

Aplikasi Chatbot pada Sistem Informasi Penyewaan Scaffolding Menggunakan Metode TF-IDF

Dimas Wahyu Wibowo¹, Habibie Ed Dien², Trianta Almira Ramadhani³

^{1,2,3}Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang
¹dimas.w@polinema.ac.id, ²habibie@polinema.ac.id, ³tara.trianmadhani@gmail.com

Abstrak— Perkembangan teknologi saat ini semakin memudahkan pengguna dalam mengakses aplikasi yang ada. Salah satunya dengan menerapkan Sistem Informasi Berbasis Website. Berdasarkan hasil penelitian sebelumnya yang menerapkan *Chatbot* dengan metode N-gram, TF-IDF, dan Cosine Similarity, terbukti Sistem *Chatbot* pada aplikasi tersebut dapat memudahkan masyarakat dalam mencari informasi

Oleh sebab itu, CV. Scaffolding Samarinda membutuhkan pemberian layanan sistem informasi yang cepat di setiap saat. Berdasarkan pengujian yang telah dilakukan, maka dapat disimpulkan bahwa Sistem ini dapat melakukan *Tokenizing* (memisah kata penyusun dari suatu dokumen) dan menghitung bobot *TF-IDF* dan *Cosine Similarity* untuk mencari jawaban pada Sistem. Sehingga saat pelanggan menuliskan pertanyaan, akan muncul jawaban sesuai harapan *user*. Sistem juga dapat memudahkan pelanggan dalam menerima informasi sesuai dengan yang diharapkan saat itu juga.

Kata kunci— Chatbot, TF-IDF, Cosine Similarity, Scaffolding

I. PENDAHULUAN

Perkembangan teknologi saat ini semakin memudahkan pengguna dalam mengakses aplikasi yang ada. Penggunaan aplikasi menggunakan desktop sudah dikalahkan dengan luasnya penggunaan internet yang bisa digunakan di berbagai platform dan bisa diakses oleh pengguna dari kalangan manapun, dan menjadikan website dapat memberikan kemudahan bagi masyarakat dalam mengaksesnya. Salah satunya dengan menerapkan Sistem Informasi Berbasis Website.[1]

Berdasarkan hasil penelitian sebelumnya terkait penelitian yang dilakukan oleh (Kavitha, Cethana, 2019) yang menerapkan *chatbot* untuk membantu masyarakat jika ingin mengetahui penyakit yang mungkin di derita dengan menanyakannya melalui aplikasi android dan akan diproses oleh sistem pada web dan akan menampilkan jawaban sesuai harapan dengan metode N-gram, TF-IDF, dan Cosine Similarity. Sistem *Chatbot* pada aplikasi tersebut menggantikan peran Dokter untuk membantu dokter untuk mengurangi biaya perawatan dan menghemat waktu. Sehingga pengguna aplikasi tidak perlu ke dokter atau ke

spesialis untuk menanyakannya.[9] Penelitian lain yang saya gunakan milik (Dhebys, Eka, 2017) yang menggunakan *chatbot* untuk memudahkan masyarakat dalam mencari objek wisata di daerah Jawa Timur. *Chatbot* pada aplikasi tersebut memudahkan objek wisata memberikan informasi kepada pengunjung tanpa harus menyebarkan pamflet, brosur, dan poster.[2]

CV. Scaffolding Samarinda merupakan usaha penyewaan scaffolding di Samarinda dan sekitarnya. Salah satu permasalahan yang dimiliki oleh CV. Tersebut adalah keterbatasan waktu jam kerja operator merespon pertanyaan dari pelanggan yang membutuhkan informasi mengenai penyewaan scaffolding. Dikarenakan operator memiliki jam kerja yang ditentukan oleh peraturan perusahaan. Sehingga pelanggan sulit berkomunikasi dengan Operator diluar jam kerja operator. Sedangkan pelanggan membutuhkan respon yang cepat untuk menangani kendala pada perusahaannya yang membutuhkan jasa dari CV. Scaffolding Samarinda.

TF-IDF (*Term Frequency - Inverse Document Frequency*) merupakan metode algoritma yang menentukan frekuensi dari kemunculan sebuah term dalam dokumen yang bersangkutan. Metode ini untuk menghitung nilai *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) pada setiap kata di setiap dokumen. Semakin besar jumlah kemunculan suatu term (TF tinggi) dalam dokumen, semakin besar pula bobotnya atau akan memberikan nilai kesesuaian yang semakin besar. (informatikalogi.com).

