

Penerapan Metode Content Based Filtering Dalam Implementasi Sistem Rekomendasi Tanaman Pangan

Putri Nastiti

Program Studi Sistem Informasi
Universitas Atma Jaya Yogyakarta
putri.nastiti@uajy.ac.id

Abstrak - Panjangnya rantai distribusi merupakan salah satu permasalahan pokok di sektor pangan yang berdampak pada mahalannya harga-harga pangan. Permasalahan ini sejalan dengan prioritas pembangunan Kabupaten Sleman tahun 2017 untuk mewujudkan kemandirian ekonomi dengan menggerakkan sektor strategis ekonomi lokal termasuk diantaranya peningkatan penerapan teknologi pertanian. Hal ini berkaitan dengan bagaimana cara memperpendek rantai pemasaran sehingga dapat meningkatkan kesejahteraan petani. Dalam upaya memangkas rantai distribusi yang terlalu panjang, maka penelitian ini menawarkan solusi sebuah sistem rekomendasi yang dapat mencatat data lahan pertanian serta merekomendasikan lahan pertanian mana saja yang berpotensi menghasilkan komoditas tanaman pangan. Dengan adanya sistem ini, pemasaran hasil pertanian khususnya kepada pedagang akan lebih efektif dan efisien, artinya sistem mampu memberikan rekomendasi lahan yang sesuai untuk pengguna. Dalam pembuatannya, sistem rekomendasi ini akan menerapkan teori *Content Based Filtering*.
Kata Kunci: *content based filtering*, sistem rekomendasi, tanaman pangan.

I. PENDAHULUAN

Di negara Indonesia yang merupakan daerah agraris, masih terdapat banyak petani yang kesulitan dalam mendistribusikan hasil panennya. Seperti yang disampaikan oleh Direktur Jendral Hortikultura bahwa rantai distribusi bahan pangan yang berlangsung selama ini adalah sebagai berikut, sebagian besar petani yang berkelompok dan membentuk suatu kelompok tani biasanya menyalurkan hasil pertaniannya kepada pengepul. Terbagi menjadi beberapa jenis pengepul yaitu pengepul tingkat kecil, tingkat besar, baik di tingkat kecamatan, maupun tingkat kabupaten, selanjutnya diteruskan kepada pedagang besar [1]. Permasalahan panjangnya distribusi merupakan salah satu permasalahan pokok di sektor pangan yang berdampak pada mahalannya harga-harga pangan [2].

Hal ini juga terjadi di Propinsi Daerah Istimewa Yogyakarta, khususnya Kabupaten Sleman. Tercantum di

dalam prioritas pembangunan Kabupaten Sleman tahun 2017 poin ketujuh yaitu mewujudkan kemandirian ekonomi dengan menggerakkan sektor strategis ekonomi lokal termasuk diantaranya peningkatan penerapan teknologi pertanian. Ini juga terkait dengan bagaimana cara memperpendek rantai pemasaran sehingga dapat meningkatkan kesejahteraan petani [3]. Idealnya hasil pertanian dapat disalurkan oleh petani langsung kepada konsumen. Namun kenyataannya, hasil pertanian tidak bisa langsung sampai pada konsumen, melainkan harus melewati beberapa rantai distribusi. Hal ini menjadikan proses penyaluran hasil pertanian menjadi kurang efektif dan efisien, dari segi waktu dan juga biaya.

Dalam dua dekade terakhir, akibat terjadi revolusi dalam teknologi internet, jumlah informasi yang tersedia juga meningkat signifikan. Sistem rekomendasi adalah salah satu inovasi dalam revolusi tersebut [4]. Sistem rekomendasi menyediakan pendekatan dalam memfasilitasi keinginan dari pengguna sistem [5] dan secara rutin digunakan untuk menyarankan *item* berdasarkan preferensi pengguna di masa lalu [6]. Model statistik yang terdiri dari *Vector Space Model* dan *probabilistic model* menjadi dasar dari algoritma sistem rekomendasi untuk mengambil informasi yang relevan [7].

Dalam upaya untuk memangkas rantai distribusi yang terlalu panjang, maka solusi yang ditawarkan berupa sebuah sistem yang dapat mencatat data lahan pertanian serta merekomendasikan lahan pertanian mana saja yang berpotensi menghasilkan komoditas tanaman pangan, berupa padi khususnya untuk daerah Cangkring yang masih dalam proses pemulihan pasca bencana Merapi. Sistem tersebut harus dapat diakses dimanapun dan kapanpun, pengguna juga secara cepat dapat mengakses informasi melalui sistem tersebut.

Dengan adanya aplikasi *mobile* ini, pemasaran hasil pertanian khususnya kepada pedagang akan lebih efektif dan efisien, artinya sistem mampu memberikan rekomendasi lahan yang sesuai untuk pengguna. Dalam pembuatannya, sistem rekomendasi ini akan menerapkan teori *content based filtering*.

II. METODOLOGI PENELITIAN

A. Tinjauan Pustaka

Penelitian ini dilakukan tidak terlepas dari penelitian-penelitian yang pernah dilakukan sebelumnya, antara lain: Thorat et. al. (2015) dalam publikasi ilmiah berjudul “*Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System*” memberikan gambaran menyeluruh mengenai sistem rekomendasi yang mencakup metode *collaborative filtering*, *content-based filtering* dan pendekatan *hybrid recommender system* [8]. Dalam penelitian tersebut dikatakan bahwa untuk meningkatkan kualitas rekomendasi, beberapa pendekatan *hybrid* digunakan terutama untuk *collaborative filtering* dan *content based filtering*. Algoritma *hybrid* digunakan untuk mengintegrasikan informasi lokasi ke dalam algoritma rekomendasi yang ada.

