

SISTEM PENCARIAN LONTAR BERBASIS WEB DENGAN METODE VECTOR SPACE MODEL PADA DINAS KEBUDAYAAN PROVINSI BALI

I Kadek Yuda Setiadi¹, Made Sudarma², Duman Care Khrisne³

Program Studi Teknik Elektro Fakultas Teknik Universitas Udayana – Bali

Email: yudhasetiadi2012@gmail.com¹, imasudarma@gmail.com², duman@unud.ac.id³

Abstrak

Penelitian ini bertujuan untuk membantu dalam pencarian gambar digitalisasi lontar dengan Information Retrieval Sistem yang dibangun menggunakan metode Vector Space Model. Pengujian sistem pencarian menghasilkan sistem pencarian lontar yang mendapatkan nilai recall: 75.4% dan precision: 100% berdasarkan grafik analisis receiver operating characteristic (ROC). Pengujian dengan System Usability Scale (SUS) yang diuji di Dinas Kebudayaan Provinsi Bali mendapat nilai tertinggi pada pertanyaan poin 1, 5 dan 7 yang mencapai 42.

Kata kunci: Lontar, ROC, SUS, vector space model

Abstract

This study aims to assist in the search for lontar images with Information Retrieval System built using the Vector Space Model method. The search system testing resulted in a lontar search system that received recall values: 75.4% and precision: 100% based on the graph of the receiver operating characteristic (ROC) analysis. Testing with System Usability Scale (SUS) tested at the Bali Provincial Culture Office got the highest score on statement point 1, 5 and 7 which reached 42.

Key Words: Lontar, ROC, SUS, vector space model

1. PENDAHULUAN

Lontar adalah daun dari pohon tal yang dikeringkan, lontar merupakan media untuk menulis karya sastra. Pelestarian budaya Bali berupa lontar saat ini menjadi perhatian banyak kalangan mulai dari pemerintah dan masyarakat Bali khususnya. Pelestarian ini dapat dilakukan dengan digitalisasi lontar [1].

Digitalisasi lontar telah dilakukan oleh Dinas Kebudayaan Pemerintah Provinsi Bali sejak tahun 2010 hingga pertengahan 2012, jumlah lontar yang telah melalui proses digitalisasi sebanyak tujuh persen dari jumlah lontar yang disimpan sekitar 6000 lontar. Banyaknya gambar lontar yang disimpan menimbulkan permasalahan baru dalam pencarian lontar yaitu, menyebabkan proses pencarian gambar lontar semakin lama. Hal ini tidak dapat dihindari, karena di Dinas

Kebudayaan Pemerintah Provinsi Bali belum tersedia sistem pencarian gambar lontar. Tidak tersedianya sistem pencarian gambar lontar menyebabkan pustakawan dan pelajar mengalami kesulitan dalam mencari dan mempelajari lontar [2].

Vector Space Model (VSM) adalah metode yang paling sederhana dalam sistem temu kembali informasi, *Vector Space Model* merupakan model *Information Retrieval* yang mempresentasikan dokumen dan query dalam vektor dimensional. Konsep dasar dari VSM adalah menghitung jarak antar dokumen kemudian mengurutkan berdasarkan tingkat kedekatannya.

Pada penelitian ini dibangun sebuah *Information Retrieval System* yang diterapkan dalam sebuah sistem penyimpanan arsip lontar di Dinas Kebudayaan Provinsi Bali. Metode

VSM digunakan dalam pembobotan pada setiap dokumen alih bahasa lontar yang yang dimasukkan ke dalam sistem dan diterapkan dalam proses pencarian, yaitu untuk mengurutkan hasil pencarian berdasarkan tingkat relevansi informasi dengan query. Aplikasi pencarian ini akan menampilkan lontar-lontar yang terdapat di Dinas Kebudayaan Pemerintah Provinsi Bali dalam bentuk citra digital dan menampilkan alih bahasa lontar dalam bentuk text.

