

Pencarian Pasal Pada UU ITE Berdasarkan Kasus Cyber Crime Dengan Metode Latent Semantic Indexing (LSI)

Pramana Yoga Saputra¹, Dika Rizky Yuniarto², Satria Setya Arissandy³

^{1,2,3}Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang
¹pramanay@gmail.com, ²dikarizkyuniarto@polinema.ac.id, ³satryasetya@gmail.com

Abstrak - Sebagai negara hukum, Indonesia memiliki peraturan-peraturan yang mengatur hubungan antar masyarakat, salah satunya adalah peraturan tentang informasi dan transaksi elektronik. Peraturan tersebut tertulis pada UU No 19 tahun 2016 tentang perubahan atas undang-undang nomor 11 tahun 2008 UU ITE. UU ITE dibuat dikarenakan pada era digital ini marak terjadi kasus kriminal terutama pada bidang informasi dan transaksi elektronik (Cyber Crime) seperti, penipuan online, perjudian online, penggunaan data yang ilegal, dan berita palsu (hoax). Kasus-kasus tersebut membuktikan bahwa UU ITE belum tersosialisasi dengan baik kepada masyarakat, sehingga perlu sebuah sistem pencarian informasi (Information Retrieval) berdasarkan kasus tentang undang-undang ini agar memudahkan dalam memahaminya. Pada penelitian ini, information retrieval atau sistem temu kembali informasi digunakan untuk mencari pasal-pasal di UU ITE yang relevan pada suatu kasus berdasarkan deskripsi kasus tersebut. Pada pencarian pasal tersebut digunakan metode Latent Semantic Indexing (LSI). Teknik dalam LSI mengadopsi proses matematis reduksi dimensi Singular Value Decomposition (SVD). Metode cosine similarity digunakan untuk menghitung kesamaan atau kedekatan dokumen pasal dengan query. Sistem ini menggunakan 21 pasal ketentuan beserta 21 pasal pidananya dan 10 query kasus atau deskripsi kejahatan sebagai data uji. Sistem menghitung dan menentukan pasal yang terkait berdasarkan query atau deskripsi kasus kejahatan yang dimasukkan. Tingkat keberhasilan dari metode ini untuk menemukan hasil pencariannya yang relevan dapat dilihat melalui besarnya 83.33% untuk recall, 50% untuk precision dan 62.5% untuk f-measure.

Kata kunci—*information retrieval, latent semantic indexing, singular value decomposition, cosine similarity, text mining, UU ITE*

I. PENDAHULUAN

UU ITE dibuat karena di era digital ini marak terjadi kasus kriminal terutama pada bidang informasi dan transaksi elektronik seperti, penipuan online, perjudian online, penggunaan data yang ilegal, dan berita palsu (Hoax). Kasus-kasus tersebut membuktikan bahwa UU ITE belum tersosialisasi dengan baik kepada masyarakat. Hal ini dapat diketahui dari sebagian besar masyarakat Indonesia yang belum mengerti dan memahami peraturan – peraturan pada UU ITE. Sehingga perlu sebuah sistem pencarian informasi berdasarkan kasus tentang undang-undang ini agar memudahkan dalam memahaminya .

Sistem pencarian informasi pada zaman teknologi ini memunculkan revolusi dan inovasi dalam ilmu pengetahuan, salah satunya adalah teknologi Information Retrieval (Sistem Temu Kembali Informasi). Information Retrieval umumnya dapat dikembangkan dengan Vector Space Model (VSM), yaitu fungsi yang digunakan untuk menghitung besarnya derajat kemiripan antara vektor. Dalam perhitungan VSM digunakan perhitungan Cosine Similarity, yaitu untuk menghitung kesamaan atau kedekatan antar dokumen [7].

Telah banyak penelitian tentang Sistem Temu Kembali Informasi yang menggunakan VSM terutama dengan metode Cosine Similarity. Namun metode tersebut hanya menentukan kemiripan saja (Similarity) yang berarti tidak mampu membedakan dan mengenali kata synonym dan polysemy. Menurut Kamus Besar Bahasa Indonesia (KBBI), sinonim adalah kata yang memiliki bentuk yang berbeda dengan kata lain namun memiliki arti, makna atau pengertian yang sama atau mirip. Sinonim bisa disebut juga dengan persamaan makna kata atau padanan kata. Sedangkan polisemi merupakan kata-kata yang memiliki makna atau arti lebih dari satu.

