Aplikasi Vector Space Model dalam Pencarian Arsip Karya Tulis Mahasiswa ITB

Fairuz Astra Pratama /13514104
Program Studi Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
pratamafairuz@gmail.com

***Abstract*—Pemodelan *database* merupakan salah satu konsep penting untuk mempermudah pencarian informasi dalam penyimpanan informasi seperti kumpulan makalah. Salah satu metode yang dikembangkan untuk melakukan hal ini adalah *Vector Space Model* yang mengubah isi dokumen dan *query* pengguna menjadi Vektor Euclidian dan menggunakan konsep besar sudut antara dua vektor untuk menentukan kedekatan antara keduanya dalam meranking relevansi dokumen untuk tiap query.**

***Keywords*—Vektor Euclidian, *Database*, *Vector Space Model*, Pencarian Informasi**

I. PENDAHULUAN

Baik dalam rangka lomba, tugas dari dosen, maupun tugas akhir, setiap mahasiswa ITB telah menulis setidaknya satu karya tulis selama masa kuliahnya sebelum diberikan gelar sarjana kepadanya. Karya tulis ini merupakan bukti bahwa para mahasiswa telah melalui proses pembelajaran dan mampu melakukan penelitian yang bermanfaat bagi masyarakat umum nantinya.

Didalamnya terdapat berbagai macam kajian pustaka dari berbagai cabang ilmu pengetahuan mengenai permasalahan permasalahan ataupun inovasi baru yang akan sangat bermanfaat bagi masyarakat umum. Suatu koleksi karya ilmiah yang tersusun baik akan menjadi sumber pengetahuan yang sangat bergunadan aplikatif.

Sayangnya hal ini tidaklah berlku bagi sebagian besar karya yang dibuat oleh para mahasiswa ITB. Setelah diadakannya sidang dan kelulusan, karya yang mereka buat pun diletakan di rak, dan kemudian dilupakan keberadaannya. Mencari informasi spesifik dari kumpulan karya tulis ini pun juga sangatlah sulit, karena tempat penyimpanan mereka yang tersebar serta tidak adanya sistem pencarian yang memadai.

Oleh karena itulah perlu dibuat *database* dan sistem pencarian informasi yang memadai untuk memudahkan akses informasi ke karya karya mahasiswa ITB hingga seluruh universitas di Indonesia nantinya.

Pembuatan database dapat berupa peng-*upload*-an semua isi atau setidaknya isi abstrak karya karya tulis yang telah dibuat ke internet, ataupun pembuatan server lokal penampung semua informasi tersebut.

Aspek berikutnya yang perlu diperhatikan adalah pembuatan sistem penarsipan iniadalah pencarian, dengan menggunakan sebuah kumpulan kata yang disebut *querry* masukan pengguna, kita harus dapat menampilkan kumpulan karya tulis yang relevan terhadapnya, diurutkan dari relevansi yang tertinggi ke terendah.

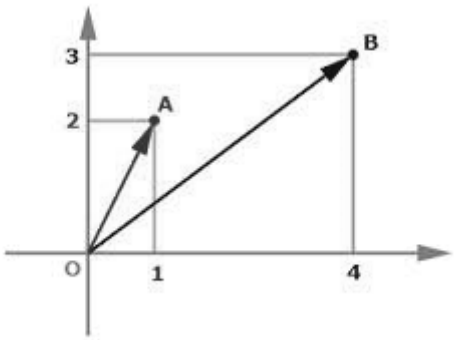
Dengan kata lain, dibutuhkan semacam Mesin Pecari agar masyarakat dapat dengan mudah mengambil informasi yang dibutuhkan dari kumpulan karya karya yang telah dibuat mahasiswa ITB dan/atau universitas lainnya. Ada banyak cara meraih hal ini, salah satunya adalah melalui *vector space model*, yang akan dibahas lebih lanjut di makalah ini.