2. KAJIAN PUSTAKA

2.1 Lontar

Kata lontar memiliki kaitan erat terhadap bahan yang menjadi dasar dalam pembuatannya, yaitu daun tal (sejenis daun palma/borassus flabelliformis). Lontar adalah salah satu budaya yang ada di Bali, lontar telah diakui dan menjadi warisan budaya dunia. Goresan artistik yang terdapat di atas daun lontar adalah aksara Bali. Aksara Bali merupakan bahasa yang telah mengambil fungsi dan peran sebagai lambang identitas masyarakat Bali. Aksara Bali adalah sarana atau wahana untuk mengungkapkan mengenai kebudayaan yang ada di Bali [3].

2.2 Vector Space Model

Vector Space Model (VSM) adalah metode untuk mengamati tingkat kecocokan atau kesamaan term dengan cara pembobotan term. Pada Vector Space Model, sebuah istilah dipresentasikan kedalam sebuah dimensi ruang vector. Relevansi sebuah dokumen ke sebuah query didasarkan pada similaritas diantara vektor dokumen dan vektor query biasa [4].

2.3 TF.IDF

Frekuensi kemunculan term pada suatu dokumen disebut tf.idf. Jenis formula yang digunakan untuk perhitungan *term frekuensi* (tf) yaitu tf murni (*raw tf*). Maka rumus umum untuk tf.idf adalah penggabungan dari formula perhitungan raw tf dengan formula perhitungan idf. tf.idf diperoleh dengan mengalikan term frequency dan inverse document frequency [5].

$$w_{ij} = tf_{ij} \times idf_j \quad (1)$$

$$w_{ij} = tf_{ij} \times \log(D/df_j) \quad (2)$$

Keterangan :

w_{ij} : Bobot *term* t_j terhadap dokumen d_i

tf_{ij} : Jumlah kemunculan t_j dalam d_i

D : Jumlah dokumen di dalam *database*

df_j : Jumlah dokumen yang mengandung *term* t_j

2.4 Similarity Measure

Setelah bobot dokumen diketahui, maka dilakukan proses perangkingan dokumen berdasarkan tingkat relevansi dokumen dengan query. Penentuan relevansi dokumen dengan query dipandang sebagai pengukur kesamaan (similarity measure) antara vektor dokumen dengan vektor query. Semakin mirip suatu vektor dokumen dengan vektor query maka dokumen dapat dipandang semakin relevan dengan query [6].

$$sim(Q, D_i) = \frac{\sum_{j=1}^t w_{qj} \cdot w_{ij}}{\sqrt{\sum_{j=1}^t (w_{ij})^2 \times \sum_{j=1}^t (w_{qj})^2}} \quad (3)$$

2.5 Recall and Precision

Evaluasi dalam *Information Retrieval System* dipengaruhi oleh dua parameter, yaitu *recall* (R) dan *precision* (P). *Recall* adalah rasio antara dokumen relevan yang berhasil ditemukembali dari seluruh dokumen relevan yang ada dalam sistem. *Precision* adalah rasio dokumen relevan yang berhasil ditemukembali dari seluruh dokumen yang berhasil ditemukembali [7].

$$P = \frac{\text{kata yang dipisahkan dengan benar}}{\text{jumlah yang dipisahkan}} \quad (4)$$

$$R = \frac{\text{kata yang dipisahkan dengan benar}}{\text{jumlah kata sebenarnya}} \quad (5)$$

2.6 Usability Testing

Sistem usability scale (SUS) merupakan salah satu standar dalam pengukuran usability suatu aplikasi dengan menggunakan sepuluh pernyataan tentang usability. Sepuluh pernyataan disusun dengan pernyataan ganjil bernilai positif, dan pernyataan genap bernilai negatif. Skala yang digunakan dari 1 sampai 5 untuk mengekspresikan respon pengujian terhadap setiap pernyataan [8].