Untuk menutupi kekurangan dari VSM tersebut, maka dapat menggunakan metode Latent Semantic Indexing (LSI), yang dapat menemukan hubungan semantik antar kata dan ketekaitan makna dengan kata (Arum Sari & Ridok, 2012). Maka dari itu, penulis mengusulkan judul “Pencarian Pasal pada UU ITE berdasarkan kasus Cyber Crime” dengan metode LSI (Latent Semantic Indexing) yang dapat memberikan kemudahan bagi masyarakat dan penegak hukum dalam mendapat informasi atau bahkan menganalisa suatu kasus yang berkaitan tentang kejahatan Cyber. Oleh karena itu pada jurnal ini diusulkan sebuah sistem pencarian pasal UU ITE menggunakan metode Latent Semantic Indexing (LSI). Metode LSI digunakan untuk mencari keterkaitan makna antar kata pada sebuah kalimat yang diharapkan agar hasil pencarian menggunakan LSI dapat lebih relevan.

II. TINJAUAN PUSTAKA

A. UU ITE

Undang-undang Informasi dan Transaksi Elektronik atau Undang Undang nomor 11 tahun 2008 atau UU ITE adalah UU yang mengatur tentang informasi serta transaksi elektronik, atau

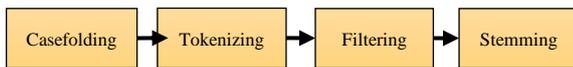
teknologi informasi secara umum. UU ini memiliki yurisdiksi yang berlaku untuk setiap orang yang melakukan perbuatan hukum sebagaimana diatur dalam Undang-Undang ini, baik yang berada di wilayah Indonesia maupun di luar wilayah hukum Indonesia, yang memiliki akibat hukum di wilayah hukum Indonesia dan/atau di luar wilayah hukum Indonesia dan merugikan kepentingan Indonesia [4].

B. Sistem Temu Kembali Informasi

Information Retrieval System atau Sistem Temu Kembali Informasi akan memberikan informasi keberadaan dan keterangan informasi atau dokumen yang berhubungan dengan permintaan pengguna, dimana permintaan pengguna tersebut memakai bahasa natural sebagai bahasa query. Information Retrieval System memberikan kemudahan kepada pengguna dalam mempresentasikan atau menyampaikan kebutuhan informasinya ke dalam sistem dalam bentuk query[9].

C. Text Processing

Suatu data dokumen teks, mempunyai struktur kata yang tidak teratur, maka dalam melakukan proses text mining diperlukan beberapa proses tambahan yang bertujuan untuk menyiapkan dan mengubah data teks mentah menjadi lebih terstruktur. Salah satu tahapan dari text mining adalah preprocessing text. Pada tahap preprocessing text ini dilakukan proses seleksi data berupa text pada setiap dokumen, yang kemudian akan dilakukan proses selanjutnya. Pada tahap preprocessing ini melalui beberapa tahapan yaitu case folding, tokenizing, filtering, dan stemming [2]. Detai dari proses preprocessing akan di gambarkan pada Gambar 1:



Gambar 1. Tahapan Preprocessing

- *Casefolding*

Tahap pertama dalam preprocessing yaitu case folding. Proses ini digunakan untuk melakukan konversi atau mengubah seluruh huruf pada teks dokumen pasal maupun query menjadi huruf kecil (lower case), hanya huruf ‘a’ sampai dengan ‘z’ saja yang diterima. Karakter selain huruf yaitu tanda baca atau delimiter akan dihilangkan pada proses case folding ini.

- *Tokenizing*

Tokenizing merupakan proses pemotongan string pada suatu dokumen berdasarkan tiap kata yang menyusunnya. Sehingga proses ini dilakukan pemecahan data latih berupa kalimat menjadi kata-kata. Tahapan ini dilakukan setelah data latih pasal melewati tahap *Case Folding*.

