IMPLEMENTASI METODE SUPPORT VECTOR MACHINES UNTUK PENCARIAN INFORMASI BUKU

RIKI HIDAYAT 10108371

Latar Belakang Masalah

1. Jumlah buku yang semakin banyak.
2. Belum adanya sistem pencarian informasi buku untuk mempermudah atau mempercepat pengunjung untuk melakukan pencarian buku.
3. Terbatasanya informasi mengenai identitas buku yang diketahui pengunjung perpustakaan.

Identifikasi Masalah

Bagaimana membangun sistem pencarian informasi buku yang **dinamis**, dengan menerapkan metode ***vector space model*** dan ***support vector machines***.

Maksud dan Tujuan

Maksud

membuat suatu desain dan implementasi sistem pencarian informasi buku yang dinamis di perpustakaan daerah provinsi Jawa Barat dengan menerapkan metode VSM dan SVMs

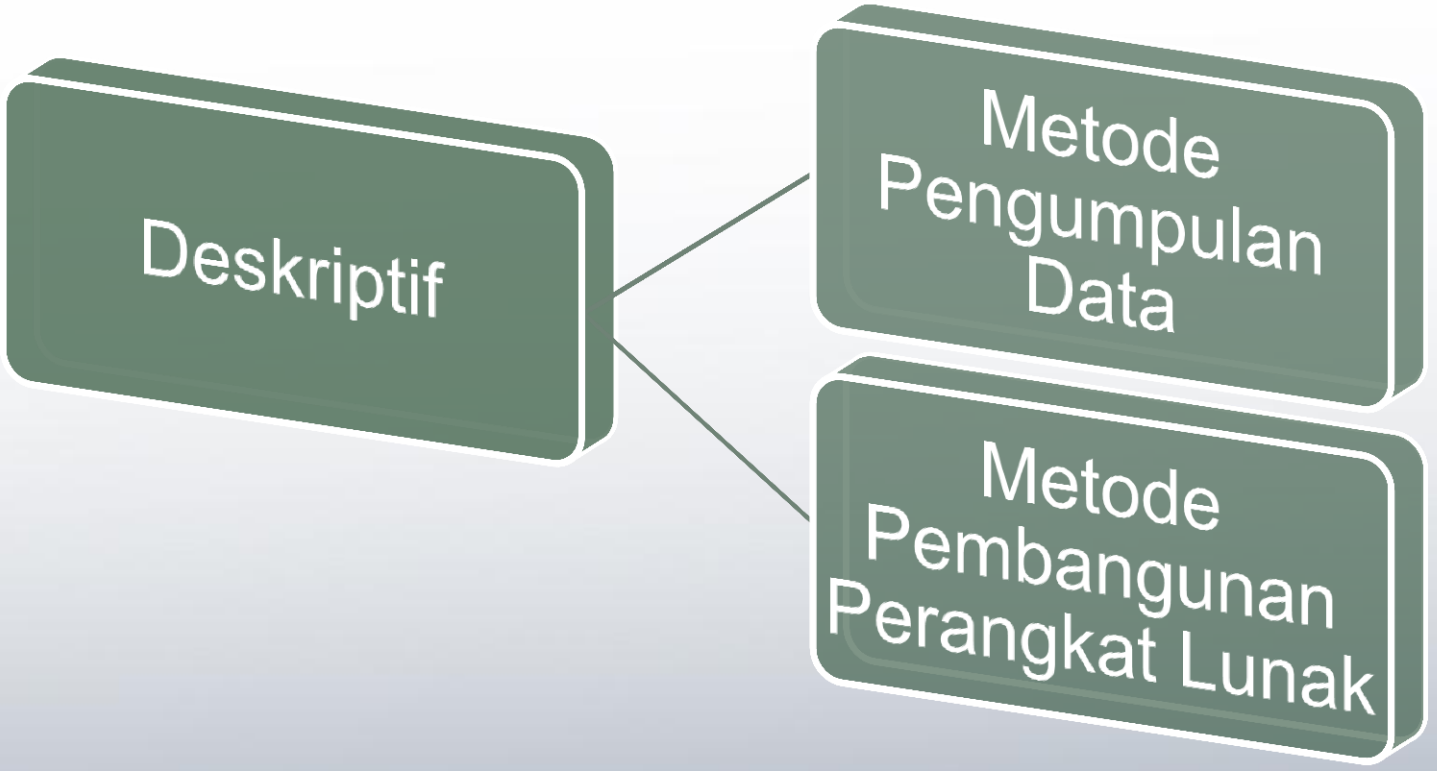
Tujuan

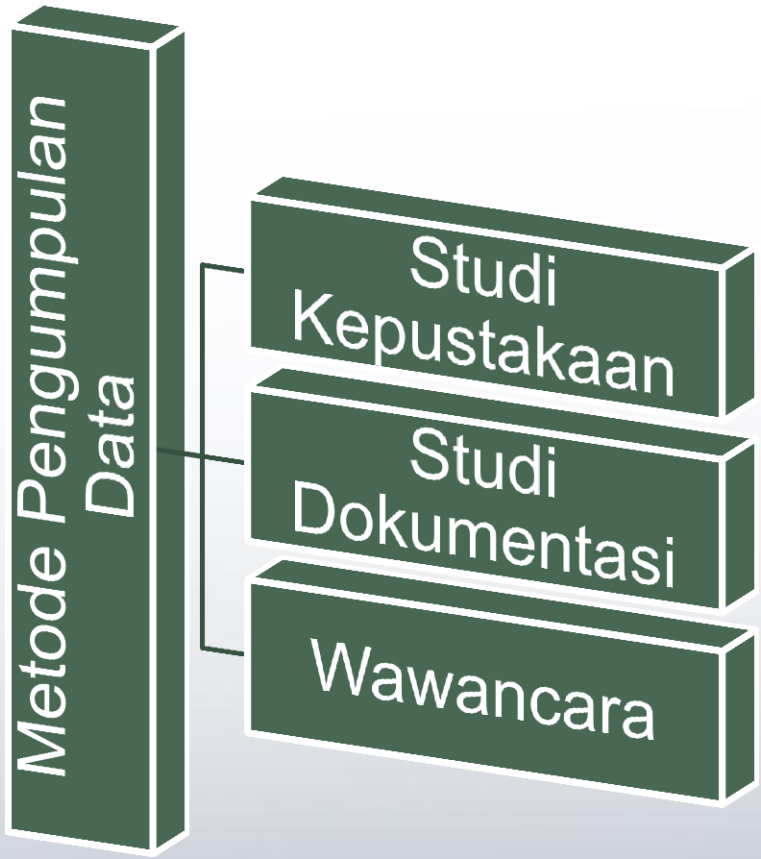
Mempermudah melakukan pencarian informasi buku dengan menggunakan *query* sebagai masukan terhadap sistem.

