Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.202183556
Vol. 8, No. 1, Februari 2021, hlm. 127-134 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

IMPLEMENTASI LATENT DIRICHLET ALLOCATION (LDA) UNTUK KLASTERISASI CERITA BERBAHASA BALI

Ngurah Agus Sanjaya ER[*1]

[1]Program Studi Teknik Informatika, Universitas Udayana
Email: [1]agus_sanjaya@unud.ac.id
[*]Penulis Korespondensi

(Naskah masuk: 13 Mei 2020, diterima untuk diterbitkan: 02 Februari 2021)

Abstrak

Cerita-cerita berbahasa Bali memiliki topik yang beragam namun memuat nilai kearifan lokal yang perlu untuk dilestarikan. Jika cerita-cerita tersebut dapat dikelompokkan berdasarkan topik, tentu akan sangat memudahkan bagi para pembacanya dalam memilih bacaan yang diinginkan. *Latent Dirichlet Allocation* (*LDA*) mengasumsikan bahwa suatu dokumen dibangun dari perpaduan topik-topik tersembunyi. Dengan menerapkan *LDA* pada kumpulan dokumen, maka dapat diketahui distribusi topik-topik tersembunyi pada kumpulan dokumen secara umum maupun masing-masing dokumen. Pada penelitian ini, distribusi topik yang ditemukan oleh LDA pada kumpulan cerita berbahasa Bali digunakan untuk melakukan pengelompokkan cerita secara otomatis. Tahapan penelitian meliputi digitalisasi cerita, tokenisasi, *case-folding*, *stemming*, pencarian topik dengan *LDA*, representasi dokumen dan klasterisasi hirarki secara *agglomerative*. Pengujian dilakukan menggunakan 100 buah data cerita berbahasa Bali yang didapat dari situs daring maupun Dinas Kebudayaan Provinsi Bali untuk menghitung akurasi hasil klasterisasi. Evaluasi dilakukan juga untuk melihat pengaruh jumlah kata dan ukuran kesamaan yang digunakan terhadap akurasi. Akurasi hasil klasterisasi tertinggi yang didapatkan adalah 62% pada saat jumlah kata yang digunakan sebagai representasi dokumen berjumlah 3000 kata. Selain itu, didapatkan suatu kesimpulan bahwa akurasi klasterisasi juga sangat dipengaruhi oleh ukuran kesamaan yang digunakan ketika melakukan penggabungan dokumen serta jumlah kata sebagai representasi dokumen.

Kata kunci: *cerita berbahasa Bali, hierarchical clustering*, *latent dirichlet allocation, klasterisasi*

IMPLEMENTATION OF LATENT DIRICHLET ALLOCATION (LDA) FOR CLUSTERING BALINESE FOLKLORE

Abstract

Balinese folklores have diverse topics but contain local wisdom that needs to be preserved. Grouping the stories based on the topics can certainly help readers to choose their readings accordingly. Latent Dirichlet Allocation (LDA) assumes that a document is built from a combination of hidden topics. By applying LDA to a collection of documents (corpus), the global distribution of hidden topics in the corpus as well as the distribution of each individual document in the corpus can be identified. In this research, the individual distribution of topics in Balinese folklores is used to group stories based on common topics. The research stages include story digitization, tokenization, case-folding, stemming, topic search with LDA, document representation and agglomerative hierarchical clustering. Performance evaluation was carried out using 100 Balinese folklores data obtained from online sites and the Bali Provincial Cultural Office to calculate the accuracy of the clustering results. Evaluation is also carried out to see the effect of the number of words and the similarity measure used on accuracy. The highest accuracy obtained is 62% when the number of words used as the representation of a document is 3000 words. In addition, it can be concluded that accuracy is also greatly influenced by the similarity measure used when merging the documents and the number of words for document representation.

Keywords: *Balinese folklores, hierarchical clustering, latent dirichlet allocation, clustering*

1. PENDAHULUAN

Cerita-cerita berbahasa Bali memiliki topik yang bervariasi namun sarat akan nilai kearifan lokal yang harus dilestarikan. Sebagai contoh, *I Lutung teken I Kekua* (Si Monyet dan Si Kura-kura) memberikan pesan moral agar pembaca selalu bersikap hormat kepada orang lain dan tidak membanggakan diri. Pesan ini disampaikan pula pada banyak cerita-cerita seperti *I Tiwas teken I Sugih* (Si Kaya dan Si Miskin), *Cupak Grantang* (Si Cupak dan Si Grantang) dan lain sebagainya. Jika cerita-cerita yang memiliki kesamaan topik dapat dikelompokkan tentu akan sangat memudahkan pembaca dalam memilih bacaan yang sesuai. Pencarian topik pada cerita-cerita berbahasa Bali dan pengelompokkannya dapat dilakukan secara manual namun tentu saja membutuhkan tenaga dan waktu selain subyektifitas dari pembaca akan sangat berpengaruh pada pengelompokkan yang dihasilkan.