Sehingga pada penelitian ini, penulis membuat Aplikasi *Chatbot* Pada Sistem Informasi Scaffolding Menggunakan Metode TF-IDF yang diharapkan dapat memudahkan serta dapat mengatasi permasalahan yang telah dipaparkan di atas. Aplikasi *chatbot* ini diharapkan dapat menjawab pertanyaan dari calon pelanggan dengan informasi/penjelasan yang mudah dipahami.

Apakah penerapan *chatbot* dengan metode N-gram, TF-IDF, dan Cosine Similarity dapat meringankan beban kerja Operator CV. Scaffolding Samarinda dalam melayani memberikan informasi kepada Pelanggan setiap saat dalam keterbatasan waktu jam kerja perusahaan, maka pada penelitian ini, penulis membuat Aplikasi *Chatbot* pada Sistem Informasi Scaffolding dengan Menggunakan Metode TF-IDF yang diharapkan dapat memudahkan serta dapat mengatasi permasalahan yang telah dipaparkan di atas.

Dengan aplikasi chatbot ini diharapkan dapat menjawab pertanyaan dari calon pelanggan dengan informasi/penjelasan yang mudah dipahami, dan terlaksana dalam memberikan informasi pada saat itu juga. [3]

Tujuannya untuk meringankan beban kerja Operator CV. Scaffolding Samarinda dalam melayani pemberian informasi kepada Pelanggan setiap saat dalam keterbatasan waktu jam kerja perusahaan, Batasan masalahnya yaitu tentang:

- Informasi mengenai barang yang tersedia
- Informasi mengenai harga sewa barang per unit
- Informasi mengenai jangka waktu peminjaman barang
- Informasi mengenai berat barang per unit

II. TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama dengan judul penelitian penulis. Namun, penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Dari beberapa judul penelitian yang penulis baca, penulis memilih penelitian yang dilakukan oleh Kavitha B. R. dan Dr. Chetana R. Murthy yang berjudul "Chatbot for healthcare system using Artificial Intelligence" sebagai referensi utama penulis dalam melakukan penelitian. Hal ini dikarenakan adanya persamaan metode yang digunakan oleh penulis, yaitu TF-IDF dan *Cosine Similarity*, sebagai metode utama dalam pembuatan aplikasi *chatbot*.

B. Chatbot

Chatbot adalah sebuah simulator percakapan yang berupa program komputer yang dapat berdialog dengan penggunanya dalam bahasa alami.[4] Karena chatbot hanya sebuah program, dan bukan robot (chatbot tidak memiliki tubuh dan tidak memiliki mulut sehingga tidak dapat berbicara seperti manusia), maka yang dimaksud dengan dialog antar manusia sebagai pengguna dengan chatbot dilakukan dengan cara mengetik apa yang akan dibicarakan dan chatbot akan memberikan respon. Orang membuat dan mengembangkan program chatbot disebut bot master. [5]

C. XAMPP

XAMPP adalah pengembangan PHP di lingkungan paling populer. XAMPP merupakan distribusi Apache yang benar-benar gratis dan mudah dipasang yang berisi MariaDB, PHP, dan Perl. Paket open source XAMPP telah diatur agar sangat mudah untuk diinstal dan digunakan.

D. MySQL

Menurut Kustiyahningsih (2011:145), "MySQL adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dari setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel".

E. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang dapat digunakan untuk tujuan umum. PHP lebih populer digunakan untuk pengembangan aplikasi web. Dalam proses pembuatan halaman web, PHP tidak memerlukan kode yang panjang seperti pada Perl dan Python karena kode PHP dapat disisipkan di dalam kode HTML. PHP dapat dijalankan dalam sebagian besar sistem operasi, termasuk linux, varian-varian UNIX (HP-UX, OpenBSD), Windows, dan Mac OS X. PHP juga mendukung sebagian besar server web yang ada saat ini, seperti: Apache, IIS, nginx, dan lighttpd. Bahasa yang digunakan dalam CodeIgniter menggunakan PHP dimana bahasa pemrograman PHP digunakan sebagai bahasa pemrograman umum.