Batasan Masalah

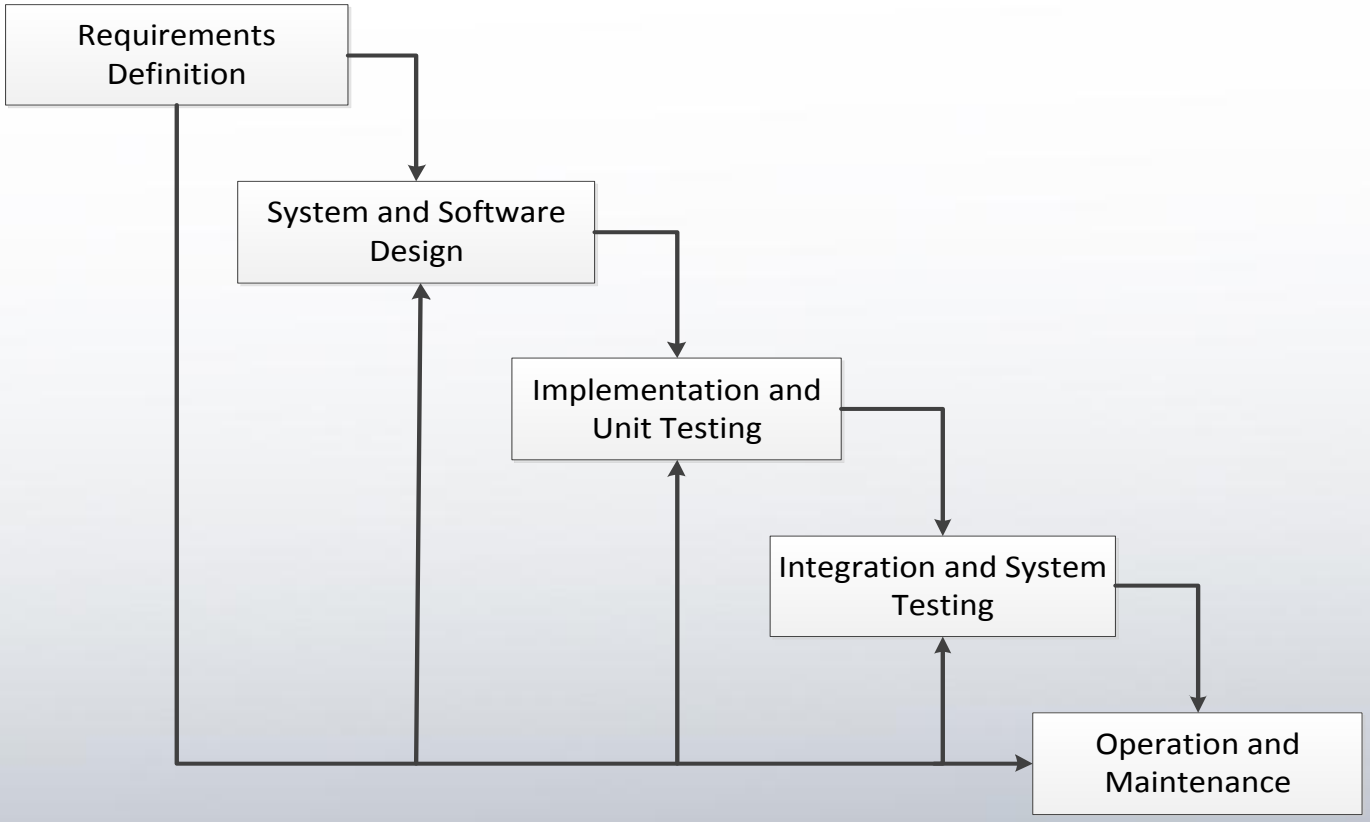
1. Sistem akan dibangun berbasiskan *website*,
2. Sistem yang dibangun adalah sistem pencarian informasi buku yang ada diperpustakaan daerah Bandung, provinsi Jawa Barat,
3. Sistem merupakan pengembangan dari http://www.bapusipda.jabarprov.go.id,
4. Sistem digunakan oleh dua kategori pengguna yaitu operator dan pengunjung situs,
5. Sistem hanya membutuhkan satu pengguna sebagai operator,
6. Informasi yang diberikan sistem yaitu berupa identitas buku (judul, pengarang, dan golongan) dan deskripsi buku,
7. Sistem akan dibangun menggunakan bahasa pemrograman PHP dan *Database Management System* MySQL.