3. METODE PENELITIAN

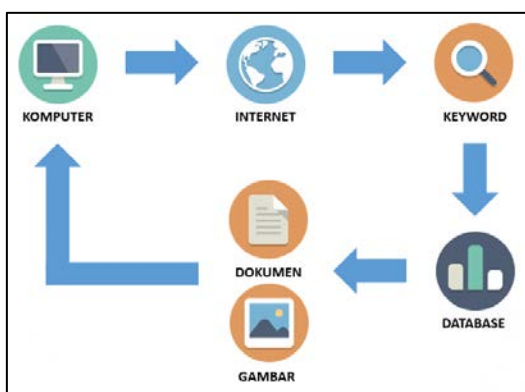
Data yang digunakan dalam penelitian ini adalah lontar dan alih bahasa lontar yang diperoleh dari Dinas Kebudayaan

Provinsi Bali. Data lontar dan alih bahasa lontar yang digunakan dalam penelitian adalah sebagai berikut.

- Bhuwana Mahbah
- Yama Purwana Tatwa
- Yama Purana Tatwa
- Yama Purwa Tatwa
- Sila Kramaning Aguron-guron

3.1 Gambaran Umum

Gambar 1 merupakan cara kerja dari aplikasi. User membuka halaman aplikasi dengan komputer yang terhubung ke Internet. Selanjutnya user mengetikkan keyword dari lontar yang ingin dicari pada halaman aplikasi. Keyword dicocokkan dengan data yang ada di dalam database. Data yang cocok dengan keyword, dikirim dari database ke komputer pengguna. Data yang dikirim berupa gambar dan dokumen tentang lontar.



Gambar 1. Gambaran Umum Sistem.

3.2 Skenario Perhitungan

Contoh perhitungan bobot dokumen terhadap query yang dimasukan oleh pengguna, dengan menggunakan metode pembobotan $tf.idf$ dengan tiga dokumen yang akan dihitung tingkat kesamaannya dengan query yang dimasukan adalah sebagai berikut

D1: "Yama Purwa Tattwa"

D2: "Yama Purana"

D3: "Tattwa Jnana"

Pengguna memasukan query: "Yama Yama Tattwa"

Total jumlah dokumen dalam koleksi (D) = 3. Untuk setiap query dan dokumen dalam koleksi, dilakukan pemotongan string berdasarkan tiap kata yang menyusunnya, menghilangkan tanda baca, dan mengubahnya menjadi huruf kecil.

yama : $\log(3/2) = 0.176$

: $df = 0.176$

: $df+1 = 1.176$
 purwa : $\log(3/1) = 0.477$
 : $df = 0.477$
 : $df+1 = 1.477$
 tattwa : $\log(3/2) = 0.176$
 : $df = 0.176$
 : $df+1 = 1.176$
 purana : $\log(3/1) = 0.477$
 : $df = 0.477$
 : $df+1 = 1.477$
 jnana : $\log(3/1) = 0.477$
 : $df = 0.477$
 : $df+1 = 1.477$

Metode $tf.idf$ digunakan untuk mencari representasi nilai dari tiap dokumen dalam koleksi. Semakin besar nilai perhitungan bobot yang diperoleh maka tingkat relevansi dokumen terhadap query akan semakin tinggi. Berikut adalah contoh perhitungan bobot term query "yama" dalam dokumen 1 (D1). Jumlah kemunculan term "yama" dalam dokumen 1 (D1) adalah sebanyak satu kali ($tf = 1$), total dokumen dalam koleksi ada sebanyak tiga dokumen ($D=3$), dan total dokumen yang mengandung term "yama" adalah dua dokumen ($df=2$). Sehingga diperoleh nilai bobot term "yama" pada dokumen 1 (D1):

$$w_{ij} = 1 \times (\log(3/2) + 1)$$

$$w_{ij} = 1 \times (0.176 + 1)$$

$$w_{ij} = 1.176$$

Nilai term frequency (tf), inverse document frequency (idf), dan $idf + 1$ untuk setiap term dalam masing-masing dokumen, yang dapat di lihat pada Tabel 1.