Jovita et. al. (2015) dalam publikasi ilmiah berjudul “*Using Vector Space Model in Question Answering System*” menggunakan metode *vector space model* untuk merepresentasikan pengetahuan dan mengambil jawaban untuk setiap pertanyaan [9]. Dalam penelitian tersebut setiap *query* akan dibandingkan dengan pengetahuan berdasarkan pengukuran kesamaan keduanya. Data sampel yang digunakan untuk menguji model diambil dari dua kementerian yaitu Kementerian Pendidikan dan Kebudayaan serta Kementerian Pariwisata dan Ekonomi Kreatif. Dalam percobaan, 150 pertanyaan diberikan pada sistem. Percobaan tersebut menghasilkan 0,662 *recall* dan 0,548 *precision*.

B. Content Based Filtering

Sistem rekomendasi memanfaatkan berbagai sumber informasi untuk menyediakan pengguna suatu prediksi dan rekomendasi. Sistem rekomendasi memanfaatkan konsep *information filtering* [10]. Pada *information filtering* pengguna sudah mempunyai profil yang merepresentasikan kepentingan jangka panjang dan sistem mencoba memberikan kepada setiap pengguna *item* yang relevan. Berdasarkan pada ukuran kesamaan antara masing-masing profil, sistem memilih dan membuat peringkat pada *item* yang relevan, kemudian diberikan kepada pengguna. Terdapat dua pendekatan pada *information filtering*, yaitu *collaborative filtering* dan *content-based filtering*.

Penelitian ini akan menggunakan pendekatan *content-based filtering*. Sistem akan memilih dan melakukan peringkat *item* berdasarkan kesamaan profil pengguna dan profil *item*. Keuntungan dari pendekatan ini adalah pengguna mendapatkan wawasan tentang mengapa suatu *item* dianggap relevan untuk mereka, karena konten di setiap *item* diketahui dari representasinya. Namun pendekatan ini juga mempunyai kelemahan, misalnya kenyataan bahwa pendekatan ini berfokus pada kemiripan kata kunci. Pendekatan ini tidak mampu menangkap hubungan yang lebih kompleks pada level semantik yang lebih dalam, berdasarkan pada berbagai jenis atribut yang berhubungan dengan obyek terstruktur dari teks. Kesamaan antara representasi dari pengguna dan representasi dari *item* akan didasarkan pada prinsip kedekatan yang menyatakan bahwa jarak dari dua deskripsi

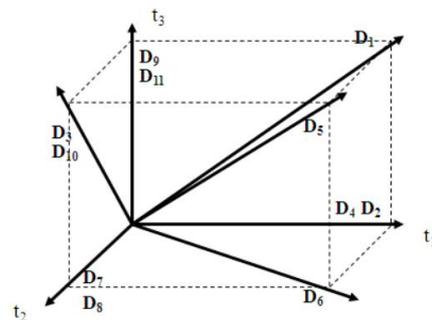
item secara langsung berkaitan dengan kesamaan mereka [11].

Minat atau preferensi pengguna juga diwakili oleh serangkaian fitur yang sama, yang disebut profil pengguna. Rekomendasi dibuat dengan membandingkan profil pengguna dengan *item* kandidat yang dinyatakan dalam set fitur yang sama. Pendekatan paling sederhana untuk rekomendasi berbasis konten adalah dengan menghitung kesamaan profil pengguna dengan setiap *item*. Terdapat beberapa cara untuk merepresentasikan agar dapat digunakan sebagai komponen pembelajaran, salah satunya adalah *Vector Space Model*.

Pada metode ini, dokumen D direpresentasikan sebagai vektor m dimensional. Dimana setiap dimensi berkorespondensi terhadap *term* yang berbeda dan m adalah total jumlah *term* yang dipakai dalam koleksi dokumen. Vektor dokumen adalah ditulis sebagai, w_i adalah bobot dari *term* t_i yang menunjukkan tingkat kepentingan. Jika pada dokumen D tidak mengandung *term* t_i maka bobot dari w_i adalah nol. Bobot *term* dapat ditentukan dengan menggunakan skema *tf-idf* [12]. Pada pendekatan ini bobot dihitung berdasarkan pada seberapa sering sebuah *term* muncul pada sebuah dokumen, dan seberapa sering ditemukan dalam koleksi dokumen.

C. Vector Space Model

Vector space model adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu *query*. Pada model ini, *query* dan dokumen dianggap sebagai vektor-vektor pada ruang n -dimensi, dimana n adalah jumlah dari seluruh *term* yang ada dalam leksikon [13]. Leksikon adalah daftar semua *term* yang ada dalam indeks. *Vector space model* digunakan karena pada metode ini memungkinkan proses pemeringkatan dokumen. Metode ini menghitung nilai cosinus dari dua vektor. Dua vektor tersebut adalah bobot dari tiap dokumen dan bobot dari *query*. Bobot dokumen dan *query* digunakan untuk proses pemeringkatan dokumen, seperti yang terlihat pada Gambar 1 berikut.



Gambar 1. *Vector Space Model*.