II. DASAR TEORI VEKTOR

Vektor pada dasarnya adalah sebuah besaran yang mempunyai arah. Sebuah vektor sederhana dapat direpresentasikan sebagai sebuah garis berpanah, dimana ujung panah merupakan titik asal vektor dan ujung panah merupakan titik akhirnya.

2.1 Vektor di Ruang 2 & 3 Dimensi

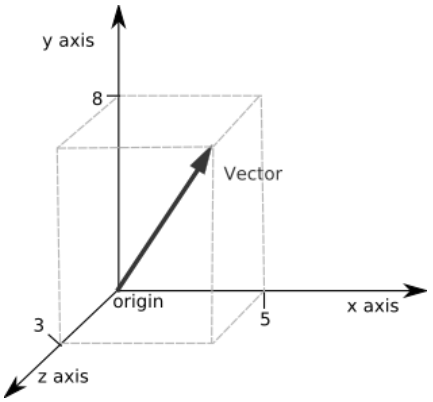
Dalam ruang dua dimensi, vektor bisa dibagi menjadi 2 unsur: komponen horizontal x / i, dan komponen vertikal y / j. Dari dua komponen ini dapat ditentukan besar dan arah suatu vektor.



Gambar 2.1.1 Vektor di ruang dua dimensi,
[1] http://higherorderfun.com/blog/2009/06/13/math-for-game-programmers-03-geometrical-representation-of-vectors,diakses pada 22 November 2015

Pada gambar diatas vector A terdiri dari 1 unit komponen horizontal dan 2 unit vertical yang dapat dinotasikan sebagai : **A** = **i** + 2**j** dan juga vector B sebagai **B** =4**i** + 3**j**

Sedangkan dalam ruang 3 dimensi, vektor dapat direpresentsikannya sebagai kombinasi dari 3 komponen



Gambar 2.1.2 Vektor di ruang tiga dimensi, vector V diatas dapat dinotasikan sebagai : **V** = 5**i** + 8**j** + 3**k**
[2]https://mathemotion.wikispaces.com/Vector+Definitions, diakses pada 22 November 2015

Dari membagi bagi sebuah vector ke komponen komponennya, besar vector pun dapat dengan mudah dicari menggunakan rumus Pythagoras. Besaran ini disebut sebagai norm (dimana norm dari vector **V** dinotasikan sebagai ||**V**||), dan merupakan besaran skalar yang menggambarkan panjang vector yang bersangkutan.
Di ruang dua dimensi, rumus norm sebuah vector adalah:

$$\boldsymbol{V} = a\boldsymbol{i} + b\boldsymbol{j}, \qquad \left|\left|\boldsymbol{V}\right|\right| = \sqrt{(\boldsymbol{a}^2 + \boldsymbol{b}^2)}$$

Sedangkan di ruang tiga dimensi,

$$\boldsymbol{V} = a\boldsymbol{i} + b\boldsymbol{j} + c\boldsymbol{k}, \qquad \left|\left|\boldsymbol{V}\right|\right| = \sqrt{(\boldsymbol{a}^2 + \boldsymbol{b}^{2+} + \boldsymbol{c}^2)}$$

Selain besarnya, komponen kedua dari vector yang tidak kalah pentingnya adalah arah / orientasi. Di ruang vector, orientasi dapat dinyatakan sebagai besar sumbu antara suatu vector dengan vector lain / sumbu dimensi. Sebagai contoh, di ruang dua dimensi, kita biasa menyatakan arah vector sebagai besar derajat antara vector tersebut dengan sumbu X positif.