Tabel 1. Susunan nilai tf dan idf

	Yama	Purwa	Tattwa	Purana	Jnana
D1	1.176	1.477	1.176	0	0
D2	1.176	0	0	1.477	0
D3	0	0	1.176	0	1.477
df	2	1	2	1	1
Idf	0.176	0.477	0.176	0.477	0.477
Idf +1	1.176	1.477	1.176	1.477	1.477

Dengan demikian dapat diperoleh nilai bobot (w) untuk setiap term pada query dalam masing-masing dokumen, yang dapat di lihat pada Tabel 2.

	Yama	Purwa	Tattwa	Purana	Jnana
D1	1.176	1.477	1.176	0	0
D2	1.176	0	0	1.477	0
D3	0	0	1.176	0	1.477

Tabel 2. Susunan nilai bobot w

Ketika pengguna memasukan query berupa “Yama yama tattwa” dalam sistem pencarian lontar, sistem pencarian lontar akan menghitung vector tf.idf untuk query dan menghitung nilai setiap dokumen yang berhubungan dengan query yang dimasukkan. Nilai bobot query dapat di lihat pada Tabel 3 Ketika menghitung nilai tf.idf untuk query term dibagi frekuensi dengan frekuensi maksimum (2) dan kalikan dengan nilai-nilai idf.

Tabel 3. Nilai bobot query

q	1.176	0	0.588	0	0
---	-------	---	-------	---	---

Hitung panjang setiap dokumen dengan query. Menggunakan persamaan 3 fungsi similarity.

$$D1 = \sqrt{1.176^2 + 1.477^2 + 1.176^2} = 2.223$$

$$D2 = \sqrt{1.176^2 + 1.477^2} = 1.887$$

$$D3 = \sqrt{1.176^2 + 1.477^2} = 1.887$$

$$q = \sqrt{1.176^2 + 0.588^2} = 1.314$$

Selanjutnya diitung tingkat kecocokan setiap dokumen dengan query.

$$\cosSim(D1.q) = \frac{(1.176 * 1.176 + 1.477 * 0 + 1.176 * 1.088 + 0 * 0 + 0 * 0)}{(2.223 * 1.314)} = \frac{2.661}{2.921} = 0.91$$

$$\cosSim(D2.q) = \frac{(1.176 * 1.176 + 0 * 0 + 0 * 0)}{(1.887 * 1.314)} = \frac{1.382}{2.479} = 0.557$$

$$\cosSim(D3.q) = \frac{(0 * 1.176 + 0 * 0 + 1.176 * 1.088 + 0 * 0 + 1.477 * 0)}{(1.887 * 1.314)} = \frac{1.279}{2.479} = 0.515$$

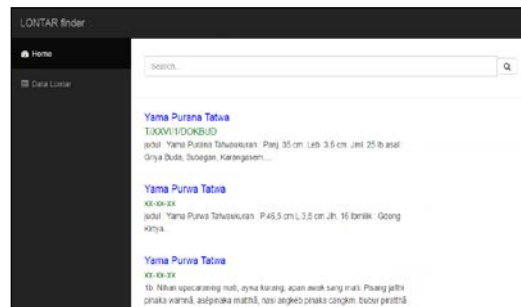
Menurut tingkat kecocokan dokumen dengan query. Urutan dokumen yang ditampilkan menjadi: D1, D2, D3.