F. JavaScript

JavaScript adalah bahasa script berdasar pada objek yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen HTML. Dimana objek tersebut dapat berupa suatu window, frame, URL, dokumen, form, button, atau item yang lain. Yang semuanya itu mempunyai properti yang saling berhubungan dengannya, dan masing-masing memiliki nama, lokasi, warna nilai, dan atribut lain. JavaScript digunakan untuk pembuatan Desain Website dan Desain Chatbot yang dibuat.

G. TF-IDF

Metode Term Frequency Inverse Document Frequency (TF-IDF) adalah cara pemberian bobot hubungan suatu kata (term) terhadap dokumen.

TF murni (raw TF), nilai TF diberikan berdasarkan jumlah kemunculan suatu term di dokumen

$$tf \quad (1)$$

Inverse Document Frequency (IDF) merupakan sebuah perhitungan dari bagaimana term di distribusikan secara luas pada koleksi IDF dokumen yang bersangkutan. IDF menunjukkan hubungan ketersediaan sebuah term dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung term mengandung term yang dimaksud, maka nilai IDF semakin besar.

$$IDF_j = \log(D / df_j) \quad (2)$$

Jenis formula TF yang biasa digunakan untuk perhitungan adalah TF murni (raw TF). Dengan demikian rumus umum untuk Term Weighting TFIDF adalah penggabungan dari formula

perhitungan raw TF dengan formula IDF dengan cara mengalikan nilai TF dengan nilai IDF:

$$w_{ij} = tf_{ij} \times \log(D/df_j) \quad (3)$$

H. Cosine Similarity

Cosine similarity merupakan rumus yang digunakan Untuk menghitung kesamaan atau *similarity* dengan menentukan sudut antara vektor dokumen dengan vektor *query* dalam dimensi V pada bidang *Euclidean*. Hasil dari *cosinesimilarity* memiliki nilai antara 0 sampai dengan 1. Nilai 0 merupakan nilai yang didapat apabila dokumen tidak berhubungan dengan *query*, sedangkan nilai 1 berarti dokumen memiliki keterhubungan tinggi dengan *query* (Lahitani, Permanasari dan Setiawan, 2016). *Cosine Similarity* digunakan untuk mengambil jawaban yang ditanya oleh customer. Setelah TF-IDF selesai dihitung, bobot tertinggi akan dimasukkan ke dalam rumus *Cosine Similarity*. [6]

$$\cos(d_j, q_k) = \frac{\sum_{i=1}^n (td_{ij} \times tq_{ik})}{\sqrt{\sum_{i=1}^n td_{ij}^2 \times \sum_{i=1}^n tq_{ik}^2}} \quad (4)$$

Ket: $\cos(d_j, q_k)$ = Tingkat kesamaan dokumen dengan *query* tertentu

td_{ij} = Term ke- i dalam vektor untuk dokumen ke- j

tq_{ik} = Term ke- i dalam vektor untuk *query* ke- k

n = Jumlah term yang unik dalam data set

I. Recall and Precision

Menurut Kurniawan (2010) *Recall* adalah perbandingan jumlah dokumen relevan yang terambil sesuai dengan *query* yang diberikan dengan total kumpulan dokumen yang relevan dengan *query*. *Precision* adalah perbandingan jumlah dokumen yang relevan terhadap *query* dengan jumlah dokumen yang terambil dari hasil pencarian. *Precision* dapat diartikan sebagai ketepatan atau kecocokan (antara permintaan informasi dengan jawaban terhadap permintaan itu). Sedangkan istilah *recall* dibidang sistem temu kembali informasi (*information retrieval*) berkaitan dengan kemampuan menemukan kembali informasi yang sudah tersimpan (Pendit 2008).

Rumus penilaian *precision* yang digunakan dalam penelitian ini adalah :

$$Precision = \frac{\text{Jumlah dokumen relevan yang ditemukan}}{\text{Jumlah semua dokumen yang ditemukan}} \times 100\%$$

Sedangkan nilai *relatif Recall* dihitung dengan rumus :

$$Recall = \frac{\text{Jumlah dokumen relevan yang ditemukan}}{\text{Jumlah semua dokumen relevan di dalam koleksi}} \times 100\%$$

Kedua ukuran diatas biasanya diberi nilai dalam bentuk persentase, 1 sampai 100%. Sebuah sistem informasi akan dianggap baik jika tingkat *recall* maupun

precision-nya tinggi. Jika seseorang mencari dokumen tentang “Perpustakaan” dan sistem tersebut memiliki 100 buku tentang perpustakaan maka kinerja yang paling baik adalah jika sistem tersebut berhasil menemukan 100 dokumen tentang perpustakaan.