Untuk menentukan perbedaan sudut antara dua vector v1 dan v2 dapat digunakan rumus berikut :

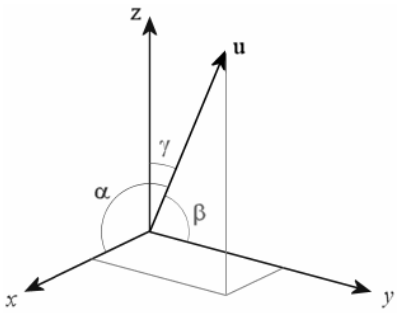
$$\cos\theta = \frac{\boldsymbol{v_1}.\boldsymbol{v_2}}{\left|\left|\boldsymbol{v_1}\right|\right|\left|\left|\boldsymbol{v_2}\right|\right|}$$

Dimana **v1**.**v2** adalah dot-product dimana jika :

$$\boldsymbol{v_1} = a_1\boldsymbol{i} + b_1\boldsymbol{j} + c_1\boldsymbol{k},$$
$$\boldsymbol{v_2} = a_2\boldsymbol{i} + b_2\boldsymbol{j} + c_2\boldsymbol{k},$$

maka

$$\boldsymbol{v_1}.\boldsymbol{v_2} = \boldsymbol{a_1}\boldsymbol{a_2} + \boldsymbol{b_1}\boldsymbol{b_2} + \boldsymbol{c_1}\boldsymbol{c_2}$$



Gambar 2.1.2 Ilustrasi penggunaan sudut sebagai pernyataan orientasi vector di ruang 3 dimensi
[3] http://www.intmath.com/vectors/7-vectors-in-3d-space.php, diakses pada 22 November 2015

Konsep ini menjadi kurang praktis di dimensi lebih tinggi, sehingga penampilan orientasi melalui rasio antar komponen vector lebih sering digunakan, namun konsep jarak antar dua vector ini akan sangat berguna di pembahasan berikutnya.

2.2 Vektor Euclidian

Sampai saat ini telah dibahas mengenai vector di ruang 2 dan 3 dimensi. Walaupun representasi geometric vector di dimensi 4 dan keatas belum dapat digambarkan, konsep dan aljabar vektor masih dapat bekerja dengan baik dengan basis basis yang lebih tinggi.

Orang yang pertama kali mempelajari mengenai vektor di ruang R^n ini bernama Euclidian, karena itulah vektor yang berada di ruang n-Euclides ini dinamai vektor Euclidian. Bentuk umum vektor Euclidian ini adalah :

$$\boldsymbol{V} = (\boldsymbol{v_1}, \boldsymbol{v_2}, \boldsymbol{v_3}, \ldots, \boldsymbol{v_n})$$

Dimana **V** adalah sebuah vektor di ruang n-Euclidian, **n** adalah dimensi terbesar ruang tempat V berada, dan $\mathbf{v_1}$, $\mathbf{v_2}$, $\mathbf{v_n}$ adalah komponen komponen vektor V seperti i,j,k di vektor 2-3 dimensi

Selain bentuk yang hampir serupa dengan vektor yang di ruang 2-3 dimensi, operasi operasi yang telah dibahas seperti Norm, Dot Product, dan Perbedaan Sudut vektor Euclidian juga mirip dengan rumus sebelumnya.

Di ruang dimensi n-Euclidian, Norm sebuah vector didefinisikan sebagai:

$$\boldsymbol{V} = a\boldsymbol{v}_1 + b\boldsymbol{v}_2 + \cdots + z\boldsymbol{v}_n, \qquad \left|\left|\boldsymbol{V}\right|\right| = \sqrt{(\boldsymbol{a}^2 + \boldsymbol{b}^2 + \boldsymbol{c}^2)}$$

Sedangkan dot-product antara dua vektor dimensi n-Euclidian V dan W didefinisikan sebagai

$$\boldsymbol{V} = a_1\boldsymbol{v_1} + b_1\boldsymbol{v_2} + \cdots + z_1\boldsymbol{v_3},$$
$$\boldsymbol{W} = a_2\boldsymbol{v_1} + b_2 v_2 + \ldots + z_2\boldsymbol{v_3},$$

maka

$$\boldsymbol{V}.\boldsymbol{W} = a_1a_2 + b_1b_2 + \cdots + z_1z_2$$

Dengan adanya kedua rumus diatas, maka sudut antara dua vektor euclidian **V** dan **W** dapat didefinisikan dengan menggunakan rumus yang sama seperti vektor ruang 2-3 dimensi, yaitu :

$$\cos\theta = \frac{\boldsymbol{V}.\boldsymbol{W}}{||\boldsymbol{V}||||\boldsymbol{W}||}$$