4. HASIL DAN PEMBAHASAN

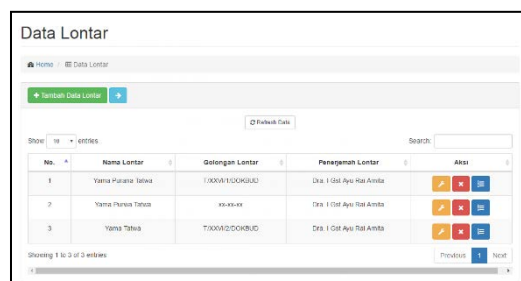
4.1 Hasil Perancangan

Hasil perancangan membahas mengenai keluaran atau hasil akhir dari pembangunan sistem pencarian lontar. Gambar 2 merupakan tampilan halaman pencarian lontar, dalam tampilan pencarian lontar terdapat dua menu utama. Pada menu Lontar terdapat hasil pencarian data

lontar, sedangkan menu data lontar mengarah ke halaman data lontar yang berfungsi untuk penambahan data lontar, pengubahan data lontar dan penghapusan data lontar. Halaman data lontar bisa dilihat pada gambar 3.



Gambar 2. Halaman pencarian lontar



Gambar 3. Tampilan halaman data lontar

4.2 Pembahasan

Evaluasi sistem dilakukan dengan metode recall dan precision. Pengujian dilakukan dengan kata yang digunakan sebagai bahan uji seperti yang tertera pada Tabel 4. Pengujian dilakukan dari K1 hingga K15, sesuai dengan kata yang digunakan sebagai bahan uji pada Tabel 4.

Tabel 4. Pengujian Ukuran Marker

No. Pengujian	Kata Kunci (query)	Halaman Relevan
K1	Pinak	6
K2	Sari	9
K3	Maulam	6
K4	Krama	5
K5	Gading	5
K6	Sida Tinambanan	1
K7	Nama Swaha	8
K8	brata mangan	2
K9	Sakeng Sariranta	2
K10	Prabhu Pralaya	1

K11	Tingkah Nusang Sawa	3
K12	Sasab Mararjna Amati	1
K13	Reh Suba Kamma	1
K14	Nista Duhur Dulange	1
K15	Carik Windu Modre	2

Perhitungan untuk evaluasi sistem dengan metode recall dan precision. Pengujian dilakukan dengan kata yang digunakan sebagai bahan uji seperti yang tertera pada Tabel 4. Pengujian dilakukan sebanyak 5 kali untuk setiap query pengujian. Pengujian dilakukan dengan perhitungan *recall* dan *precision*.

Tabel 5 merupakan pengujian recall dan precision satu kata. Berdasarkan pencarian dengan satu kata, tidak terlihat perbedaan antara recall dan precision dalam pengujian K1 hingga K5. Dari pengujian pada Tabel 5 dapat di lihat bahwa nilai rata-rata pencarian satu kata yang didapat adalah sebagai berikut.

Precision : 100%
Recall : 100%

Tabel 5. Pengujian 1 kata

No. Pengujian	Precision (%)	Recall (%)
K1	100	100
K2	100	100
K3	100	100
K4	100	100
K5	100	100

Tabel 6 merupakan pengujian recall dan precision dua kata. Dalam pencarian dengan dua kata, hasil pengujian tabel dua kata terlihat adanya penurunan nilai recall dengan pengujian satu kata. Dari pengujian pada Tabel 6 dapat di lihat bahwa nilai rata-rata pencarian dua kata yang didapat adalah sebagai berikut.

Precision : 100%
Recall : 59.2%

Tabel 6. Pengujian 2 kata

No. Pengujian	Precision (%)	Recall (%)
K6	100	62.5
K7	100	59
K8	100	49.5

K9	100	50
K10	100	75

Tabel 7 merupakan pengujian *recall* dan *precision* tiga kata. Dalam pencarian dengan tiga kata, hasil pengujian tabel tiga kata terlihat adanya perbedaan dengan pengujian satu kata dan pengujian dua kata tentang nilai *recall*. Dari pengujian pada Tabel 7 dapat dilihat bahwa nilai rata-rata pencarian tiga kata yang didapat adalah sebagai berikut.