Jika sistem tersebut memberikan 100 temuan, dan ditemukan tersebut ada 50 dokumen tentang perpustakaan, maka nilai *recall*-nya adalah 0,5 (atau 50%) dan nilai *precision*-nya juga 0,5. Kalau sistem tersebut memberikan 1 dokumen saja dan dokumen tersebut adalah “perpustakaan” maka *recall*-nya bernilai 0,01 dan *precision*-nya 1. Nilai *precision*-nya yang tinggi sebenarnya terjadi karena sistem memberikan 1 jawaban kepada pencari informasi. Kalau sistem memberikan 100 dokumen dan hanya 1 yang relevan maka nilai *recall*-nya tetap 0,01 tetapi *precision*-nya merosot ke 0,01 (Pendit 2008). Metode ini digunakan untuk menghitung kecocokan jawaban dari sistem kepada user. [8]

III. METODELOGI PENELITIAN

A. Studi Literatur

Studi literatur adalah pencarian referensi dari berbagai sumber yang memiliki hubungan dengan studi kasus yang ditemukan dan mencari solusi dari permasalahan yang ada. Referensi berisi tentang :

- Chatbot
- TF-IDF
- Cosine Similarity
- Scaffolding

Referensi yang dibutuhkan bisa di dapat dari jurnal, artikel laporan penelitian, dan situs-situs internet. Output dari studi literatur adalah terkumpulnya referensi yang relevan dengan studi kasus yang ada.

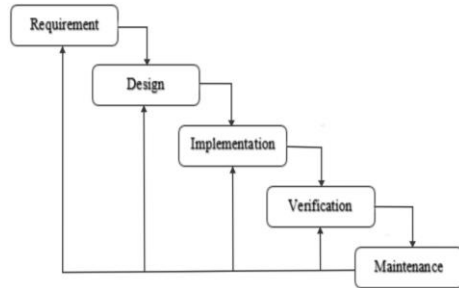
B. Identifikasi Masalah

Masalah yang dihadapi adalah sulitnya customer menghubungi admin karena keterbatasan waktu sehingga dengan adanya aplikasi diharapkan customer lebih cepat mendapat respon jawaban sesuai dengan jam bertanya.

C. Metode Pengumpulan Data

Metode yang digunakan dalam pengambilan data adalah melalui wawancara. Wawancara dilakukan menggunakan media *Whatsapp* dengan admin penyewaan scaffolding. Wawancara dilakukan pada bulan November 2019 hingga bulan maret. Hasil wawancara yang dilakukan adalah didapatkan data barang yang disewakan, harga barang, dan screenshot chat admin dengan customer, dan informasi-informasi lain terkait dengan operasional penyewaan.

D. Metode Pengembangan Sistem (Waterfall)



Gambar 3.1 Waterfall Model

E. Metode Pengolahan Data

Pada penelitian ini, dilakukan pengolahan data layanan penyewaan, data barang beserta harga yang diperoleh dari wawancara dengan admin penyewaan. Data-data tersebut akan diolah menjadi data set jawaban chatbot yang akan dirancang. Kemudian data set yang telah diolah akan masuk ke dalam tahapan *preprocessing*. Tahapan *preprocessing* adalah sebagai berikut:

1. Case Folding
2. Tokenizing
3. Filtering
4. Stemming

F. Perhitungan Manual

TABEL 3.1 CONTOH RESPON CHATBOT

Kode	Jawaban/Respon
Q	Harga sewa scaffolding
d1	Penyewaan buka pada pukul 8 pagi
d2	Harga sewamain frame 1,9 Rp55.000 untuk sebulan
d3	Harga sewa main frame 1,9 Rp45.000 untuk 2 minggu
d4	Harga sewa main frame 1,9 Rp35.000 untuk 1 minggu
d5	Hari Jumat dan Minggu Libur.