Walaupun definisi geometri dari sudut antar dua vektor di ruang n-Euclidian belum jelas, kita dapat menggunakan sudut untuk membayangkan seberapa miripnya dua buah vektor, yang akan menjadi salah satu dasar dari *Vector Space Model*.

III. PEMODELAN DOKUMEN - VEKTOR

Vector Space Model (yang selanjutnya akan disebut sebagai Model Ruang Vektor), adalah suatu metode yang digunakan untuk merepresentasikan makalah (atau dokumen lainnya), sebagai vektor Euclidian. Dalam pemodelan dokumen, metode ini menggambarkannya sebagai sebuah vektor di ruang dimensi N, dimana setiap komponen arah N_i menggambarkan frasa / kata tertentu yang mungkin terdapat pada dokumen tersebut (dapat disebut indeks). Selain itu besar komponen vektor N_i yang menggambarkan seberapa banyak kata itu digunakan di dokumen tersebut (bernilai 0 jika tidak ditemukan).

	d_1	d_2	d_3	d_4	d_5	d_6
ship	1	0	1	0	0	0
boat	0	1	0	0	0	0
ocean	1	1	0	0	0	0
voyage	1	0	0	1	1	0
trip	0	0	0	1	0	1

Gambar 3.1 Contoh Model Ruang Vektor pada 5 dokumen yang berbeda dengan 5 istilah yang berbeda
[4] https://alaathoughts.wordpress.com/2012/08/22/vector-space-model/, diakses pada 14 Desember 2015

Sebagai contoh pada gambar diatas dapat dilihat bahwa pada dokumen satu terdapat istilah kapal, laut, dan perjalanan; sedangkan pada dokumen empat terdapat istilah mengenai Perjalanan. Dari model ruang vektor diatas dapat kita perkirakan isi tiap dokumen, dan dokumen mana yang paling relevan jika kita ingin mencari mengenai Kapal yang dapat berlayar di samudra.

Langkah pertama untuk membuat mesin pencari berdasarkan pemodelan ruang vektor adalah memodelkan makalah makalah menjadi bentuk vektor. Tentu saja untuk menghitung jumlah kemunculan kata di tiap makalah dan menyimpannya dalam bentuk matriks seperti pada gambar 3.1 bukanlah masalah besar, namun masih perlu dilakukan beberapa operasi terhadap data tersebut sebelum dapat kita gunakan.

Dalam Sub-Bab ini, akan dibahas berbagai macam metode yang digunakan dalam mengubah sebuah makalah / dokumen lainnya menjadi representasi vektor. Pada dasarnya, langkah ini dapat dibagi menjadi dua tahap, yaitu Pengindeksan Dokumen, Pembobotan Istilah.

3.1 Pengindeksan Dokumen (*Document Indexing*)

Pada gambar 3.1 dapat dilihat bahwa dokumen tiga hanya bernilai tidak-nol pada “dimensi” kapal. Apakah ini berarti bahwa dokumen tersebut hanya terdiri dari kata “kapal” saja ?, tentu saja tidak. Dari semua kata yang mendirikan makalah, pasti terdapat beberapa kata yang tidak mencerminkan isi makalah tersebut, seperti kata hubung “dan”, “yang”, hingga “di”

Tujuan dari pelaksanaan pengindeksan dokumen ini adalah untuk menghilangkan kata kata tersebut (yang dapat membentuk sekitar 40-50% isi makalah), sehingga “dimensi” yang tersisa pada model ruang vektor makalah tersebut hanya terdiri dari istilah yang mengandung makna, seperti “vektor”, “makalah”, dan “Euclidian” yang digunakan pada makalah ini. Secara umum, terdapat dua cara untuk melakukan hal ini.