- *Filtering*

Filtering merupakan proses penghapusan kata-kata yang tidak penting pada data latih. Pada proses ini kita harus membuat atau mempunyai list kata-kata yang tidak penting atau yang biasa disebut dengan Stopword. Kata-kata hasil pemrosesan sebelumnya yaitu tokenizing, akan dicocokkan dengan data pada stopwords. Jika kata yang diperiksa sama dengan kata yang ada dalam daftar stopwords, maka kata tersebut akan dihapus.

- *Stemming*

Stemming merupakan tahapan dimana dilakukan pencarian terhadap kata dasar dari tiap kata hasil filtering. Stemming mengubah berbagai bentuk kata ke dalam suatu bentuk kata dasar yang sama. Tahap ini dilakukan pemotongan atau penghilangan imbuhan seperti prefix (awalan) dan suffix (akhiran) serta confix (awalan dan akhiran).

D. TF-IDF

TF-IDF merupakan metode pembobotan term yang banyak digunakan sebagai metode pembandingan terhadap metode pembobotan baru. Pada metode ini, perhitungan bobot term dalam sebuah dokumen dilakukan dengan mengalikan nilai Term Frequency dengan Inverse Document Frequency.

Term Frequency (TF) adalah faktor yang menentukan bobot term pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen tersebut. Untuk menghitung bobot term frequency menggunakan persamaan:

$$w_{tf_{t,d}} = \begin{cases} 1 + \log_{10} tf_{t,d}, & \text{if } tf_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

dimana $w_{tf_{t,d}}$ adalah bobot *term frequency (tf weight)*, $tf_{t,d}$ adalah banyaknya kemunculan *term/kata t* dalam dokumen *d*.

Inverse Document Frequency (IDF) adalah pengurangan dominansi term yang sering muncul di berbagai dokumen. Term yang banyak muncul di berbagai dokumen, dapat dianggap sebagai term umum (common term) sehingga tidak penting nilainya. Kata yang muncul pada sedikit dokumen harus dipandang sebagai kata yang lebih penting (uncommon term) daripada kata yang muncul pada banyak dokumen. IDF dihitung menggunakan persamaan :

$$idf_t = \log_{10} N/df_t \quad (2)$$

dimana idf_t adalah bobot *inverse document frequency*, *N* adalah banyaknya dokumen yang ada, dan df_t adalah banyaknya dokumen yang mengandung *term/ kata t*.

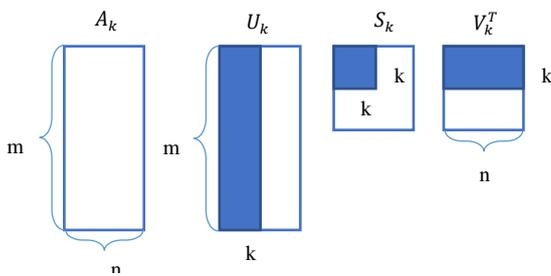
Kemudian rumus untuk menghitung bobot *tf-idf* untuk setiap kata *t* dalam dokumen *d*, merupakan hasil perkalian antara *tf weight* dengan *idf* dihitung menggunakan persamaan :

$$w_{t,d} = w_{tf_{t,d}} \times idf_t \quad (3)$$

dimana $w_{t,d}$ adalah bobot *tf-idf* kata *t* dalam dokumen *d*, $w_{tf_{t,d}}$ adalah bobot *term frequency (tf weight)*, dan idf_t adalah bobot *inverse document frequency*.

E. Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) merupakan model matematis yang digunakan untuk reduksi dimensi data. Ditunjukkan pada Gambar 2:



Gambar 2. Ilustrasi matriks SVD

Singular Value Decomposition (SVD) merupakan model matematis yang digunakan untuk reduksi dimensi data. Proses SVD dilakukan dengan mendekomposisi matriks menjadi tiga bagian, seperti pada Gambar 1.

Matriks U dan V adalah matriks orthonormal, dimana baris pada matriks U menggambarkan banyaknya baris pada matriks A, sementara kolom pada matriks V menggambarkan banyaknya kolom pada matriks A. K-rank digunakan untuk mereduksi dimensi dari matriks A. Matriks S merupakan matriks simetris yang berisi nilai positif di sepanjang diagonal, daerah selain diagonal berisi 0 [8].