Pembobotan *term* dalam vektor dokumen dapat ditentukan dalam banyak cara. Pendekatan yang umum, dan digunakan dalam tugas akhir ini adalah metode *TF-IDF* [12, 13]. Pada metode ini, bobot *term* ditentukan oleh dua faktor: seberapa sering *term* j terdapat dalam dokumen i (*term*

frequency $tf_{i,j}$) dan seberapa sering muncul dalam seluruh dokumen koleksi (*document frequency* df_j). Tepatnya bobot $term_j$ pada dokumen i dirumuskan seperti berikut:

$$w_{i,j} = tf_{i,j} \times idf_j = tf_{i,j} \times \log N/df_j \quad (1)$$

N merupakan jumlah dokumen yang terdapat dalam koleksi dokumen. idf merupakan singkatan dari *inverse document frequency*. Metode ini memberikan bobot tinggi pada *term* yang sering muncul pada sedikit dokumen pada dokumen set. Setelah bobot *term* dihitung, diperlukan suatu fungsi pemeringkatan untuk mengukur kemiripan antara *query* dan dokumen vektor. Pengukuran kemiripan yang umum dikenal sebagai pengukuran kosinus. Pengukuran ini menentukan sudut antara vektor dokumen dan *query* ketika direpresentasikan dalam *V-dimensional Euclidean*, dimana v adalah ukuran *vocabulary*. Tepatnya kemiripan antara dokumen D_i dan *query* Q didefinisikan sebagai berikut:

$$sim(Q, D_i) = \frac{\sum_{j=1}^V w_{Q,j} \times w_{i,j}}{\sqrt{\sum_{j=1}^V w_{Q,j}^2 \times \sum_{j=1}^V w_{i,j}^2}} \quad (2)$$

D. Pengukuran Performansi

Cara konvensional untuk mengukur kualitas hasil yang dikembalikan oleh sebuah sistem dalam menanggapi permintaan adalah dengan menggunakan *recall* dan *precision*.

i. Recall

Recall adalah proporsi dari semua dokumen relevan yang dapat ditemukan-kembali oleh sebuah proses pencarian di sistem. Pada kasus ini adalah lahan pertanian ditemukan dalam proses pencarian. Rumusnya adalah :

$$Recall = \frac{\text{jumlah dokumen relevan yang berhasil ditemukan}}{\text{jumlah seluruh dokumen yang ditemukan}} \quad (3)$$

ii. Precision

Precision adalah proporsi jumlah dokumen yang ditemukan dan dianggap relevan untuk kebutuhan si pencari informasi.

$$Precision = \frac{\text{jumlah dokumen relevan yang berhasil ditemukan}}{\text{jumlah seluruh dokumen yang relevan}} \quad (4)$$

E. Tahapan Penelitian

1. Pengumpulan Data dan Analisis

Penulis melakukan studi pustaka untuk mengumpulkan informasi. Studi pustaka dilakukan dengan mempelajari teori-teori serta literatur yang mendukung penelitian ini terutama yang berhubungan dengan *recommender system*, *content-based filtering*, serta perangkat lunak yang digunakan untuk membangun sistem. Selain itu penulis juga mengumpulkan data seperti data lahan pertanian yang terdapat di Kabupaten Sleman yang digunakan untuk penelitian ini. Data lahan pertanian yang akan digunakan

dalam penelitian ini sebanyak 1000 data lahan pertanian yang terdiri dari informasi berikut: kelompok tani, jenis tanaman pangan, varietas, tanggal tanam, tanggal panen, lokasi (*longitude*, *latitude*), hasil panen.

2. Pengembangan sistem

Metode yang dipakai untuk mengembangkan sistem yang dipakai adalah pendekatan *content-based filtering* dengan pengukuran kemiripan (*measuring similarity*) antara profil *item* dan profil user menggunakan algoritma *cosine similarity*.

3. Evaluasi Sistem

Kualitas hasil yang dikembalikan oleh sistem dalam menjawab respon permintaan dari pengguna akan diukur menggunakan metode *recall* dan *precision*. Nilai *recall* *precision* inilah yang menentukan berhasil atau tidaknya sistem yang dibangun dengan metode *content-based filtering* ini dan implementasi *vector space model*.

III. HASIL DAN PEMBAHASAN

Sebelum dilakukan implementasi program, perlu dilakukan analisis dan desain sistem untuk mempermudah implementasi program karena sebagai acuan untuk menghasilkan program yang baik.

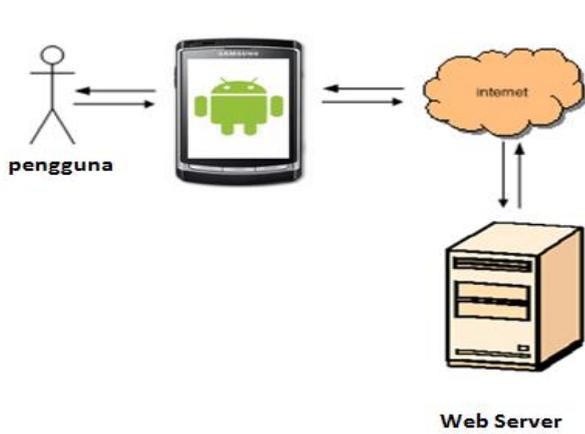
a. Gambaran Umum Sistem

Seperti yang telah disampaikan di latar belakang masalah, sistem yang akan dibuat ini digunakan untuk memberikan informasi mengenai lahan pertanian di kawasan Sleman, DIY. Sistem yang akan dibangun mempunyai sasaran pengguna yaitu masyarakat pada umumnya dan para pedagang pada khususnya yang ingin mengetahui informasi hasil pertanian berupa tanaman padi yang berada di kawasan Sleman tersebut. Informasi yang akan didapatkan oleh pengguna sistem adalah pemilik lahan, tanggal tanam, perkiraan masa panen, dan perkiraan hasil panen, letak lokasi, dan luas lahan.