Untuk	0	0	1	0	0	0	1	6
bulan	0	0	0	1	1	1	3	2
Minggu	0	0	0	0	0	1	1	6
Hari	0	0	0	0	0	1	1	6
Jumat	0	0	0	0	0	1	1	6
Libur	1	0	1	1	1	0	4	1,5

TABEL 3.3 PERHITUNGAN TF-IDF

Dokumen	Q	d1	d2	d3	d4	d5
Harga	0,1761	0	0,1761	0,1761	0,1761	0
Sewa	0	0	0	0	0	0
Scaffolding	0,7782	0	0	0	0	0
Buka	0	0,7782	0	0	0	0
Pukul	0	0,7782	0	0	0	0
Pagi	0	0,7782	0	0	0	0
Main frame	0	0	0,3010	0,3010	0,3010	0
Untuk	0	0	0,7782	0	0	0
Bulan	0	0	0	0,3010	0,3010	0,3010
Minggu	0	0	0	0	0	0,7782
Hari	0	0	0	0	0	0,7782
Jumat	0	0	0	0	0	0,7782
Libur	0,1761	0	0,1761	0,1761	0,1761	0

Kemudian dihitung hasil perkalian skalar masing-masing query jawaban terhadap query key jawaban. Hasil perkalian dari setiap jawaban dengan query dijumlahkan. Proses perkalian skalar ini dapat dilihat dalam tabel 3.6.

Tabel 3.4Perkalian Skalar

Dokumen	d1	d2	d3	d4	d5
Harga	0	0,031008	0,031008	0,031008	0
Sewa	0,006270	0,006270	0,006270	0,006270	0
Scaffolding	0	0	0	0	0
Buka	0	0	0	0	0
Pukul	0	0	0	0	0
Pagi	0	0	0	0	0
Main frame	0	0	0	0	0
Untuk	0	0	0	0	0
Bulan	0	0	0	0	0
Minggu	0	0	0	0	0
Hari	0	0	0	0	0
Jumat	0	0	0	0	0

Proses TF-IDF yang berdasarkan data pada tabel 3.1 sebagai contoh perhitungan TF menggunakan proses *preprocessing*. Dalam melakukan perhitungan digunakan beberapa dokumen untuk mempermudah perhitungan TF-IDF. Hasil proses TF-IDF termuat pada tabel 3.2 dan tabel 3.3.

TABEL 3.2 PERHITUNGAN DF

Dokumen	q	d1	d2	d3	d4	d5	df	log(d/df)
Harga	1	0	1	1	1	0	4	1,5
Sewa	1	1	1	1	1	0	5	1
Scaffolding	1	0	0	0	0	0	1	6
Buka	0	1	0	0	0	0	1	6
Pukul	0	1	0	0	0	0	1	6
Pagi	0	1	0	0	0	0	1	6
Main frame	0	0	1	1	1	0	3	2

Libur	0,006270	0,037278	0,037278	0,037278	0
Jumlah	0	0,031008	0,031008	0,031008	0

Langkah selanjutnya yaitu melakukan perkalian vektor tiap query key jawaban dengan query jawaban. Hasil perkalian vektor ini dapat dilihat dalam tabel 3.7.

TABEL 3.5 PERKALIAN VEKTOR

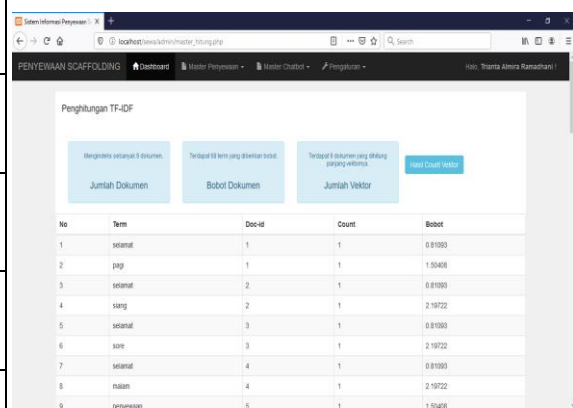
Dokumen	Q	d1	d2	d3	d4	d5
Harga	0,031008	0	0,031008	0,031008	0,031008	0
Sewa	0,00627	0,00627	0,00627	0,00627	0,00627	0
Scaffolding	0,605519	0	0	0	0	0
Buka	0	0,605519	0	0	0	0
Pukul	0	0,605519	0	0	0	0
Pagi	0	0,605519	0	0	0	0
Main frame	0	0	0,090619	0,090619	0,090619	0
Untuk	0	0	0,605519	0	0	0
Bulan	0	0	0	0,090619	0,090619	0,090619
Minggu	0	0	0	0	0	0,605519
Hari	0	0	0	0	0	0,605519
Jumat	0	0	0	0	0	0,605519
Libur	0,642797	1,822828	0,733416	0,218516	0,218516	1,907177
Jumlah	0,031008	0	0,031008	0,031008	0,031008	0
Jumlah Akar	0,006270	0,006270	0,006270	0,006270	0,006270	0