Metodologi Penelitian

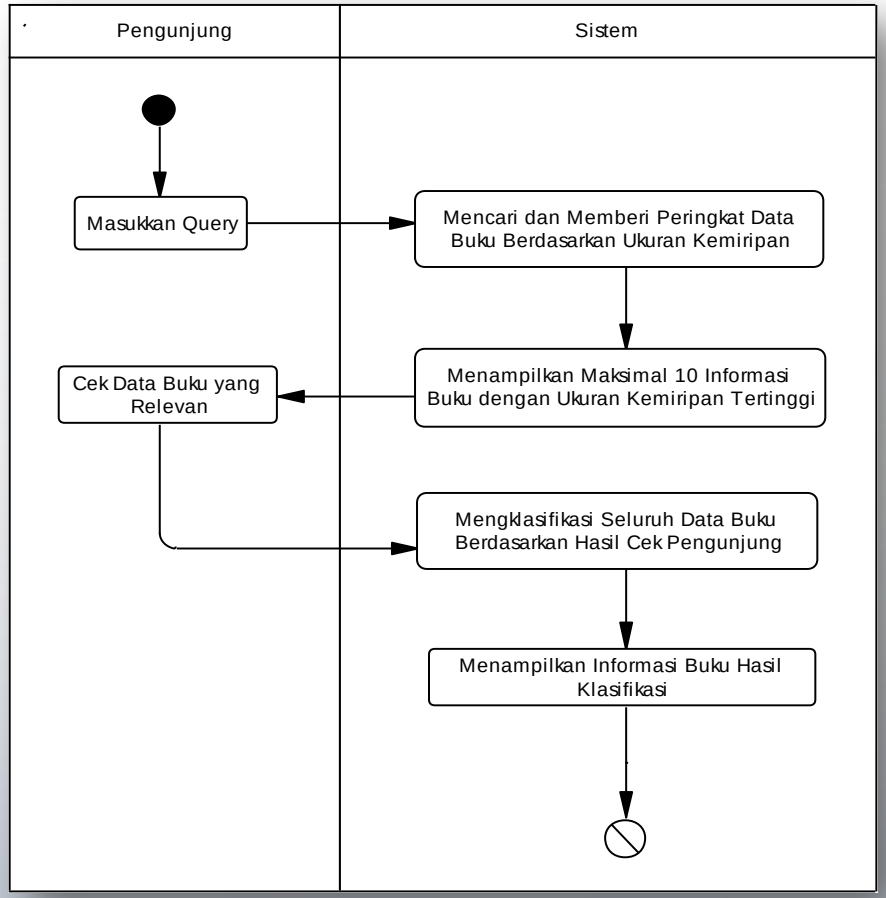



Metode Pengumpulan Data
Studi Kepustakaan
Studi Dokumentasi
Wawancara

Metode Pembangungan Perangkat Lunak



Arsitektur Sistem

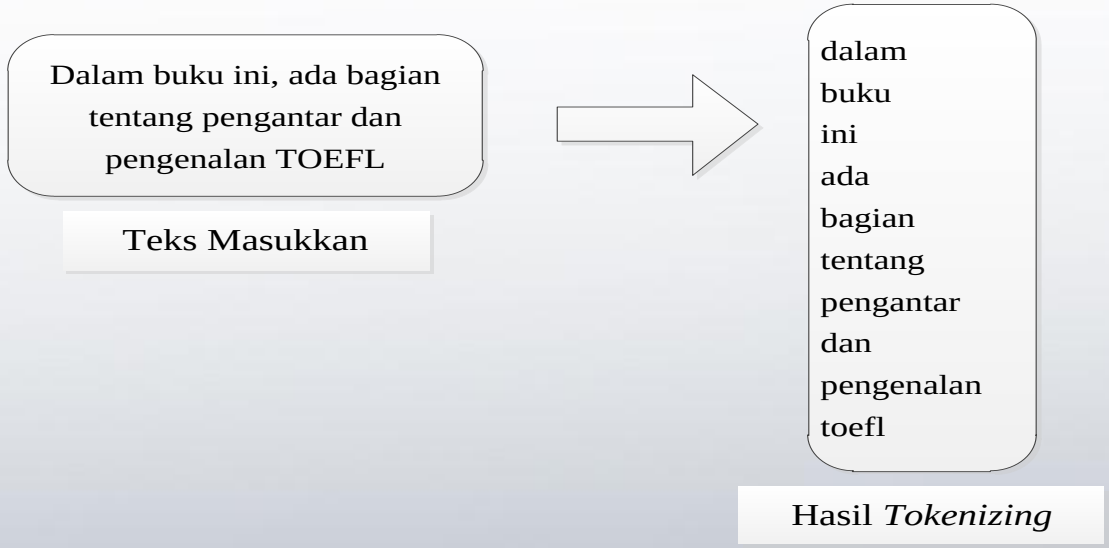


Analisis Metode

1. Tokenizing
2. Filtering
3. Stemming
4. Pembuatan Keyword
5. Vector Space Model (VSM)
6. Support Vector Machines (SVMs)

Tokenizing

proses pemotongan string input berdasarkan tiap kata yang menyusunya serta membedakan karakter-karakter tertentu yang dapat diperlakukan sebagai pemisah kata atau bukan