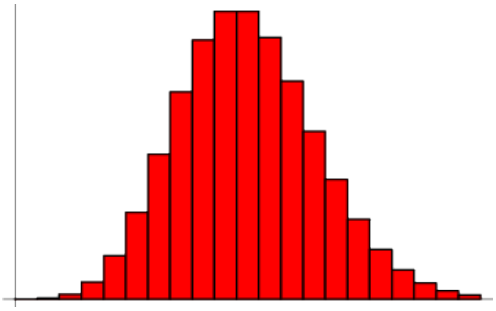
Cara pertama (yang paling intuitif) adalah dengan membuat semacam daftar yang mengandung semua kata yang tidak signifikan (juga disebut sebagai kata fungsi) dalam suatu bahasa. Saat program menelusuri makalah untuk mencatat jumlah kata, program juga akan *crosscheck* kata yang didapat dengan list kata fungsi ini (penggunaan fungsi hash disarankan untuk mempercepat pengecekan), jika ada, maka kata tersebut akan diabaikan.

Salah satu kelemahan dari metode ini adalah, tentu saja terdapat ketergantungan daftar kata terhadap bahasa yang digunakan sebuah makalah. Hal menyebabkan sebelum pengubahan makalah ke vektor dilakukan, algoritma harus menelusuri beberapa kata pertama makalah untuk menentukan bahasa apa yang digunakan makalah, dan perlunya disediakan daftar kata penghubung untuk tiap bahasa. Untungnya, untuk membuat sistem pencarian khusus untuk makalah Mahasiswa ITB hal ini tidak terlalu menjadi masalah karena mayoritas ditulis dalam bahasa Indonesia / Inggris.

Selain itu juga terdapat metode lain yang disebut sebagai Pengindeksan Probabilistik (Probability Indexing) yang didasarkan pada asumsi bahwa terdaat perbedaan statistik (seperti jumlah dan persebaran) antara kata fungsi dan kata yang mengandung makna. Salah satu contoh sederhana dari pengindeksan probabilistik adalah dengan mengurutkan kata yang telah kita hitung jumlahnya dari rendah di kiri ke tinggi di kanan, dan menganalogikannya sebagai sebuah grafik persebaran Poisson untuk menebak kata mana saja yang mempunyai makna yang penting.

Pada gambar 3.1.1, graf tersebut menggambarkan kepentingan tiap kata dalam sebuah makalah, dimana kata yang memiliki jumlah menengah lebih mencerminkan isi makalah daripada kata yang sedikit digunakan dan yang sering sekali digunakan dalam makalah tersebut.

Hal ini cukup masuk akal, karena kata yang memiliki frekuensi penggunaan sangat tinggi kemungkinan besar merupakan kata hubung seperti "yang" dan "dari", dan kata yang memiliki frekuensi penggunaan rendah tidak mungkin mencerminkan isi makalah. Oleh karena itu, kata kata tersebut mempunyai tingkat kepentingan yang rendah dan dapat dihapus dari matriks kata-frekuensi.



Gambar 3.1.1 Contoh graf persebaran Poisson
[4]http://mathworld.wolfram.com/PoissonDistribution.html, diakses pada 15 Desember 2015

3.2 Pembobotan Istilah (*Term Weighting*)

Setelah kita mendapatkan jumlah semua kata yang mengandung makna dalam sebuah dokumen dari melakukan pengindeksan, langkah berikutnya untuk mengubah sebuah makalah menjadi vektor adalah Pembobotan Istilah / Term Weighting. Pada tahap ini kita akan memberikan "nilai" atau bobot pada tiap kata pengandung makna dalam suatu makalah.