Sistem mempunyai 2 level pengguna yaitu pedagang dan petani yang bertugas melakukan *update* data. Pengguna tersebut mempunyai hak untuk mendapatkan rekomendasi mengenai informasi lahan pertanian yang ditampilkan oleh sistem. Dalam pembangunan sistem rekomendasi lahan pertanian ini, penulis mengumpulkan data pertanian Kecamatan Cangkringan dari Dinas Pertanian Kabupaten Sleman.

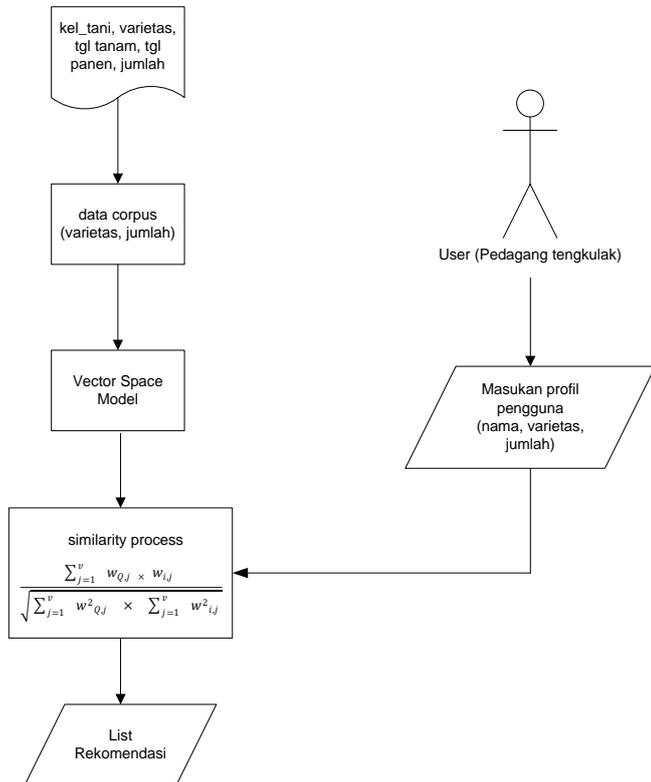
b. Arsitektur Sistem

Pada Gambar 2 terlihat bahwa *mobile* digunakan sebagai *dumb terminal*. Sedangkan penyimpanan data dan proses rekomendasi, seperti perhitungan menggunakan *vector space model* terjadi di *web server*.

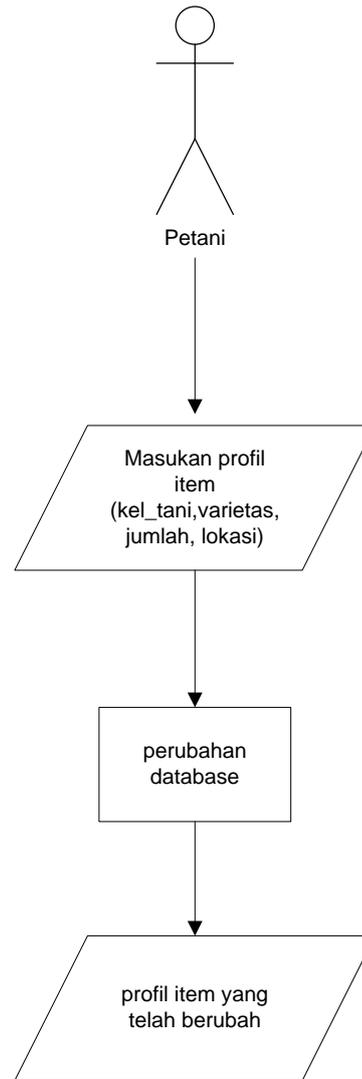


Gambar 2. Arsitektur Sistem Secara Umum.

Gambar 3 dan 4 berikut merupakan arsitektur aplikasi, atau proses yang terjadi pada *web server*:



Gambar 3. Arsitektur Sistem Untuk Pedagang.



Gambar 4. Arsitektur Sistem Untuk Petani.

Hasil rekomendasi pada penelitian ini didapatkan dengan menggunakan metode *Vector Space Model*, dengan mengukur kedekatan antara profil pengguna dan profil *item* menggunakan rumus *cosine similarity*. Pada implementasi *vector space model* pada umumnya adalah untuk menghitung kemiripan dokumen. Namun pada penelitian ini, *vector space model* tidak digunakan untuk menghitung kemiripan dokumen, tetapi menghitung kemiripan profil *item* dan profil pengguna. Berikut adalah contoh perhitungan kemiripan profil dengan algoritma *vector space model*, yang telah disesuaikan dengan kasus pada penelitian ini :

Kolom pedagang pada Tabel 1 tersebut merupakan *user query*. Pada Tabel 1 terlihat bahwa *user* (pedagang) menginginkan padi varietas impari sebesar 30 ton, situbagendit 40 ton, dan ciherang 20 ton. Tetapi untuk melakukan proses perhitungan selanjutnya angka tersebut perlu dinormalisasi. Tabel 2 adalah hasil normalisasinya.

Ide dari metode *vector space model* ini adalah menghitung nilai *cosinus* sudut dari dua vector. Pada kasus ini adalah *W* dari petani sebagai profil *item* dengan *W* dari pedagang sebagai profil *user*. Nilai *w* sudah didapat, seperti yang tercantum pada Tabel 3.