Kode	Dokumen	Nilai Similarity	Persentase
d1	Penyewaan buka pada pukul 8 pagi	0,005792	0,58%
d2	Harga sewa main frame 1,9 Rp55.000 untuk sebulan	0,032241	3,22%
d3	Harga sewa main frame 1,9 Rp45.000 untuk 2 minggu	0,093118	9,31%
d4	Harga sewa main frame 1,9 Rp35.000 untuk 1 minggu	0,170595	17,06%
d5	Hari Jumat dan Minggu Libur.	0	0%

Berdasarkan hasil perhitungan *Cosine Similarity* pada tabel di atas, maka bisa disimpulkan jika respon pada dokumen ke-2 merupakan respon yang paling mendekati dari pertanyaan yang diajukan oleh user.

IV. HASIL

Data yang digunakan dalam penelitian ini adalah data harga layanan percetakan yang sudah diubah ke dalam bentuk dokumen jawaban chatbot. Sebelum diimplementasikan ke dalam algoritma *cosine similarity*, data dokumen jawaban tersebut akan masuk ke dalam tahap *preprocessing*. Setelah dokumen tersebut melalui tahap *preprocessing* maka akan masuk ke dalam tahap pembobotan TF-IDF. Penghitungan bobot TF-IDF dilakukan untuk memberi bobot tiap *term* yang terdapat pada kalimat yang dicari oleh user. Penghitungan bobot TF-IDF menggunakan rumus (3) dan menghasilkan pembobotan sebagai berikut:



No	Term	Doc-id	Count	Bobot
1	sekolah	1	1	0,81083
2	pagi	1	1	1,90459
3	sekolah	2	1	0,81083
4	sekolah	2	1	2,19722
5	sekolah	3	1	0,81083
6	sekolah	3	1	2,19722
7	sekolah	4	1	0,81083
8	sekolah	4	1	2,19722
9	sekolah	5	1	1,90459

GAMBAR 6.1 CONTOH PERHITUNGAN BOBOT TF-IDF

Pada gambar 6.1 merupakan contoh hasil penghitungan TF-IDF yang telah dilakukan oleh system. Setelah dilakukan pembobotan kata, maka akan dilakukan penghitungan menggunakan *Cosine Similarity* untuk membandingkan tingkat kemiripan antar dokumen dengan query yang dimasukkan oleh user. Penghitungan *Cosine Similarity* menggunakan rumus sesuai dengan rumus (4). Sebagai contoh user memasukkan input “harga sewa scaffolding” ke dalam chatbot. Hasil penghitungan cosine similarity

Langkah terakhir adalah menghitung nilai *Cosine Similarity* dengan rumus seperti yang tercantum dalam rumus 2.10.1. Hasil perhitungan *Cosine Similarity* dapat dilihat dalam tabel 3.6.

Tabel 3.6 Hasil Perhitungan Similarity

menghasilkan 3 dokumen jawaban yang memiliki tingkat kemiripan tertinggi diantara dokumen lainnya. Tingkat kemiripan 3 dokumen tersebut dapat dilihat pada table 6.1.

TABEL 6.1 HASIL PENGHITUNGAN COSINE SIMILARITY

NO.	QUERY USER	DOKUMEN	NILAI SIMILARITY
1	HARGA SEWA SCAFFOLDING	HARGA SEWA 1 SET MAIN FRAME 1,7 SELAMA 1 MINGGU 30.000 RUPIAH, SELAMA 2 MINGGU 40.000 RUPIAH, SELAMA 1 BULAN 50.000 RUPIAH	0.192847
2		PENYEWAAN BUKA PUKUL 8 PAGI	0.365148
3		Penyewaan buka pada hari Senin, Selasa, Rabu, Kamis, dan Sabtu	0.374634

Berdasarkan hasil pada table 6.1 maka didapati pada dokumen yang ke 3 merupakan dokumen dengan tingkat similaritas tertinggi terhadap query yang diinputkan oleh user. Dokumen tersebut kemudian dikirim sebagai respon chatbot atas kalimat yang telah diinputkan oleh user.