Sampai saat ini, nilai yang kita gunakan dalam membobot adalah jumlah kemunculan kata dalam sebuah dokumen, hal ini tidaklah adil bagi makalah yang relatif lebih pendek. Karena itulah dibutuhkan metode penilaian yang lebih baik lagi. Ada beberapa metode indexing yang tersedia, masing masing dengan ketepatan yang bervariasi Namun secara keseluruhan ada 3 hal yang mempengaruhi nilai sebuah istilah dalam makalah :

- Frekuensi Kemunculan Kata,

  Karena semua kata fungsi telah dihilangkan di Pengindeksan makalah, maka secara intuitif semua kata pasti mengandung makna, dan kata yang digunakan paling banyak di dokumen tersebut hampir selalu mencerminkan isi dokumen lebih baik daripada kata yang digunakan lebih sedikit.

- Panjang Dokumen,

  Misalkan makalah A berisi informasi mengenai perkapalan dan makalah B membahas transportasi secara umum. Kedua makalah menggunakan kata "kapal" didalamnya, namun makalah B panjangnya dua kali lipat makalah A dan menggunakan kata jumlah "kapal" yang sama dengan makalah A.

  Apakah itu berarti relevansi kata "kapal" di kedua makalah itu sama?, tentu saja tidak; relevansi kata "kapal" di makalah A secara intuitif harus bernilai lebih tinggi. Karena itulah panjang makalah harus tetap diingat saat memberi nilai istilah yang digunakan.

Dengan membagi jumlah kata yang dihitung dengan panjang dokumen, kita dapat membuat satuan nilai yang lebih baik, yaitu Frekuensi Penggunaan Kata.

- Kemunculan Kata di Dokumen Lain,

  Misalkan di *database* makalah kita terdapat 10.000 makalah, dan salah satu makalah (sebut saja makalah C) mengandung kata kunci istilah "Graf" dan "Makan Siang" Di database terdapat 1000 makalah lain yang mengandung kata kunci "Graf", namun hanya 5 yang mengandung frasa "Makan Siang". Jika kedua istilah itu digunakan dengan frekuensi yang sama, apakah keduanya pantas berbobot sama ?, tentu saja tidak.

  Walaupun keduanya bernilai sama dalam mencerminkan isi makalah, istilah unik dalam makalah itu yang membedakannya dari ribuan makalah lainnya pantas mendapat bobot yang lebih, dan beberapa eksperimen telah membuktikan bahwa dengan memperhatikan faktor ini dalam pemberian bobot istilah, hasil pencarian akan lebih baik. Oleh karena itulah faktor ini digunakan dalam kebanyakan metode pembobotan istilah yang digunakan secara luas.

Salah satu metode yang sering digunakan dalam menghasilkan suatu nilai relatif yang menggambarkan seberapa pentingnya sebuah istilah dalam dokumen adalah tf-idf(*term frequency-inverse document frequency*) sesuai namanya, metode ini menilai dengan menghitung dua hal, *term frequency* (frekuensi penggunaan kata) dan *inverse document frequency* yang menggambarkan kemunculan kata di dokumen lain dengan rumus berikut :

$$TF(t) = \frac{(\text{Jumlah Kata t di Makalah})}{(\text{Jumlah Kata di Makalah})}$$

yang merupakan frekuensi penggunaan kata,

$$\text{IDF(t)} = \log \frac{(\text{Jumlah Total Makalah})}{\text{Jumlah Makalah yang Mengandung Kata t}}$$

yang merupakan frekuensi frekuensi kemunculan kata di dokumen lain, dan

$$TF.IDF(t) = TF(t).IDF(t)$$

yang merupakan nilai istilah t dalam suatu makalah, relatif terhadap semua makalah dalam database.

$$\begin{pmatrix} & T_1 & T_2 & \dots & T_t \\ D_1 & w_{11} & w_{21} & \dots & w_{t1} \\ D_2 & w_{12} & w_{22} & \dots & w_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ D_n & w_{1n} & w_{2n} & \dots & w_{tn} \end{pmatrix}$$