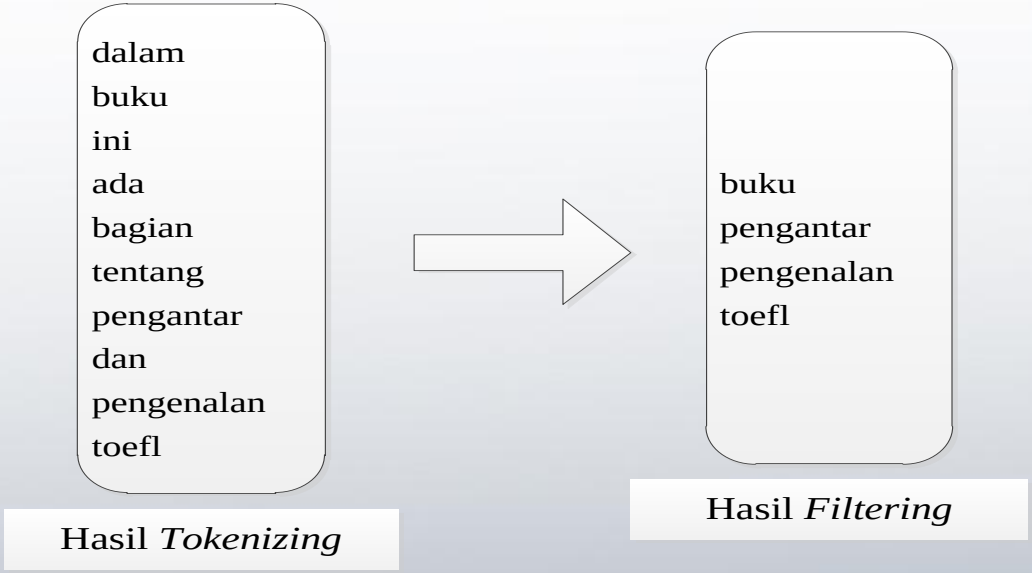


Karakter yang Dihapus

Karakter	Karakter	Karakter	Karakter
!	~	+	\
@	&	=	/
#	*	{	“
$	(	}	‘
%	)	[	:
^	-	]	;
`	-	|	.
,	<	>	?
white space (tab, spasi, enter)			

Filtering

Proses mengambil kata-kata penting dari hasil tokenizing dengan cara menghapus kata yang tidak penting (*stop word*).

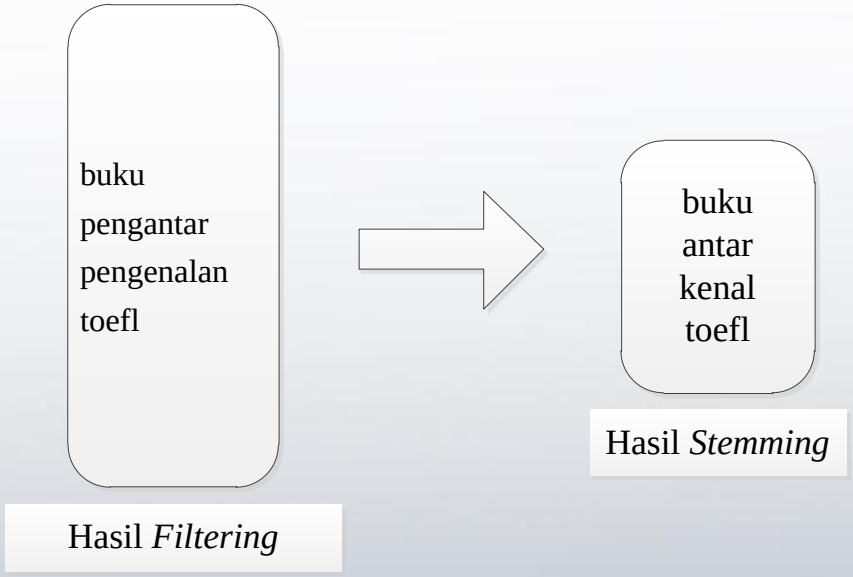


Daftar Stop Word

Daftar Stop Word	Daftar Stop Word
yang	sudah
mampu	tetapi
tentang	bisa
di	melakukannya
setelah	lakukan
semua	memang
hampir	baik
juga	lain
am	pernah
antara	setiap
dan	untuk
ada	dari
seperti	mendapatkan
jadi	punya
karena	telah
of	mr
mrs	…, dan lain-lain

Stemming

Proses mencari kata dasar dari tiap kata hasil *filtering.*



Algoritma Nazief & Adriani

1. Cari kata dikamus.
2. Hapus *Inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”)
3. Hapus *Derivation Suffixes* (“-i”, “-an” atau “-kan”)
4. Hapus *Derivation Prefix* (“be-”, “di-”, “ke-”, “me-”, “pe-”, “se-”, danc “te-”)
5. Melakukan *Recoding*
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*.

Recoding

Aturan	Format Kata	Pemenggalan
1	berV…	ber-V... | be-rV...
2	berCAP…	ber-CAP... dimana C!='r' & P!='er'
3	berCAerV...	ber-CaerV... dimana C!='r'
4	belajar	bel-ajar
5	beC1erC2...	be-C1erC2... dimana C1!={'r'|'l'}
6	terV...	ter-V... | te-rV...
7	terCerV...	ter-CerV... dimana C!='r'
8	terCP...	ter-CP... dimana C!='r' dan P!='er'
9	teC1erC2...	te-C1erC2... dimana C1!='r'
10	me{l|r|w|y}V...	me-{l|r|w|y}V...
11	mem{b|f|v}...	mem-{b|f|v}...
12	mempe...	mem-pe...
13	mem{rV|V}...	me-m{rV|V}... |me-p{rV|V}…
14	men{c|d|j|z}...	men-{c|d|j|z}...
15	menV...	me-nV... | me-tV
16	meng{g|h|q|k}...	meng-{g|h|q|k}...
17	mengV...	meng-V... | meng-kV...
18	menyV...	meny-sV...
19	mempV...	mem-pV... dengan V!='e'
20	pe{w|y}V...	pe-{w|y}V...
21	perV...	per-V... | pe-rV...
22	perCAP	per-CAP... dimana C!='r' dan P!='er'
23	perCAerV...	per-CAerV... dimana C!='r'
24	pem{b|f|V}...	pem-{b|f|V}...
25	pem{rV|V}...	pe-m{rV|V}... | pe-p{rV|V}…
26	pen{c|d|j|z}...	pen-{c|d|j|z}...
27	penV...	pe-nV... | pe-tV...
28	peng{g|h|q}...	peng-{g|h|q}...
29	pengV...	peng-V... | peng-kV...
30	penyV...	peny-sV...
31	pelV...	pe-lV... kecuali "pelajar" yang menghasilkan "ajar"
32	peCerV...	per-erV... dimana C!={r|w|y|l|m|n}
33	peCP...	pe-CP... dimana C!={r|w|y|l|m|n} dan P!='er'
34	terC1erC2...	ter-C1erC2... dimana C1!='r'
35	peC1erC2...	pe-C1erC2... dimana C1!={r|w|y|l|m|n}