F. Latent Semantic Indexing (LSI)

Reduksi dari SVD digunakan dalam LSI. LSI merupakan salah satu bentuk teknik proses temu kembali dengan menggunakan Vector Space Model (VSM), untuk menemukan informasi yang relevan. Keterkaitan makna dalam LSI sifatnya tersembunyi. Fungsi matematis di dalam LSI mampu menemukan hubungan semantik antar kata. Representasi dari LSI dapat dilihat pada persamaan:

$$q' = q^T \cdot U_k \cdot S_k^{-1} \quad (4)$$

dimana q' adalah query vector representasi dari LSI, q^T adalah transpose dari pembobotan TF-IDF query, U_k adalah reduksi dimensi k dari matriks U, dan S_k^{-1} adalah inverse dari reduksi dimensi k matriks S [8].

G. Vector Space Model (VSM)

Vector Space Model (VSM) adalah cara konvensional yang biasa digunakan dalam proses temu kembali informasi. Prosesnya dengan menghitung kemiripan dua buah vektor, yaitu antara vektor dari corpus dan vektor dari query. Untuk melakukan perhitungan terhadap kemiripan antar vektor digunakan rumus Cosine Similarity.

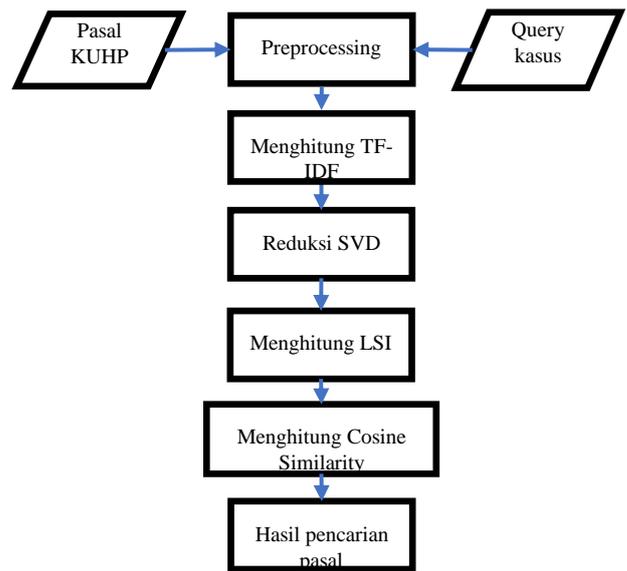
$$\text{CosSim}(d_i, q) = \frac{d_i \cdot q}{|d_i| |q|} \quad (5)$$

dimana d_i adalah dokumen vector ke i yang diambil dari nilai matriks V, q adalah kata kunci/query vector hasil perhitungan LSI [7].

III. METODE PENELITIAN

Penelitian dilakukan melalui langkah langkah sebagai berikut:

- Memasukkan data berupa pasal-pasal UU ITE. Kumpulan pasal tersebut disebut sebagai corpus. Inputan sistem terdiri atas corpus (pasal UU ITE) dan query (deskripsi kasus kejahatan).
- Preprocessing file corpus dan query.
- Membentuk matriks pembobotan TFIDF pada corpus dan query.
- Mendekomposisi matriks pembobotan corpus dengan SVD.
- Menghitung query vector yang merupakan representasi dari LSI.
- Mencari kemiripan antara corpus dan query dengan cosine similarity.
- Pengurutan nilai cosine similarity secara descending order.
- Pengambilan top-n teratas nilai cosine similarity hasil pengurutan.
- Melakukan evaluasi dari hasil penelitian dengan Precision, Recall, dan F-Measure.



Gambar 3. Alur Kerja Sistem Secara Umum

IV. HASIL PENELITIAN

Data yang digunakan adalah pasal UU ITE dengan jumlah pasal pada data training yaitu sebanyak 21 pasal, menggunakan deskripsi kasus kejahatan (query) sebanyak 10 kasus, yang terdiri dari berbagai macam kasus kejahatan yang terkait dengan UU ITE. Berikut adalah Contoh kasus yang digunakan pada penelitian ini.