Gambar 3.2.1 Matriks bobot tiap term pada tiap dokumen

[5]http://blogs.msdn.com/themes/blogs/generic/post.aspx?WeblogApp=spt&y=2008&m=03&d=05&WeblogPostName=information-retrieval-search-basic-ir-models&GroupKeys, diakses pada 15 Desember 2015

Setelah semua istilah diberi nilai, maka kita akan mendapat sebuah matriks yang merepresentasikan vektor Euclidian dimensi n untuk semua makalah dalam *database* (n adalah jumlah istilah pengandung makna, dan besar n_i adalah bobot istilah ke-i dalam makalah tersebut). Tentu saja akan terdapat beberapa istilah yang hanya dikandung oleh makalah tertentu.

Jika suatu istilah tidak digunakan suatu makalah, namun digunakan makalah lain dalam database, dan karenanya istilah tersebut menjadi salah satu dimensi VSM database tersebut, maka nilai dimensi istilah makalah yang tidak mengandung tersebut adalah 0.

IV. MERANKING RELEVANSI MAKALAH

Setelah kita mengubah *database* makalah kita menjadi sekumpulan vektor euclidian dengan dimensi N. Setelah ini kita bisa menentukan urutan relevansi semua makalah di *database* berdasarkan permintaan pengguna melalui seubuah *query*. Query yang dimasukan berupa sebiah kalimat yang menyatakan informasi apa yang diinginkan pengguna seperti "Sejarah Perjalanan Kapal" dan kita ingin dapat menyajikan urutan makalah dari yang paling relevan hingga yang paling tidak relevan

Dengan adanya model ruang vektor untuk tiap makalah yang telah disiapkan, hal ini dapat dilakukan dengan cukup mudah, langkah pertama yang kita perlu lakukan adalah mengubah *query* masukan pengguna ke bentuk vektor dan membandingkan kedekatannya dengan tiap vektor makalah di *database*.

Mengubah *query* menjadi vektor memiliki konsep yang sama dengan mengubah sebuah dokumen menjadi vektor, bahkan lebih sederhana, Kita tidak perlu melakukan pengindeksan, karena walaupun vektor *query* mengandung dimensi untuk kata hubung, besar dimensi itu untuk mayoritas dokumen akan berukuran 0 (tidak ada) dan tidak akan menganggu proses perbandingan. Kita hanya perlu mengukur frekuensi penggunaan tiap kata pada *query* tersebut untuk menentukan nilai tiap istilah, dan memasukannya ke dalam bentuk matriks untuk membentuk representasi vektornya.

Dapat dikatakan bahwa dalam Model Ruang Vektor, kita memandang query sebagai salah satu dokumen juga. Kemudian kita dapat melakukan perbandingan *query* dengan tiap vektor makalah di *database*.

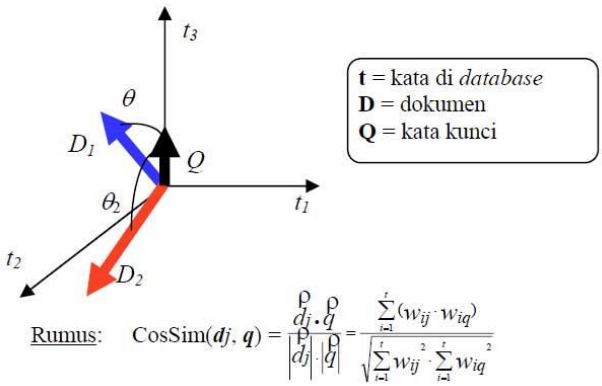
Seperti yang dibahas di dasar teori, kita dapat mengukur seberapa dekat dua vektor Euclidian dengan menggunakan konsep sudut diantara keduanya. Misal kita ingin membandingkan *query* Q dan dokumen D dengan model ruang vektor sebagai berikut :

$$\boldsymbol{Q} = 3\boldsymbol{v_1} + 2\boldsymbol{v_3}, \qquad \boldsymbol{D} = 2\boldsymbol{v_1} + 1\boldsymbol{v_2},$$