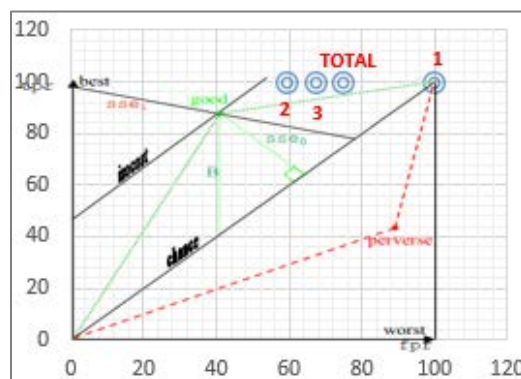
Precision : 100%
Recall : 67.2%

Tabel 7. Pengujian 3 kata

No. Pengujian	Precision (%)	Recall (%)
K11	100	33.6
K12	100	72.2
K13	100	83.3
K14	100	71.4
K15	100	75.5

Dari data pengujian yang didapat pada Tabel 5, 6 dan 7 telah dihasilkan nilai pengujian *recall* dan *precision*. Dari pengujian tersebut didapat nilai rata-rata total sebagai berikut.

Precision : 100%
Recall : 75.4%



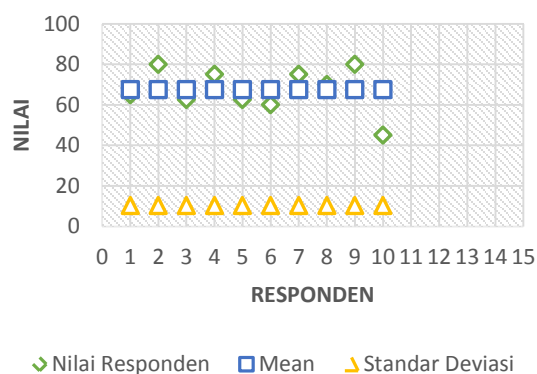
Gambar 4. Grafik analisis ROC perbandingan nilai *recall*(x) dan *precision*(y)

Dari Gambar 4 terlihat nilai pengujian *recall* dan *precision* dalam grafik analisis ROC, pengujian satu kata mendapat hasil good seperti yang terlihat pada grafik analisis ROC dikarenakan nilai recall dan precision yang mencapai 100%. Untuk

pengujian dua dan tiga kata juga mendapat hasil good seperti yang terlihat pada grafik analisis ROC. Dari pengujian tersebut didapat nilai rata-rata total hasil pencarian berdasarkan grafik analisis ROC adalah good.

Pengujian kepada responden dilakukan di Dinas Kebudayaan Provinsi Bali dengan pegawai dan pengunjung sebagai responden untuk diminta kesediaan menggunakan sistem pencarian lontar. Responden merupakan pegawai dan pengunjung Dinas Kebudayaan Provinsi Bali, masing-masing berjumlah 5 orang. Responden diminta menggunakan Laptop dengan system pencarian lontar yang telah terdapat di dalamnya, lembar skenario, dan lembar angket *System Usability Scale* (SUS).

Gambar 5 merupakan grafik hasil responden Lontar Finder yang diambil. Nilai tertinggi adalah pada responden 2 dan 9 dengan nilai 80 dan nilai terendah mencapai 45 pada responden 10. Rata-rata hasil yang didapat dari semua partisipan adalah 67.5 dengan standar deviasi 10.2 dari hasil rata-rata tersebut dapat diketahui nilai dari aplikasi terdapat pada kualitas bagus lalu dengan standar deviasi 10.2 diketahui rentang jarak rata-rata maksimum dan minimum adalah 10.2.



Gambar 5. Grafik hasil responden

Sistem mendapatkan nilai tertinggi pada pertanyaan poin 1, 5 dan 7 yang mencapai 42. Dan nilai terendah pada pertanyaan poin 4 dan 10 yang mencapai 31. Hal ini mengindikasikan bahwa untuk penggunaan aplikasi diperlukan bantuan teknisi yang untuk menjelaskan penggunaan aplikasi.

5. KESIMPULAN

Sistem temu kembali informasi yaitu Lontar Finder dapat mencari informasi mengenai judul maupun alih bahasa lontar.