Penelitian-penelitian untuk pengelompokkan (klasterisasi) dokumen berbahasa Bali masih belum banyak dilakukan. Sedangkan pada bahasa Indonesia penelitian-penelitian yang telah dilakukan untuk klasterisasi dokumen menerapkan pendekatan klasifikasi yaitu *K-Nearest Neighbor* (Putri, dkk., 2017), *Chi-squre* (Suharno, dkk., 2017), *Naïve Bayes* (Pratiwi dan Widodo, 2017) maupun pendekatan klasterisasi dengan *K-Means Clustering* (Hudin, dkk., 2018) (Bakti dan Jatmiko, 2017). Pendekatan klasifikasi mengasumsikan bahwa satu dokumen hanya dapat menjadi anggota dari satu kelas atau kelompok dokumen. Hal ini merupakan suatu batasan yang harus direlaksasi karena suatu dokumen dapat mengandung beberapa topik. Penentuan topik-topik yang terkandung dalam suatu dokumen tekstual secara otomatis merupakan tujuan dari pemodelan topik (*topic modelling*). Teknik-teknik pemodelan topik yang memiliki performa unggul antara lain *Probabilistic Latent Semantic Analysis* (PLSA) (Hoffman, 1999) dan *Latent Dirichlet Allocation* (LDA) (Blei, Ng, & Jordan, 2003). Kedua teknik tersebut telah diterapkan pada kumpulan dokumen berbahasa Indonesia (Suhartono, 2015) (Prihatini, dkk., 2017). PLSA dan LDA memiliki karakteristik yang hampir sama namun LDA lebih unggul karena distribusi awal topik diasumsikan diketahui dan mengikuti distribusi Dirichlet (Chiru, dkk., 2015). Untuk klasterisasi dokumen dengan topik-topik yang ditemukan oleh LDA sebagai fiturnya telah diterapkan menggunakan *K-Means* (Guan, 2016). Algoritma *K-Means* pada pendekatan klasterisasi memiliki kelemahan dalam hal penentuan titik pusat (*centroid*) awal yang dilakukan secara acak.

Pada penelitian ini diterapkan suatu klasterisasi dokumen tekstual berupa cerita berbahasa Bali berdasarkan topik-topik yang ditemukan menggunakan LDA. Algoritma klasterisasi yang digunakan adalah klasterisasi hirarki secara *agglomerative* dimana awalnya masing-masing dokumen merupakan klaster tersendiri yang kemudian digabungkan dengan melihat nilai kesamaan antar pasangan dokumen. Selanjutnya dokumen yang dimaksud pada penelitian ini mengacu pada dokumen tekstual cerita berbahasa Bali yang telah terdigitalisasi. Kontribusi-kontribusi dari penelitian ini adalah sebagai berikut:

1. Konstribusi utama dari penelitian ini adalah terciptanya suatu kerangka kerja (*framework*) untuk mengelompokkan dokumen berbahasa Bali yang menggunakan distribusi topik tiap dokumen yang dihasilkan oleh LDA sebagai representasi dokumen dimana proses klasterisasi dilakukan secara *hierarchical agglomerative clustring*.
2. Selain itu, kontribusi khusus yang dihasilkan adalah berupa terciptanya *stemmer* untuk bahasa Bali (Purnajiwa Arimbawa & Sanjaya ER, 2020) yang telah diterapkan pula dalam peringkasan dokumen berbahasa Bali (Abimanyu, dkk., 2020).

2. METODE PENELITIAN

Pada penelitian ini metodelogi yang diajukan meliputi beberapa proses yaitu pemrosesan awal (*preprocessing*), pencarian topik dengan LDA dan pengelompokkan dokumen secara *agglomerative hierarchical clustering*. Tahap pemrosesan awal dilakukan pada kumpulan dokumen sebelum proses pencarian topik. Kata-kata yang telah melalui pemrosesan awal kemudian menjadi masukan untuk proses pencarian topik dengan LDA. Luaran dari LDA, yaitu distribusi topik untuk masing-masing dokumen, selanjutnya digunakan untuk masukan pada tahapan klasterisasi.