Dapat dilihat bahwa dimensi kedua vektor itu tidak sama, untuk mengatasinya, kita anggap suatu dimensi (yang menggambarkan kemunculan suatu istilah dalam query dan/atau makalah) adalah 0. Dengannya kita dapatkan vektor model baru :

$$\boldsymbol{Q} = 3\boldsymbol{v_1} + 0\boldsymbol{v_2} + 2\boldsymbol{v_3},$$
$$\boldsymbol{D} = 2\boldsymbol{v_1} + 1\boldsymbol{v_2} + 0\boldsymbol{v_3},$$

Dengan begitu, kita dapat mengukur besar cosinus sudut diantaranya, dan dapat menebak kedekatan antara query yang dimasukan pengguna dengan isi makalah (Secara umum semakin besar sudut semakin kecil nilai cosinusnya) dan dengan mengulanginya untuk setiap vektor makalah di *database* kita dapat mengetahui urutan relevansi makalah kita relatif terhadap sebuah *query*



Gambar 4.1 Representasi perangkingan relevansi dokumen terhadap *query* di Model Ruang Vektor dimensi 3
[6]https://liyantanto.wordpress.com/2011/06/28/pencarian-dengan-metode-vektor-space-model-vsm/, diakses pada 15 Desember 2015

Selain cara diatas, masih ada metode luntuk mengukur kedekatan dua dokumen dalam VSM, seperti menggunakan Jaccard and Dice coefficients. Namun, karena cara diatas adalah metode yang paling intuitif, untuk kali ini, hanya metode ini saja yang akan dibahas

V. KESIMPULAN

Vector Space Model adalah salah satu cara pemodelan database selain pemodelan Boolean yang merepresentasikan dokumen sebagai sekumpulan istilah yang mempunyai bobotnya masing masing dalam bentuk vektor. Sebelum pencarian dapat dilakukan, semua dokumen dalam database harus diubah ke bentuk vektor masing masing. Setelahnya, dengan mengubah *query* pengguna menjadi vektor di dimensi yang sama, maka meranking relevansi dokumen akan dapat dilakukan.

Walau merupakan salah satu metode pemodelan database terbaik, cara ini memiliki beberapa kelemahan, jauh dari ideal. Salah satu permasalahannya adalah jika database terlalu besar, mesin harus membandingkan dua vektor euclidian dengan jutaan dimensi dan jutaan dokumen, yang akan memakan waktu yang cukup lama.

REFERENSI

[7]http://cogsys.imm.dtu.dk/thor/projects/multimedia/textmining/node5.html, diakses pada 14/12/2015
[8]https://alaathoughts.wordpress.com/2012/08/22/vector-space-model/, diakses pada 14/12/2015
[9]http://www.slideshare.net/dalal404/document-similarity-with-vector-space-model diakses pada 14/12/2015
[10]http://mathworld.wolfram.com/PoissonDistribution.html diakses pada 14/12/2015
[11]http://nlp.stanford.edu/IRbook/html/htmledition/term-frequency-and-weighting-1.html, diakses pada 15/12/2015
[12]http://langvillea.people.cofc.edu/DISSECTIONLAB/Emmie'sLSI-SVDModule/p3module.html diakses pada 15/12/2015
[13]https://dafiqur.files.wordpress.com/2013/02/bab-4-euclidean-vector-spaces.pdf, diakses pada 13/12/2015
[14]http://nuurilqolbii.blogspot.co.id/2013/02/rumus-rumus-vektor-matematika.html, diakses pada 13/12/2015

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 9 Desember 2015

Fairuz Astra Pratama - 13514104