Kombinasi Imbuhan Terlarang

Awalan (prefix)	Akhiran (suffix) yang dilarang
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan
te-	-an

Pembuatan Keyword

Keyword yang dimaksud dalam penelitian ini adalah data buku yang merupakan gabungan dari judul, pengarang dan deskripsi buku yang sudah mengalami proses *tokenizing*, *filtering* dan *stemming.*

Aturan Pembuatan Keyword

1. Data buku langsung diketik di form Penambahan atau Pengubahan Data Buku tidak disarankan untuk menyalin melalui media lain, seperti Microsoft Office.
2. Jika ada lebih dari satu pengarang maka gunakan tanda ” ; ” atau “ : ” sebagai pemisah antara pengarang yang satu dengan yang lainnya.
3. Jika ada poin-poin dalam deskripsi gunakan tanda “ - “ sebagai tanda poin dalam kalimat.

Vector Space Model

1. **Pembobotan**
2. **Normalisasi**
3. **Ukuran Kemiripan (Cosine Similiarity)**

Pembobotan

Pembobotan adalah proses merubah *keyword* dan *query* menjadi bentuk vektor.

tf : banyaknya kata yang ada pada sebuah dokumen

df : banyaknya dokumen pada sebuah kata

d : jumlah dokumen

idf : *inverse dokumen frequency*

w : bobot kata

$$idf = \log\left(\frac{d}{f}\right)$$

$$w = tf * idf$$

Normalisasi

$$w(kata_i) = \frac{w(kata_i)}{\sqrt{w(kata_1)^2 + w(kata_2)^2 + \ldots + w(kata_n)^2}}$$

Ukuran Kemiripan

$$sim(d_i, q) = \frac{\sum d_i * q}{\sqrt{\sum d_i^2 * \sum q^2}}$$

Support Vector Machines

1. **Reperesentasi Data**
2. **Pembelajaran dan Klasifikasi**

Representasi data

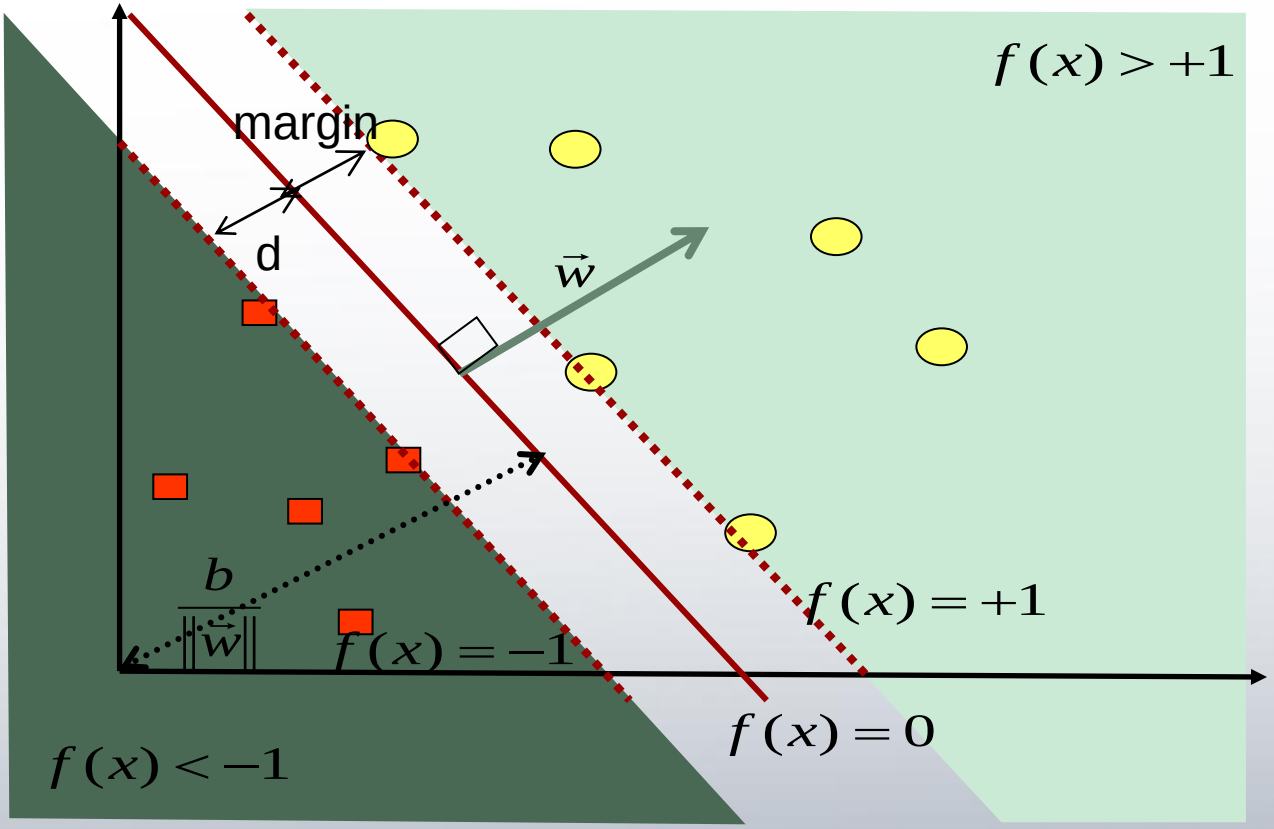
Keyword

7700 peribahasa indonesia drs nur arifin chaniago bagas pratama spd tata bahasa bahasa indonesia kenal peribahasa peribahasa kandung makna dalam hidup itu sebab orang peribahasa sampai maksud ungkap judul 7700 peribahasa indonesia saji lengkap bahasa ringkas padat