Pada tahap tersebut bobot dari *query* dikalikan dengan bobot *item* yang terdapat di setiap *record* lahan, kemudian

dijumlahkan untuk setiap lahannya. Kemudian langkah selanjutnya adalah mengkuadratkan bobot *query*. Bobot *item* di setiap lahan juga dikuadratkan. Kemudian dijumlahkan untuk tiap lahannya. Hasilnya seperti pada Tabel 4.

Langkah terakhir adalah dengan memasukkan rumus *cosine similarity*. Hasilnya seperti pada Tabel 5.

Jika dilihat dari hasil perhitungan, dapat ditarik kesimpulan bahwa yang mendekati kemiripan dengan profil pedagang adalah petani nomor 6 dengan nilai *similarity* mencapai 0,95. Dalam metode penghitungan *cosine similarity*, hasil yang paling mirip adalah yang mendekati nilai satu.

Tabel 1. Bobot Sebelum Normalisasi.

	Pedagang	Petani 1	Petani 2	Petani 3	Petani 4	Petani 5	Petani 6	Petani 7	Petani 8	Petani 9	Petani 10
Impari	30	0	80	100	0	0	50	0	0	0	100
Situbagendit	40	0	0	80	0	0	80	0	90	0	50
Ciherang	20	120	0	0	0	0	60	60	70	90	0
IR64	0	0	0	0	70	70	0	0	0	0	40
Cibodas	0	0	0	0	0	0	0	0	0	0	0
Cisadane	0	0	40	0	40	40	0	0	0	0	0
Fatmawati	0	0	0	0	0	0	0	0	0	0	0

Tabel 2. Bobot Setelah Normalisasi.

	Pedagang	Petani 1	Petani 2	Petani 3	Petani 4	Petani 5	Petani 6	Petani 7	Petani 8	Petani 9	Petani 10
Impari	0,3	0	0,8	1	0	0	0,5	0	0	0	1
Situbagendit	0,44444444	0	0	0,888889	0	0	0,888889	0	1	0	0,555556
Ciherang	0,16666667	1	0	0	0	0	0,666667	0,666667	0,777778	1	0
IR64	0	0	0	0	0	1	0	0	0	0	0,571429
Cibodas	0	0	0	0	1	0	0	0	0	0	0
Cisadane	0	0	1	0	0	1	0	0	0	0	0
Fatmawati	0	0	0	0	1	0	0	0	0	0	0

Tabel 3. Perkalian Query dan Item.

	ped*petani1	ped*petani2	ped*petani3	ped*petani4	ped*petani5	ped*petani6	ped*petani7	ped*petani8	ped*petani9	ped*petani10
Impari	0	0,24	0,3	0	0	0,15	0	0	0	0,3
Situbagendit	0	0	0,395062	0	0	0,395062	0	0,444444	0	0,246914
Ciherang	0,16666667	1	0	0	0	0,111111	0,111111	0,12963	0,166667	0
IR64	0	0	0	0	0	0	0	0	0	0
Cibodas	0	0	0	0	0	0	0	0	0	0
Cisadane	0	0	1	0	0	0	0	0	0	0
Fatmawati	0	0	0	0	0	0	0	0	0	0
SUM	0,16666667	0,24	0,695062	0	0	0,656173	0,111111	0,574074	0,166667	0,546914

Tabel 4. Kuadrat Dari Bobot.

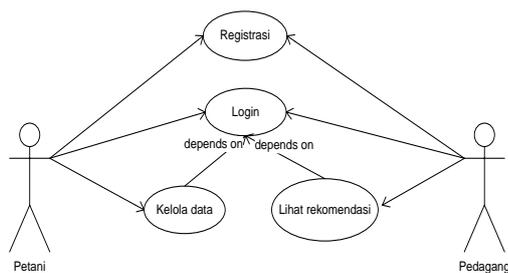
	Pedagang ²	Petani 1 ²	Petani 2 ²	Petani 3 ²	Petani 4 ²	Petani 5 ²	Petani 6 ²	Petani 7 ²	Petani 8 ²	Petani 9 ²	Petani 10 ²
Impari	0,09	0	0,64	1	0	0	0,25	0	0	0	1
Situbagendit	0,19753086	0	0	0,790123	0	0	0,790123	0	1	0	0,308642
Ciherang	0,02777778	1	0	0	0	0	0,444444	0,444444	0,604938	1	0
IR64	0	0	0	0	0	1	0	0	0	0	0,326531
Cibodas	0	0	0	0	1	0	0	0	0	0	0
Cisadane	0	0	1	0	0	1	0	0	0	0	0
Fatmawati	0	0	0	0	1	0	0	0	0	0	0
SUM	0,31530864	1	1,64	1,790123	2	2	1,484568	0,444444	1,604938	1	0,635173
AKAR	0,5615235	1	1,280625	1,337955	1,414214	1,414214	1,218428	0,666667	1,266862	1	1,278739

Tabel 5. Hasil Perhitungan.