Tahap pemrosesan awal dilakukan untuk melakukan standarisasi terhadap kumpulan dokumen. Standarisasi yang dilakukan adalah untuk menyeragamkan dokumen-dokumen yang memiliki gaya penulisan yang berbeda-beda. Gaya penulisan yang dimaksud dalam hal ini meliputi kapitalisasi huruf, penggunaan pengkodean karakter yang sama serta penyederhanaan representasi dokumen ke bentuk terkecilnya yaitu kata. Untuk mengatasi permasalahan-permasalahan tersebut maka pada pemrosesan awal ini dilakukan tahapan-tahapan antara lain proses tokenisasi (*tokenization*), *case folding*, penghapusan kata-kata tidak penting (*stopwords removal*) dan *stemming*.

Tujuan dari tokenisasi adalah untuk mengubah dokumen tekstual menjadi bagian terkecil (*token*) yang akan digunakan untuk proses-proses selanjutnya. Pada penelitian ini bagian terkecil dari suatu dokumen yang digunakan adalah berupa kata. Masukan pada tahap ini adalah kumpulan dokumen tekstual digital berbahasa Bali. Dokumen tersebut pertama kali dipecah menjadi kumpulan paragraf dengan menggunakan karakter ganti baris sebagai

pemecah. Selanjutnya, masing-masing paragraf dipecah menjadi kata-kata dengan menggunakan tanda spasi sebagai penanda akhir dari suatu kata. Sebagai contoh, kalimat "*I mémé majalan ka peken meli weh-wehan*" (Si Ibu berangkat ke pasar untuk membeli buah-buahan) setelah melewati tahap ini akan dipecah menjadi kata-kata "*I*" (Si), "*mémé*" (Ibu), "*majalan*" (berangkat), "*ke*" (ke), "*peken*" (pasar), "*meli*" (membeli), "*weh-wehan*" (buah-buahan). Pada tahap ini dilakukan pula penyeragaman penggunakan karakter dan penghapusan tanda baca. Kata "*mémé*" dan "*weh-wehan*" disini akan diubah menjadi "*meme*" dan "*weh wehan*". "*weh wehan*" kemudian akan dipecah lagi menjadi "*weh*" dan "*wehan*".

Selanjutnya *case folding* bertujuan untuk menyeragamkan kapitalisasi huruf yang dipakai. Kapitalisasi yang digunakan pada penelitian ini adalah huruf kecil. Pemilihan huruf kecil untuk kapitalisasi dapat memunculkan permasalahan ketika suatu kata sebenarnya merupakan akronim yang menunjukkan persona/lokasi. Pada penelitian ini, pengaruh pemilihan penggunaan kapitalisasi tersebut diabaikan dan tidak menjadi fokus penelitian. Dari contoh kalimat yang diberikan sebelumnya maka kata "*I*" diubah menjadi "*i*". Selanjutnya dilakukan penghapusan kata-kata yang sering muncul seperti kata ganti orang ("*I*", "*Ni*", "*Ipun*") dan kata sambung ("*ka*", "*ring*", "*saking*"). Pada contoh kalimat sebelumnya maka "*I*" dan "*ka*" akan dihapus dari daftar kata untuk kalimat tersebut sehingga tersisa kata-kata "*meme*", "*mejalan*", "*peken*", "*meli*", "*weh*" dan "*wehan*".

Tujuan dari proses *stemming* adalah untuk mengubah kata-kata yang telah mengalami imbuhan menjadi bentuk dasarnya kembali (Pati & Pati, 2017). Penelitian-penelitian sebelumnya yang bertujuan untuk melakukan *stemming* pada dokumen berbahasa Bali menggunakan berbagai metode seperti metode berbasis aturan (*rule based*) (Nata & Yudiastra, 2017) dan kombinasi metode berbasis aturan dengan *n-gram* (Subali & Fatichah, 2019). Pada (Nata & Yudiastra, 2017), kata-kata yang dapat diluluhkan hanyalah yang mendapatkan imbuhan berupa awalan dan akhiran, sedangkan sisipan maupun konfiks/simulfiks, kombinasi afiks tidak diperhitungkan. Pada penelitian ini, dikembangkan pula suatu *stemmer* berbahasa Bali yang menggabungkan antara pendekatan berbasis aturan dan morfologi bahasa Bali itu sendiri (Purnajiwa Arimbawa & Sanjaya ER, 2020)[1].