Dalam proses pengindeksan sistem temu kembali informasi yaitu Lontar Finder melalui beberapa tahapan pemrosesan kata, yang diawali dengan *parsing*, yaitu suatu pemrosesan kata yang di dalamnya terdapat proses untuk mengubah semua huruf dalam dokumen menjadi huruf kecil, selanjutnya dilakukan proses penghilangan tanda baca. Dilanjutkan dengan pemisahan teks menjadi *array* kata dan diakhiri dengan pemberian bobot ke setiap *array* kata.

Dalam proses pencarian pada sistem temu kembali informasi yaitu Lontar Finder, proses yang dilalui sama dengan proses pengindeksan akan tetapi dibagian terakhir ditambahkan fungsi *similarity* yang bertujuan untuk mencocokkan *query* yang dimasukkan dengan data yang terdapat dalam *database*.

Sistem temu kembali informasi Lontar Finder memperoleh hasil yang baik dengan rata-rata *recall* dan *precision* mencapai 100%. Sehingga dokumen yang dicari relevan dengan *query* yang dimasukkan.

Pengujian *usability* system temu kembali informasi Lontar Finder setelah diuji kepada 10 pegawai Dinas Kebudayaan Provinsi Bali mendapatkan nilai rata-rata 67.5, dan mendapat mendapatkan nilai tertinggi pada pertanyaan poin 1, 5 dan 7 yang mencapai 42.

6. SARAN

Algoritma stemming digunakan untuk pengecekan kata dasar, kata berimbuhan, stop word, dan frasa. Frasa adalah gabungan dua kata atau lebih yang bersifat non-predikatif (tidak ada yang berkedudukan sebagai predikat). Saat ini algoritma stemming untuk kata-kata dalam Bahasa Bali belum dikembangkan. Untuk membuat aplikasi pencarian lontar menjadi lebih akurat diperlukan adanya algoritma stemming untuk kata-kata dalam Bahasa Bali, sehingga dapat mengurangi ukuran indeks dan menambah akurasi dalam pencarian.

7. DAFTAR PUSTAKA

- [1] Rai Putra. Lontar Bali Manuskrip Penampang Peradaban Berkarakter. Seminar Nasional Potensi Naskah Lontar Bali yang Bernilai Luhur Dalam Penguatan Jati Diri Bangsa. Bali. 2015.

- [2] Dinas Kebudayaan Provinsi Bali, 2017.
- [3] Alit Sancana. Kajian Efektivitas Teknik dan Bahan Konservasi pada Lontar di Bali. *Jurnal Konservasi Cagar Budaya Borobudur*. 2014.
- [4] Aziz, A., Saptono, R. and P.S, K., Implementasi Vector Space Model dalam Pembangkitan Frequently Asked Questions dan Solusi yang Relevan Keluhan Pelanggan. *Scientific Journal Informatics*, 2(2). 2015.
- [5] Basmalah, V., Saptono, R., Widya, S., Analisis Perbandingan Metode Vector Space Model dan Weighted Tree Similarity dengan Cosine Similarity pada kasus Pencarian Informasi Pedoman Pengobatan Dasar di Puskesmas. *JURNAL ITSMART ISSN: 2301-7201*. 2015.
- [6] Kurchaniya, D. and Kumar, P., Analysis of Different Similarity Measures in Image Retrieval Based on Texture and Shape. *International Research Journal of Engineering and Technology (IRJET)*. 2017.
- [7] Badrus, Z., Alfian, S., dan Endah, P., Klasifikasi Dokumen Temu Kembali Informasi dengan K-Nearest Neighbour. e-ISSN 2442-5168. 2015.
- [8] Ashshidhiqi, G.B.H. Rekomendasi rancangan Information Architecture Website Institusi Pendidikan Tinggi Menggunakan Metode Card Sorting pada Metode Goal-Directed Design. e-Proceeding of Engineering, Vol 2: 2. 2015.