	Petani1	Petani2	Petani3	Petani4	Petani5	Petani6	Petani7	Petani8	Petani9	Petani10
COSINE	0,29681156	0,33375	0,925154	0	0	0,95907	0,296812	0,806995	0,296812	0,761674

c. Use Case Diagram

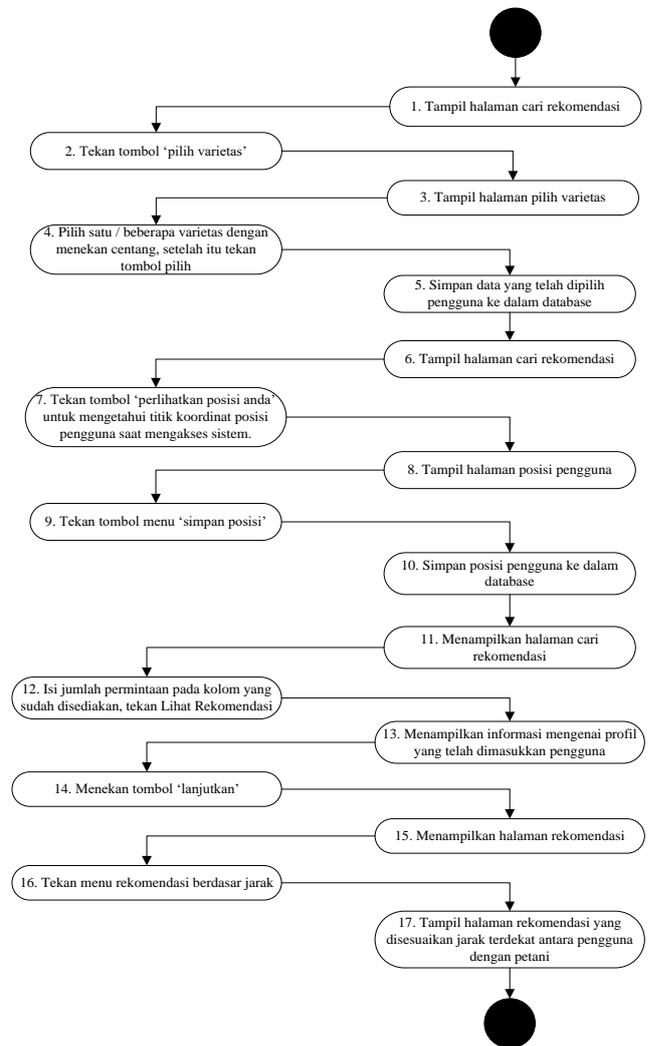
Use Case Diagram menjelaskan urutan kegiatan yang dilakukan aktor dan sistem untuk mencapai tujuan tertentu, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Use Case Diagram juga serta mempresentasikan interaksi antar aktor/pelaku dengan sistem. Gambar 5 berikut Use Case Diagram yang digunakan pada sistem rekomendasi ini.



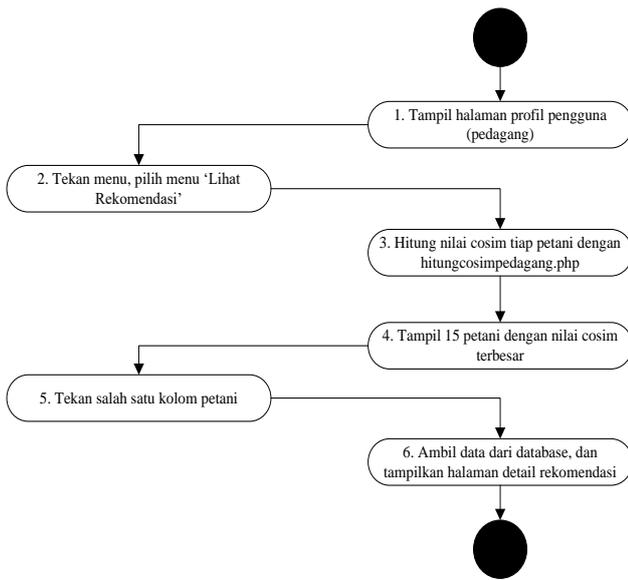
Gambar 5. Use Case Diagram Sistem.

d. Activity Diagram

Activity diagram adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas. Activity diagram menjelaskan aktivitas antara user dengan sistem. Gambar 6 dan 7 adalah activity diagram yang terdapat pada sistem ini.



Gambar 6. Diagram Aktivitas Cari Rekomendasi.



Gambar 7. Diagram Aktivitas Lihat Rekomendasi.



Gambar 9. Tampilan Halaman Hasil Rekomendasi.

e. Implementasi Use Case

1. Implementasi Use Case Lihat Rekomendasi



Gambar 8. Tampilan Halaman Lihat Rekomendasi.

Setelah berhasil *login*, maka sistem akan menampilkan data profil pedagang yang berupa varietas dan jumlah permintaan. Untuk mengetahui rekomendasi petani yang sesuai dengan profil pedagang, dapat menekan tombol ‘Lihat Rekomendasi’. Untuk menambah varietas tanaman sebagai profil, dapat menekan tombol menu pada *handphone*, dan memilih menu ‘Tambah Data’. Sedangkan untuk menu ‘Pencarian Baru’ digunakan apabila pengguna ingin melakukan pencarian rekomendasi tetapi tidak berdasarkan profil yang tersimpan dalam *database*.

Setelah menekan tombol ‘Hasil Rekomendasi’ maka sistem akan menampilkan daftar petani yang cocok dengan profil pedagang. Petani yang ditampilkan hanya yang mempunyai nilai *cosinus similarity* dengan urutan 15 terbesar.