Kata-kata dalam bahasa Bali yang mendapatkan sisipan, konfiks/simulfiks, serta kombinasi afiks dapat diluluhkan ke kata dasarnya. Aturan-aturan yang digunakan untuk melakukan peluluhan didapatkan dengan memperhatikan morfologi bahasa Bali itu sendiri. Tabel 1 memperlihatkan aturan-aturan yang digunakan untuk melakukan peluluhan pada kata-kata yang telah diberikan sisipan, konfiks, simulfik maupun kombinasi afiks. Aturan lengkap peluluhan awalan dan akhiran dalam bahasa Bali serta penjelasan rinci algoritma *stemmer* dapat dilihat pada (Purnajiwa Arimbawa & Sanjaya ER, 2020). Variasi imbuhan yang dapat diluluhkan oleh *stemmer* yang dikembangkan meliputi awalan, akhiran, sisipan, konfiks/simulfiks dan kombinasi afiks. Dari kata-kata yang digunakan sebagai contoh sebelumnya yaitu "*meme*", "*mejalan*", "*peken*", "*meli*", "*weh*" dan "*wehan*" ketika melalui proses *stemming* beberapa kata yaitu "*mejalan*" dan "*wehan*" akan diubah menjadi "*jalan*" dan "*weh*".

Hasil peluluhan akan dibandingkan lagi dengan kata-kata yang ada pada kosa kata (*vocabulary*) untuk memastikan bahwa peluluhan telah dilakukan dengan benar. Kosa kata bahasa Bali yang digunakan pada penelitian ini adalah diambil dari situs BasaBali[2].

Tabel 1. Aturan Peluluhan Sisipan, Konfiks, Simulfiks dan Kombinasi Afiks

Aturan	Operasi	Contoh	
		Kata Awal	Kata Dasar
Sisipan			
[^aiueo]*in[aiueo][a-z]*	in ⇒ None (dihapus)	*Sinurat*	*Surat*
^in[aiueo][a-z]*	in ⇒ None	*Inucap*	*Ucap*
[^aiueo]*um[aiueo][a-z]*	um ⇒ None	*Rumaksa*	*Raksa*
^um[aiueo][a-z]*	um ⇒ None	*Umawak*	*Uwak*
[^aiueo]*el[aiueo][a-z]*	el ⇒ None	*Telapak*	*Tapak*
[^aiueo]*er[aiueo][a-z]*	er ⇒ None	*Gerudug*	*Gudug*
Konfiks			
^pa[a-z]*an$	^pa ⇒ None an$ ⇒ None	*Pasirepan*	*Sirep*
^ka[a-z]*an$	^ka ⇒ None an$ ⇒ None	*Kasengsaraan*	*Sengsara*
^ma[a-z]*an$	^ma ⇒ None an$ ⇒ None	*Majemakan*	*Jemak*
^bra[a-z]*an$	^bra ⇒ None an$ ⇒ None	*Bragedegan*	*Gedeg*
^man[a-z]*in$	^ma ⇒ t,d in$ ⇒ None	*Manuturin*	*Tutur*
^mang[a-z]*ang$	^mang ⇒ None ang$ ⇒ None	*Mangorahang*	*Orah*
Simulfiks			
^mam[a-z]*	^mam ⇒ b,p	*Mamuduh*	*Buduh*
^pang[a-z]*	^pang ⇒ k,g	*Pangalung*	*Kalung*
Kombinasi Afiks			
^man[a-z]*in$	^ma ⇒ t,d In$ ⇒ None	*Manuturin*	*Tutur*
^mang[a-z]*ang$	^mang ⇒ None ang$ ⇒ None	*Mangorahang*	*Orah*

Untuk kata-kata yang telah melalui proses *stemming* namun tidak ditemukan pada kosa kata, maka akan dikembalikan suatu daftar kandidat kata. Daftar kata yang dikembalikan terdiri atas kata-kata yang dianggap paling mirip berdasarkan kesamaan antar kata yang dihitung menggunakan *Levenshtein Distance* (Levenshtein, 1966). Dari daftar kata

[1] Implementasi dari *stemmer* untuk bahasa Bali dapat diakses pada alamat https://github.com/anggapur/lematizationBahasaBali

[2] https://dictionary.basabali.org

tersebut maka nantinya dapat ditentukan salah satu kata yang merupakan kata dasar dari kata yang tidak ada pada kosa kata tersebut. Sebagai contoh, untuk kata "*satur*" yang tidak ada pada kosa kata maka contoh daftar kata yang mungkin dikembalikan antara lain "*satua*" (cerita), "*satus*" (seratus), "*sanur*" (nama lokasi), "*batur*" (nama lokasi), dan lain-lain. Jika tidak ada kata yang sesuai dari daftar kata yang dikembalikan maka kata yang tidak diketahui tadi tidak akan digunakan dalam proses selanjutnya.