TABEL I. CONTOH KASUS

Query	Kasus
q2	Terjadi penyadapan percakapan antara menteri BUMN Rini Mariani dan Dirut PT PLN. Potongan rekaman antara keduanya terkesan berbicara soal fee proyek, Namun Rini Mariani merasa namanya dicemarkan karena rekaman sudah dipotong-potong dengan tujuan menjatuhkan nama baiknya.
q2	Seorang aktivis antikorupsi bernama Mohammad Trijanto terkait kasus pencemaran nama baik terhadap Bupati Blitar, Rijanto. Trijanto dianggap bersalah atas tindakannya menyebar surat panggilan KPK palsu atas nama Rijanto di media sosial.

Pengujian sistem dilakukan untuk menjamin dan memastikan bahwa sistem yang dirancang dapat berjalan seperti yang diharapkan. Pengujian dilakukan dengan mencocokkan pasal relevan yang diperoleh dari seseorang yang ahli dibidang hukum. Kemudian data uji dari pakar hukum dihitung dengan metode yang biasa digunakan pada Information retrieval, yaitu Precision / Recall dan F-Measure dengan rumusan [5]:

$$Recall = \frac{\text{pasal relevan output sistem}}{\text{pasal relevan}} \quad (6)$$

$$Precision = \frac{\text{pasal relevan output sistem}}{\text{pasal output sistem}} \quad (7)$$

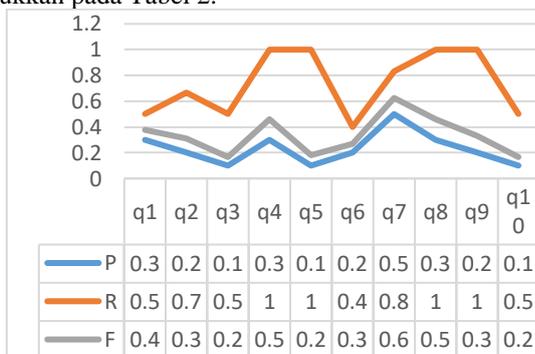
$$F\text{-measure} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (8)$$

Berikut adalah hasil pengujian sistem dengan Precision, Recall, dan F-Measure:

TABEL II. PENGUJIAN PRECISION, RECAL, DAN F-MEASURE

Kasus	Precision (%)	Recall (%)	F-Measure (%)
q1	0.3	0.5	0.375
q2	0.2	0.666667	0.307692308
q3	0.1	0.5	0.166666667
q4	0.3	1	0.461538462
q5	0.1	1	0.181818182
q6	0.2	0.4	0.266666667
q7	0.5	0.833333	0.625
q8	0.3	1	0.461538462
q9	0.2	1	0.333333333
q10	0.1	0.5	0.166666667
Rata-rata	0.23	0.74	0.334592075
%	23	74	33.45920746

Setelah proses perhitungan langkah selanjutnya adalah merepresentasikan hasil ketiganya kedalam bentuk grafik. Pengujian pada Tabel 1 menggunakan output pasal dari sistem berjumlah 10. Pada pengujian tersebut query ke 8 mendapat nilai yang paling tinggi dibanding query lainnya, maka dari itu pada query tersebut perlu dilakukan uji Precision, Recall dan F-Measure dengan mengubah-ubah keluaran dari sistem. Hal ini dilakukan untuk membandingkan dan menganalisa hasil uji sistem secara keseluruhan dengan hasil uji query yang mendapat nilai paling tinggi. Hasil pengujian terhadap query 8 ditunjukkan pada Tabel 2.

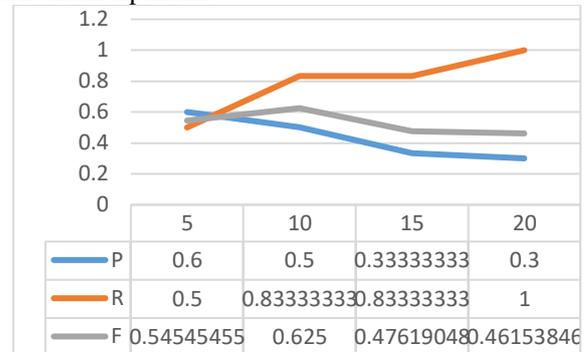


Gambar 4. Grafik Perbandingan nilai P,R dan F

TABEL III. PENGUJIAN PRECISION, RECAL, DAN F-MEASURE PADA QUERY 8.