2. Implementasi Use Case Cari Rekomendasi



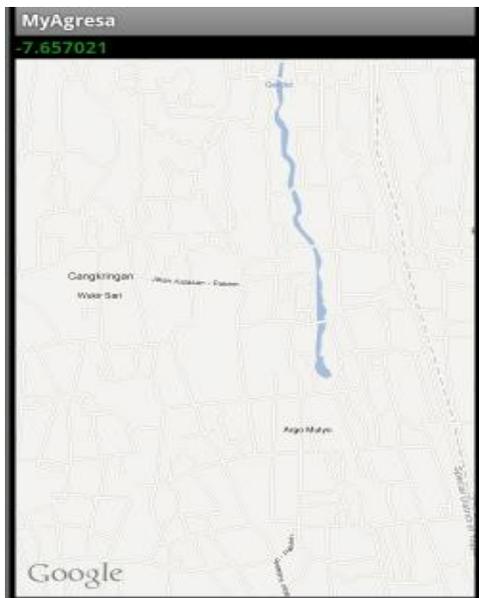
Gambar 10. Tampilan Cari Rekomendasi.

Setelah menekan tombol ‘Pencarian Baru’ maka sistem akan menampilkan halaman cari rekomendasi. Terdapat tiga hal yang harus diisi oleh pengguna, yaitu varietas, posisi sementara, dan jumlah permintaan.



Gambar 11. Tampilan Pilih Varietas

Halaman pilih varietas digunakan oleh pengguna yang terdaftar sebagai pedagang. Setelah masuk ke halaman cari rekomendasi, pengguna diminta untuk memasukkan pilihan varietas yang nantinya akan digunakan sebagai profil *item*.



Gambar 12. Tampilan Lokasi Pedagang.

Halaman lokasi pedagang digunakan oleh pengguna yang terdaftar sebagai pedagang. Setelah masuk ke halaman cari rekomendasi, pengguna diminta untuk memasukkan posisi saat mengakses aplikasi. Tekan tombol menu, dan pilih menu 'Simpan Posisi' maka longitude latitude pengguna akan tersimpan dalam database.



Gambar 13. Tampilan Menu Pilihan Rekomendasi Berdasar Jarak.



Gambar 14. Tampilan Hasil Rekomendasi Kelompok Tani.

Setelah menekan tombol 'Lanjutkan' pada halaman cari rekomendasi, lalu sistem akan menampilkan halaman hasil rekomendasi. Jika pengguna ingin mendapatkan hasil rekomendasi yang juga mempertimbangkan jarak lokasi lahan dengan posisi sementara pengguna, maka dengan menekan menu 'Rekomendasi Berdasar Jarak' sistem akan menampilkan hasil rekomendasi yang telah diurutkan berdasar jarak yang paling dekat.

f. Pengujian Kinerja Sistem

1. Pengujian Presisi Sistem untuk Rekomendasi Pedagang

Pengukuran presisi dilakukan dengan membandingkan kelompok tani yang relevan bagi pedagang, dengan kelompok tani hasil rekomendasi sistem. Terdapat total 75 sampel data produk yang digunakan sebagai *item query*, dan 10 sampel profil pedagang sebagai *item profil*. Sedangkan hasil yang dikeluarkan sistem adalah sebanyak 15 rekomendasi.

Tabel 6. Sampel Profil Kelompok Tani 1.

Varietas : Cibodas, Inpari 4, Cimelati, Situ Patenggang, Inpari 1, IR 66, Bernas prima, Cilamaya muncul, Ciherang, Cisadane, IR 36, Fatmawati, Cisantana	
Pedagang	Urutan dalam sistem :
Laurin	1
David	2
Pandu	3
Bonita	4
Christina	5
Rory	6

Tabel 7. Sampel Profil Kelompok Tani 2.

Varietas : Situ bagendit, IR 64, Cisadane, Mendawan, Intani 1, Intani 2, Bernas Prima, IR 42, Rokan, Fatmawati, Inpari 1, Konawe, Cibodas, Inpari 2, Cisantana	
Pedagang	Urutan dalam sistem :
David	1
Bonita	2
Laurin	3
Christina	4
Pandu	5
Rory	6

Tabel 8. Sampel Profil Kelompok Tani 3.

Varietas : Impari, Rokan, Cisadane, Inpari 3, Siak Raya, Ciapus, Ciherang, Hipa 8 Pioneer, Cisokan, Ciliwung, IR 66, Cibodas, IR 36, Celebes, Kalimas, Inpari 1, Inpari 2, IR 64, Fatmawati	
Pedagang	Urutan dalam sistem :
Putri	1
Pandu	2
Rory	3
Laurin	4
David	5
Christina	6
Bonita	7

Tabel 9. Sampel Profil Kelompok Tani 4.

Varietas : Ciherang, Fatmawati, Cimelati, Situ Patenggang, Inpari 1, Margasari, IR 36, Cisokan, Hipa 8 Pioneer, IR 42, Impari, IR 64, Gilingsing, Inpari 2, Cibodas	
Pedagang	Urutan dalam sistem :
Putri	1
Laurin	2
Christina	3
Pandu	4
David	5
Bonita	6

Tabel 10. Sampel Profil Kelompok Tani 5.

Varietas : Cibodas, Gilingsing, Ciapus, Ciherang, Fatmawati, Inpari 2, IR 64, Cisadane, Situ bagendit, Cisantana, Batang Gadis, Rokan, IR 36	
Pedagang	Urutan dalam sistem :
David	1
Pandu	2
Rory	3
Christina	4
Laurin	5
Bonita	6

Tabel 11. Revelansi Hasil Pencarian Rekomendasi Pedagang.

	Profil 1	Profil 2	Profil 3	Profil 4	Profil 5
Pdg 1	R	R	R	R	R
Pdg 2	R	R	R	R	R
Pdg 3	R	R	R	R	R
Pdg 4	R	R	R	R	R
Pdg 5	R	N	R	R	R
Pdg 6	R	R	R	R	N
Pdg 7	N	R	R	N	R

Keterangan :

Pdg : 7 sampel profil pedagang

Profil : 5 sampel profil kelompok tani yang diujikan

Tabel 12. Hasil Perhitungan *Precision* Rekomendasi Pedagang.