Format SVM Light

1 1:0.239011 2:0.348346 3:0.0256246 4:0.119506 5:0.157205 6:0.157205 7:0.157205 8:0.157205 9:0.157205 10:0.157205 11:0.157205 12:0.113099 15:0.0818057 18:0.157205 19:0.157205 20:0.0974526 21:0.0696691 22:0.119506 23:0.157205 24:0.0974526 26:0.157205 27:0.119506 28:0.157205 29:0.157205 33:0.0597528 34:0.0376998 36:0.157205 37:0.157205

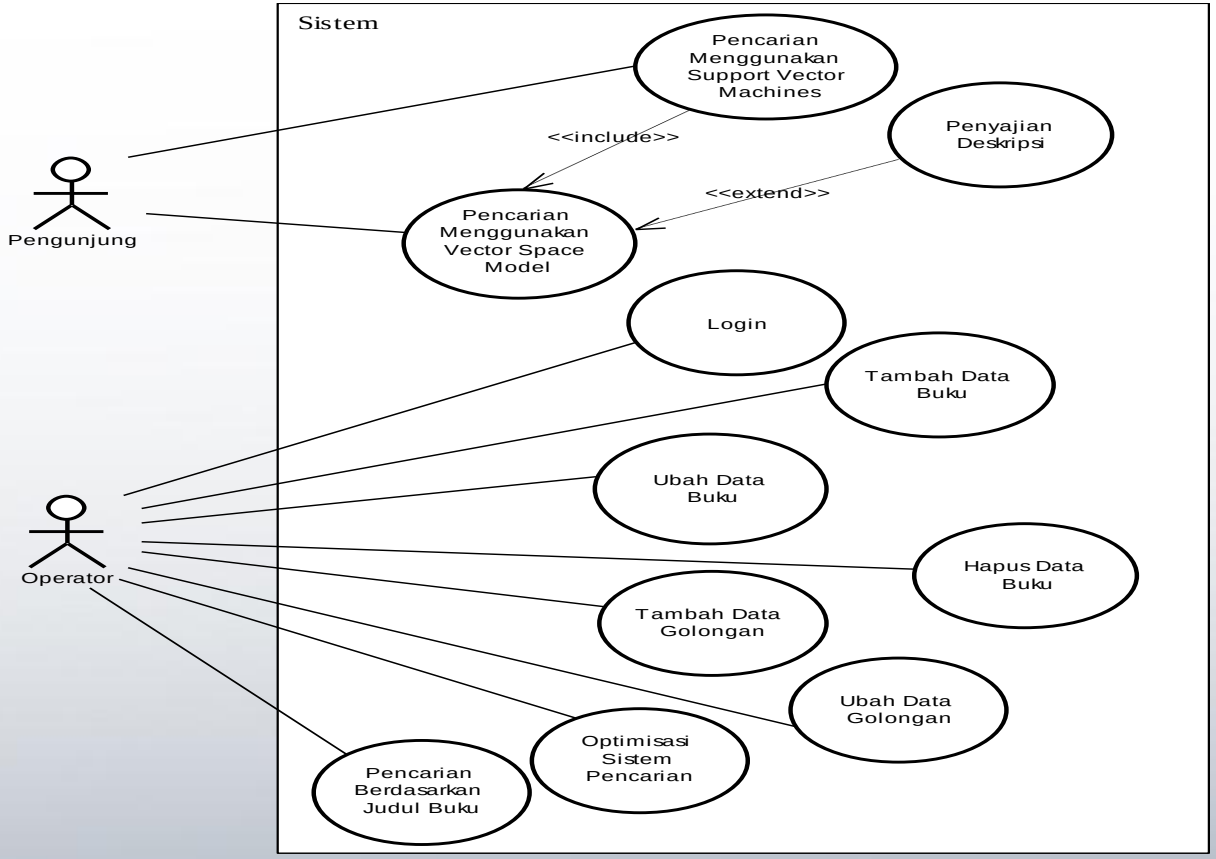
Pembelajaran dan Klasifikasi



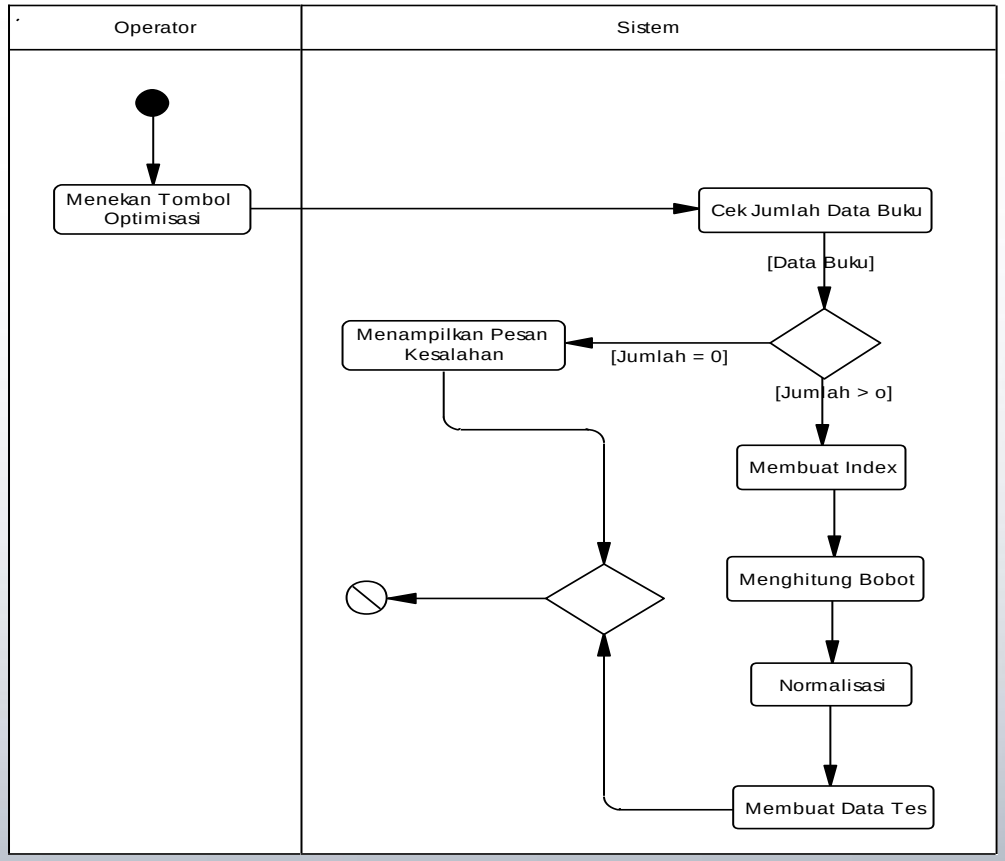
Kebutuhan Fungsional

1. **Usecase**
