di atas. Nilai untuk satu probabilitas $p(a,b)$ dicari dengan menyelesaikan (Xia, 2006):

$$-H_{p(a_1, b_1)} = \sum_j \lambda_j f_j(a_1, b_1)$$

$$-(\log_e p(a, b) - 1) = \sum_j \lambda_j f_j(a, b)$$

$$p(a, b) = e^{\sum_j \lambda_j f_j(a, b) - 1}$$

Sehingga akan didapatkan :

$$p^* = \frac{e^{\sum_j \lambda_j f_j(a, b)}}{e \times Z(b)}$$

Dengan nilai Z(b):

$$Z(b) = \sum_a \exp \left(\sum_j \lambda_j f_j(a, b) \right) = p(b)$$

Persamaan yang telah didapat tersebut merupakan suatu model parametrik untuk Maximum Entropy. Bentuk lain dari model parametrik yang diberikan (Ratnaparkhi, 1997) adalah:

$$p^*(a, b) = \pi \prod_{j=1}^n \alpha_j^{f_j(a, b)}$$

Model tersebut merupakan bentuk yang ekuivalen dengan $\pi = \frac{1}{e}$ dan $\lambda_j = \ln \alpha_j$. Kedua model tersebut mendapatkan nilai $p^*(a, b)$ dengan terlebih dahulu mencari nilai parameter λ_j atau α_j berdasarkan data *training* yang ada. Nilai dari parameter tersebut menyatakan bobot dari fitur $f_j(a, b)$, untuk setiap pasangan $(a, b) \in (A, B)$. Perbedaan dari dua model tersebut terletak dari cara atau algoritma yang dipakai untuk menghitung parameter yang akan digunakan. Model pertama menggunakan algoritma yang bernama *Improved Iterative Scaling* (IIS) sedangkan untuk model kedua algoritma yang digunakan adalah *Generalized Iterative Scaling* (GIS). IIS merupakan algoritma yang dimodifikasi dari algoritma GIS. Pembahasan kedua algoritma tersebut tidak dibahas pada subbab ini.

Proses klasifikasi suatu dokumen termasuk ke dalam suatu topik dapat dilakukan dengan menganggap dokumen tersebut sebagai sebuah vektor yang berisi kemunculan dari fitur-fitur $f_j(a, b)$ lalu mencari nilai probabilitas kelas $a \in A$ dari dokumen tersebut. Dokumen tersebut diputuskan termasuk ke dalam suatu kelas a dengan memilih probabilitas nilai $p(a, b)$ yang paling besar. Nilai probabilitas

$p(a,b)$ yang paling besar juga merupakan nilai probabilitas $p(a|b)$ yang paling besar. Hal ini dapat diketahui karena nilai $p(b)$ untuk setiap kelas akan sama. Kelas a^* sebagai hasil klasifikasi b dinyatakan sebagai :

$$a^* = \arg \max_{a \in A} p(a, b)$$