Output	P	R	F
5	0.6	0.5	0.545454545
10	0.5	0.833333	0.625
15	0.333333	0.833333	0.476190476
20	0.3	1	0.461538462
Rata-rata	0.433333	0.791667	0.527045871
%	43.33333	79.16667	52.70458708

Setelah proses perhitungan langkah selanjutnya adalah merepresentasikan hasil ketiganya kedalam bentuk grafik agar mudah untuk dipahami.



Gambar 5. Grafik Perbandingan nilai P,R dan F pada query 3

Perolehan recall berhubungan dengan kemampuan sistem untuk memanggil dokumen yang relevan, sedangkan ketepatan precision berkaitan dengan kemampuan sistem untuk tidak memanggil dokumen yang tidak relevan. Dari ketiga metode pengujian diatas antara percobaan pertama dan kedua tidak ada perbedaan yang signifikan pada nilai Precision, Recall dan F-Measure. Berdasarkan Gambar 6.1 dan Gambar 6.2 nilai Precision berbeda bahkan ada yang bertolak belakang dengan nilai Recall (lebih tinggi), sedangkan nilai F-Measure selalu berada pada grafik rata-rata antara keduanya. Setelah dilakukan berbagai percobaan dengan merubah nilai output sistem maka dapat disimpulkan bahwa output sistem paling baik berada pada nilai 10.

V. KESIMPULAN

Berdasarkan analisa dan perancangan, implementasi serta pengujian sistem aplikasi Pencarian Pasal pada UU ITE berdasarkan kasus Cyber Crime dengan metode Latent Semantic Indexing (LSI), maka didapatkan kesimpulan:

1. Dengan menerapkan semua yang dibutuhkan untuk membangun metode Latent Semantic Indexing (LSI), seperti text processing, TF-IDF, SVD dan Cosine Similarity dapat membangun sebuah sistem temu kembali informasi pada UU ITE berdasarkan kasus.
2. Berdasarkan hasil pengamatan dan uji coba, metode Latent Semantic Indexing (LSI) berhasil untuk melakukan fungsi pencarian pasal berdasarkan query kasus pengguna terhadap indeks matriks di dalam dokumen pasal.
3. Tingkat keberhasilan dari metode ini dalam menjalankan fungsinya dapat dilihat melalui besarnya nilai uji kinerja

sistem (performance measure) yang menghasilkan nilai 83.33% untuk recall, 50% untuk precision dan 62.5% untuk f-measure, dengan nilai output sistem 10.

DAFTAR PUSTAKA

- [1] Dwi, Smaradhana. 2017. Penerapan Text Mining untuk Melakukan Clustering Data Tweet Shopee Indonesia. Fakultas Matematika dan Ilmu Pengetahuan Alam, ITS. Surabaya.
- [2] Febriawan,Carfin,. 2017. Implementasi Explicit Semantic Analysis (ESA) berbahasa Indonesia menggunakan corpus Wikipedia Indonesia. Politeknik Negeri Malang.
- [3] Informatikalogi.2017. "Text Preprocessing". [online]. Tersedia: <https://informatikalogi.com/text-preprocessing/>
- [4] Kominfo . 2016. "Perubahan UU Tahun 2008". [online]. Tersedia: <https://web.kominfo.go.id/sites/default/files/users/4761/UU%2019%20Tahun%202016.pdf>
- [5] Hartanto, Katarina. 2017. Information Retrieval dengan metode Latent Semantic Indexing
- [6] LSI pada proses Searching dan Klasifikasi Buku: Studi kasus perpustakaan STMIK Bumigora.Teknik Informatika, STMIK Bumigora. Mataram.
- [7] Nurdiana, Ogie. 2016. Perbandingan Metode Cosine Similarity dengan Metode Jaccard Similarity pada aplikasi pencarian terjemahan AL-Quran dalam bahasa Indonesia. Teknik Informatika. Fakultas Sains dan Teknologi.
- [8] Ridok, Achmad. 2015. Pengklasifikasian Dokumen Berbahasa Indonesia dengan pengideksan berbasis LSI. Fakultas Ilmu Komputer Universitas Brawijaya Malang.
- [9] Sutisna, Utis. 2010. Koreksi Ejaan Query bahasa Indonesia Menggunakan Algoritma Damerau Levenshtein. Departemen Ilmu Komputer, IPB. Bogor.