2. **Aktivitas Optimisasi Sistem Pencarian**
3. **Aktivitas Pencarian Menggunakan VSM**
4. **Aktivitas Pencarian Menggunakan SVMs**

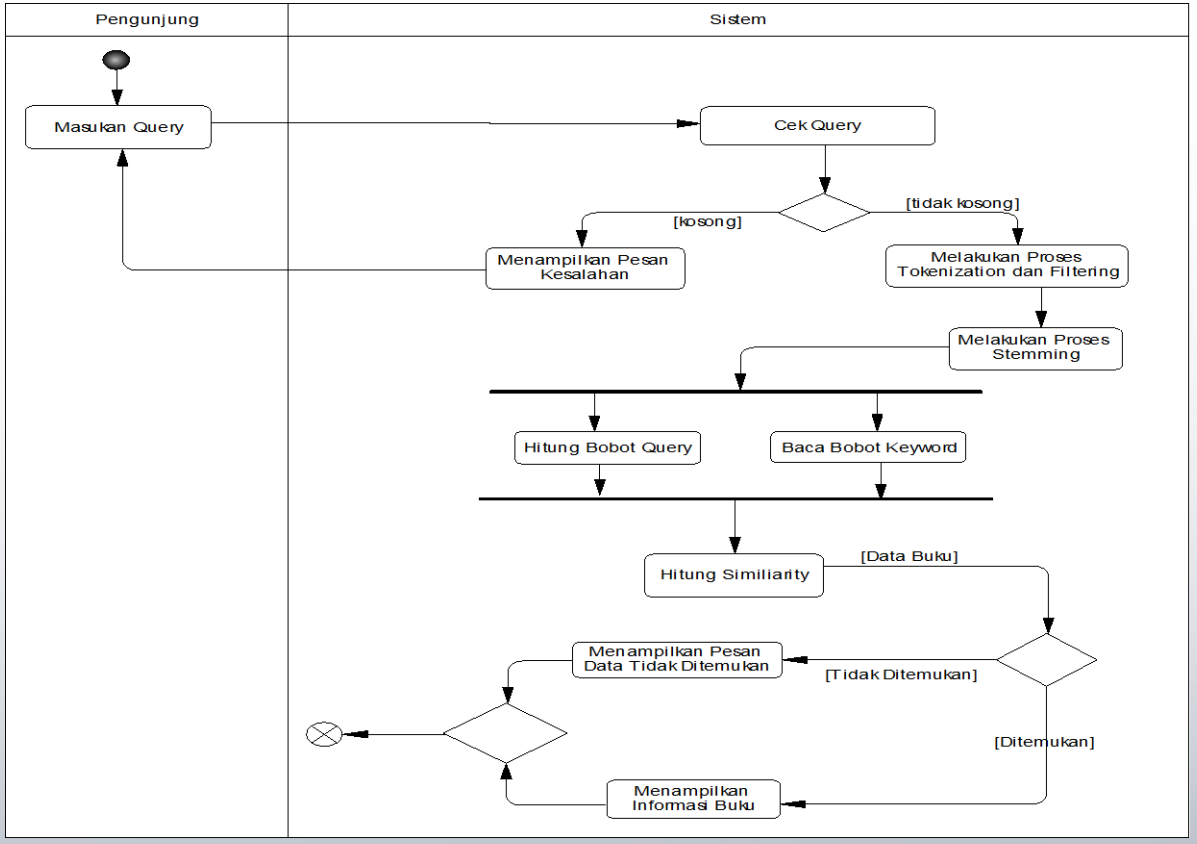
Usecase


Sistem
Pencarian Menggunakan Support Vector Machines
Penyajian Deskripsi
<<include>>
<<extend>>
Pengunjung
Pencarian Menggunakan Vector Space Model
Login
Tambah Data Buku
Ubah Data Buku
Operator
Hapus Data Buku
Tambah Data Golongan
Ubah Data Golongan
Optimisasi Sistem Pencarian
Pencarian Berdasarkan Judul Buku