No	Relevan	Not Relevan	Hasil Rekomendasi	Precision
Profil 1	6	1	7	0,85
Profil 2	6	1	7	0,85
Profil 3	7	0	7	1,00
Profil 4	6	1	7	0,85
Profil 5	6	1	7	0,85
Jumlah				4,40
Rata-rata				0,88
%				88,00%

Keterangan :

Nilai *precision* didapatkan dengan rumus :

$|\text{relevant} \cap \text{retrieved}| / |\text{retrieved}|$. Dapat dilihat hasil presisi untuk profil kelompok tani 1 sampai 5. Kemudian nilai keseluruhan presisi dirata-rata dan didapatkan hasil rata-rata presisi yaitu 88.00%.

IV. KESIMPULAN

Pembangunan sistem ini menggunakan algoritma *vector space model*. Terdapat 1000 lahan yang digunakan sebagai data sampel dalam sistem. Dari proses implementasi kesimpulan bahwa pedagang mendapatkan informasi mengenai petani mana saja yang memiliki profil yang sama dengannya. Petani mendapatkan informasi mengenai pedagang mana saja yang memiliki profil yang sama dengannya.

Pengujian presisi dilakukan dengan membandingkan kelompok tani yang relevan bagi pedagang, yaitu kelompok tani yang dipilih oleh pedagang dengan kelompok tani hasil rekomendasi sistem.

Berdasarkan pengujian terhadap 10 profil pedagang dengan 15 rekomendasi kelompok tani teratas, didapatkan hasil presisi rata-rata sebesar 78.40%.

Sistem akan lebih baik apabila pengujian sistem dilakukan tidak hanya menggunakan *precision* tetapi juga menggunakan *recall*.

REFERENSI

- [1] Rachman, F. (2017). *Begini Distribusi Pangan dari Petani, Pengepul, Sampai Konsumen*. Diakses dari: <https://finance.detik.com/berita-ekonomi-bisnis/d-3415106/begini-distribusi-pangan-dari-petani-pengepul-sampai-konsumen>
- [2] Juli, P. (2017). Penataan Sektor Pangan. *Majalah Info Singkat Ekonomi dan Kebijakan Publik*, Vol. IX, No. 07. Pusat Penelitian Badan Keahlian DPR RI.
- [3] Balai Penyuluhan Pertanian, Perikanan, Kehutanan BP3K (2017). *UPT BP3K Wilayah II Gelar Potensi Pertanian*. Diakses dari: <http://www.slemankab.go.id/9043/upt-bp3k-wilayah-ii-gelar-potensi-pertanian.slm>
- [4] Deshpande, M.G., Muddebihalkar, A.V., Jadhav, A.B. & Kokate, S. (2016). Hybrid Content-Based Filtering Recommendation Algorithm on Hadoop. *International Journal of Advanced Research in Computer Engineering & Technology*, Vol. 5.
- [5] Kalita, J., Balas, V.E., Borah, S., & Pradhan, R. (Eds.). (2019). *Recent Developments in Machine Learning and Data Analytics*, Vol. 740. Singapore: Springer Singapore. Diakses dari: <https://doi.org/10.1007/978-981-13-1280-9>
- [6] Achakulvisut, T., Acuna, D.E., Ruangrong, T., & Kording, K. (2016). Science Concierge: A fast content-based recommendation system for scientific publications. *PLoS ONE*, 11(7). Diakses dari: <https://doi.org/10.1371/journal.pone.0158423>
- [7] Singh, J.N. & Dwivedi, S.K. (2012). Analysis of Vector Space Model in Information Retrieval. *International Journal of Computer Applications*, 14-18.
- [8] B.Thorat, P., Goudar, R.M. & Barve, S. (2015). Survey on Collaborative Filtering and Content-Based Recommending. *International Journal of Computer Application*, 110(4), 31–36. Diakses dari: <https://doi.org/10.5120/19308-0760>
- [9] Jovita, Linda, Hartawan, A., & Suhartono, D. (2015). Using Vector Space Model in Question Answering System. In *Procedia Computer Science*, Vol. 59, pp. 305–311. Elsevier. Diakses dari: <https://doi.org/10.1016/j.procs.2015.07.570>
- [10] Bobadilla, J., Ortega, F., Hernando, A. & Gutiérrez, A. (2013). Recommender Systems Survey. *Knowledge-Based Systems*, 46, 109–132. Diakses dari <https://doi.org/10.1016/j.knosys.2013.03.012>
- [11] Lu, Z., Dou, Z., Lian, J., Xie, X. & Yang, Q. (2015). Content-Based Collaborative Filtering For News Topic Recommendation. In *AAAI 2015: Proceedings of the Twenty-ninth AAAI Conference on Artificial Intelligence*, pp. 217–223. Diakses dari: <https://doi.org/10.1080/714965461>
- [12] Jing, L., Ng, M.K. & Huang, J.Z. (2010). Knowledge-based vector space model for text clustering. *Knowledge and Information Systems*, 25(1), 35–55. Diakses dari: <https://doi.org/10.1007/s10115-009-0256-5>
- [13] Gadge, J., Sane, S. & Kekre, H.B. (2015). Performance Analysis of Layered Vector Space Model in Web Information Retrieval. *International Journal of Applied Information Systems*, 8(5), 7–15. Diakses dari: <https://doi.org/10.5120/ijais15-451320>