V. PEMBAHASAN

Berdasarkan dari hasil pengujian data-data penelitian yang telah dilakukan oleh peneliti, berikut paparan pembahasan hasil penelitian:

Penggunaan metode TF-IDF dan *cosine similarity* dapat diterapkan ke dalam sistem *chatbot* untuk merespon permintaan informasi dari *user*.

Perhitungan TF-IDF dapat digunakan untuk membojotkan setiap *term* yang dicari oleh *user* dalam koleksi dokumen jawaban.

Metode *cosine similarity* dapat digunakan untuk melakukan perhitungan tingkat similaritas antar dokumen terhadap *query* yang dicari oleh *user*.

Penggunaan metode *cosine similarity* kurang cocok diterapkan pada *chatbot* penjualan online karena tidak dapat merespon permintaan pesanan dari *user*.

VI. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pembahasan yang telah dilakukan pada bab I hingga bab VI, maka dapat disimpulkan bahwa Sistem ini dapat Melakukan Tokenizing (Memisah kata penyusun dari suatu dokumen) dan menghitung bobot TF-IDF setiap kata tersebut pada tiap array. Kemudian menggunakan Metode TF-IDF dan Cosine Similarity untuk mencari jawaban pada sistem. Sehingga saat user menuliskan pertanyaan, akan muncul jawaban sesuai

harapan pelanggan. Sistem juga dapat memudahkan user dalam menerima informasi sesuai dengan yang diharapkan. Sehingga tugas Operator menjadi ringan karena sudah tidak lagi harus selalu membuka *WhatsApp* untuk melayani para pelanggan yang membutuhkan informasi tentang penyewaan scaffolding.

B. Saran

Untuk pengembangan sistem informasi Aplikasi Chatbot pada Sistem Informasi Penyewaan Scaffolding Menggunakan metode TF-IDF lebih lanjut agar semakin memberikan manfaat untuk user ada beberapa hal yang bisa dijadikan bahan kajian lebih lanjut, yaitu

Untuk peneliti selanjutnya disarankan *chatbot* dapat dikembangkan lebih lanjut.

Daftar Pustaka

- [1] D. Suryani dan E. Larasati, "Aplikasi Chatbot Objek Wisata Jawa Timur Berbasis AIML," *SMARTICS Journal*, pp. 47-54, 2017.
- [2] D. Suryani dan Y. Putera, "Aplikasi Chatbot Berbasis Web Pada Sistem Informasi Layanan Publik Kesehatan di Malang dengan Menggunakan Metode TF-IDF," *Jurnal Informatika Polinema*, pp. 224-228, 2018.
- [3] D. Suryani dan I. Aulia, "Penerapan Metode TF-IDF dan N-Gram pada Pengembangan Aplikasi Chatbot Berbasis LINE untuk Layanan Publik Kesehatan di Kota Malang," *Jurnal Informatika Polinema*, pp. 7-11, 2018.
- [4] M. Astiningrum dan M. Shoburu Rohmah, "Implementasi NLP dengan Konversi Kata pada Sistem Chatbot Konsultasi Laktasi," *Jurnal Informatika Polinema*, pp. 46-52, 2018.
- [5] H. Yamaguchi dan M. Mozgoyov, "A Chatbot Based on AIML Rules Extracted From Twitter Dialogues," *Communication Paper Poznan*, pp. 37-42, 2018.
- [6] R. Melita dan T. Dirjam, "Penerapan Metode Term Frequency Inverse Document Frequency dan Cosine Similarity pada Sistem Temu Kembali Informasi untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Syarah Umdatil Ahkam)," *Jurnal Teknik Informatika*, pp. 149-164, 2018.
- [7] Y. Dwi Listio, A. Zulkarnain dan A. Tirtana, "Pembuatan Sistem Pencarian Pekerjaan Menggunakan TF-IDF," *Jurnal Ilmiah Teknologi Informasi Asia*, pp. 91-100, 2019.
- [8] I. N. Satria Paliwahet dan I. K. Gede Darma Putra, "Pencarian Informasi Wisata Daerah Bali Menggunakan Teknologi Chatbot," *Lontar Komputer*, pp. 144-153, 2017.
- [9] Kavitha B. R.. dan C. R. Murthy "Chatbot for healthcare system using Artificial Intelligence," *International Journal of Advance Research, Ideas and Innovations in Technology*, vol.5 pp. 1304-1307, June, 2019.