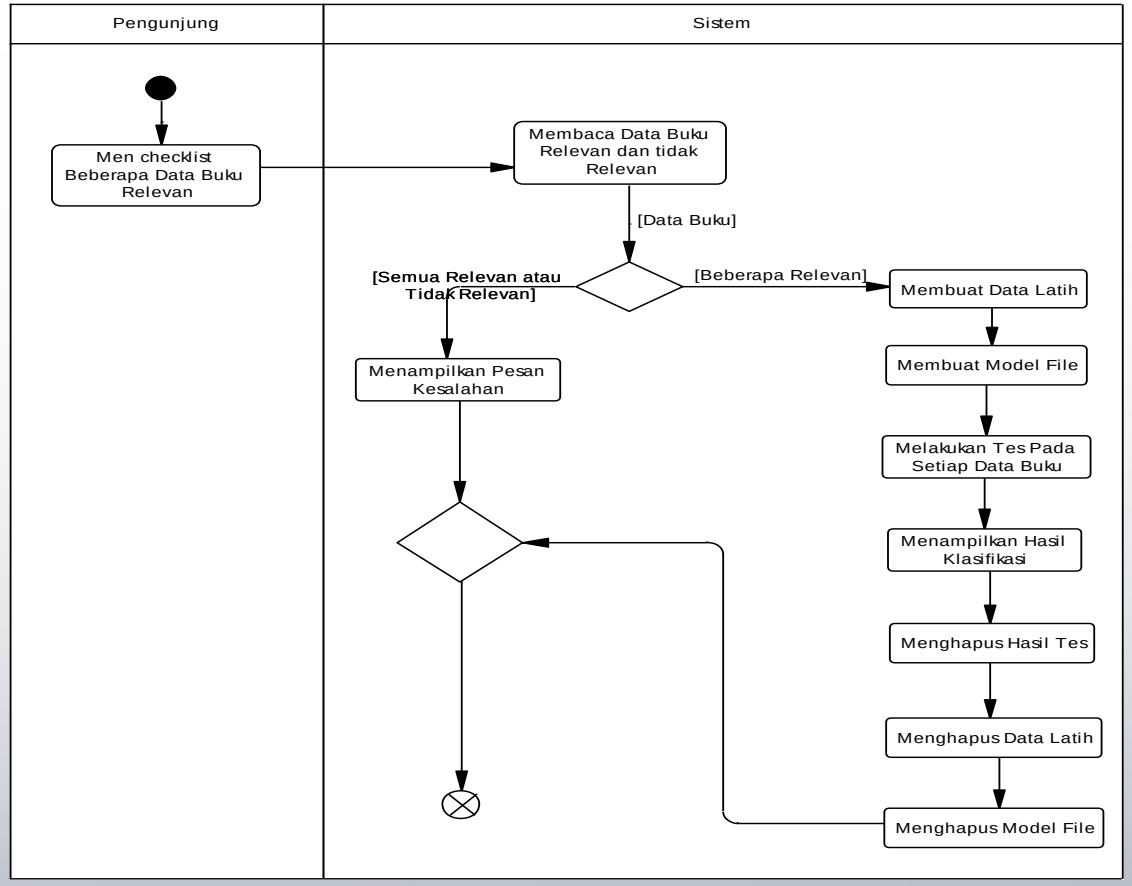
Aktivitas Optimisasi Sistem Pencarian



Aktivitas Pencarian Menggunakan VSM



Aktivitas Pencarian Menggunakan SVMs



Pengujian

- Alpha
  1. Pengujian Optimisasi Sistem Pencarian
  2. Pengujian Pencarian Menggunakan Vector Space Model
  3. Pencarian Menggunakan Support Vector Machines
- Beta

Optimisasi Sistem Pencarian

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Keyword
Yang Diharapkan	Menghitung bobot setiap kata yang terdapat di keyword, jika ada kata yang sama maka kata diwakili oleh satu nilai bobot saja kemudian nilai bobot disimpan ke database
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Bobot kata
Yang Diharapkan	Membaca nilai bobot dari database kemudian ditulis dalam file text dengan format SVM Light dan diberi nama file “[idbuku]_[judul buku].txt”
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak

Pengujian Pencarian Menggunakan Vector Space Model

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Deskripsi kebutuhan user (query)
Yang Diharapkan	Menampilkan maksimal 10 informasi buku yang memiliki tingkat ukuran kemiripan tertinggi dengan query
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak
Kasus Dan Hasil Uji (Data Salah)	
Data Masukan	Kosong atau tidak diisi
Yang Diharapkan	Menampilkan pesan kesalahan pencarian
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak

Pencarian Menggunakan Support Vector Machines

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Beberapa data buku yang di-checklist
Yang Diharapkan	Menampilkan informasi buku yang relevan terhadap data masukan dan menampilkan kembali data masukan
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak
Kasus Dan Hasil Uji (Data Salah)	
Data Masukan	Semua data buku di-checklist atau tidak di-checklist
Yang Diharapkan	Menampilkan pesan kesalahan pencarian lebih lanjut
Pengamatan	Sistem melakukan proses yang diharapkan
Kesimpulan	[√] Diterima [ ]Ditolak

Kesimpulan

1. Metode *Vector Space Model* dan *Support Vector Machines* dapat diimplementasikan dalam sistem pencarian informasi buku yang dinamis.
2. Pembangunan sistem pencarian informasi buku dapat memepermudah pengunjung untuk melakukan pencarian informasi buku berdasarkan representasi kebutuhan pengunjung.

Demo Program

Terima Kasih

1. **Nelly Indriani W, S.Si., M.T.**
2. **Adam Mukharil Bachtiar, S.Kom.**
3. **Mira Kania Sabariah, S.T., M.T.**