Setelah dilakukan tahap pemrosesan awal, tahap selanjutnya adalah pencarian topik-topik yang tersembunyi pada kumpulan dokumen dengan menggunakan LDA. Masukan pada tahap ini adalah dokumen digital cerita berbahasa Bali yang telah melalui tahap pemrosesan awal. Semua kata pada masing-masing dokumen telah diseragamkan kapitalisasinya dan telah diubah ke dalam bentuk kata dasarnya. Selain kumpulan dokumen yang sudah melalui pemrosesan awal, LDA membutuhkan masukan lain berupa jumlah topik yang diinginkan (*K*) dan jumlah kata yang merepresentasikan masing-masing topik (*N*). Disamping kedua masukan tersebut, LDA juga memerlukan dua masukan *hyperparameter* yaitu α dan β. α mengendalikan distribusi topik pada dokumen. Semakin kecil nilai α maka dokumen cenderung hanya mengandung satu topik, begitu juga sebaliknya. Jika nilai α semakin besar maka dokumen akan mengandung topik-topik secara merata (*uniform*). β mengendalikan distribusi kata pada suatu topik. Semakin kecil nilai parameter ini maka suatu topik cenderung terdiri atas sedikit kata atau tidak ada variasi kata pada topik tersebut. Nilai parameter β yang besar akan mengakibatkan topik-topik memiliki variasi kata yang banyak.

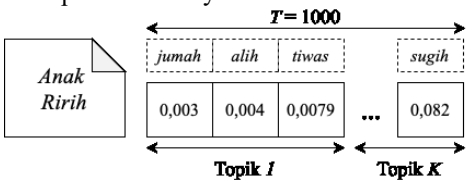
Tabel 2. Luaran dari *Latent Dirichlet Allocation* (*LDA*)
(a) Distribusi Kata per Topik

Topik (*K=10*)	**Kosa Kata**				
	alih	*jumah*	*sugih*	…	*tiwas*
1	0,004	0,003	0,082	…	0,079
2	0,021	0,012	0,001	…	0,031
…	…	…	…	…	…
10	0,007	0,042	0,002	…	0,007

(b) Distribusi Topik per Dokumen

ID/Judul	**Topik (*K=10*)**				
	1	*2*	*3*	…	*10*
1 - *Anak Ririh*	0,44	0,11	0,22	…	0,09
2 – *I Belog*	0,41	0,45	0,01	…	0,13
…	…	…	…	…	…
100 – *Subali*	0,01	0,46	0,52	…	0,01

Dengan memberikan masukan-masukan seperti yang dijelaskan sebelumnya, maka LDA dapat menemukan topik-topik tersembunyi dari suatu kumpulan dokumen. Luaran pertama yang didapatkan dari LDA adalah berupa probabilitas kata-kata pada tiap topik. Suatu topik *k* sesungguhnya mengandung semua kosa kata dari kumpulan dokumen yang digunakan. Masing-masing kata pada topik *k* tersebut akan memiliki nilai pada rentang 0 sampai dengan 1. Nilai ini mengindikasikan tingkat probabilitas dari suatu kata untuk muncul pada topik tertentu (Tabel 2a). Sebagai contoh kata-kata "*tiwas*" (miskin) dan "*sugih*" (kaya) memiliki probabilitas yang lebih tinggi untuk muncul pada Topik 1 dibandingkan topik-topik lainnya. Selain itu, LDA juga memberikan luaran berupa probabilitas dari masing-masing topik pada tiap dokumen (Tabel 2b). Pada tahap ini, representasi dari masing-masing dokumen adalah berupa probabilitas dari masing-masing topik. Dari Tabel 2b, untuk Dokumen *1* yaitu *Anak Ririh* (Anak Pintar), Topik *1* merupakan topik dengan probabilitas tertinggi. Oleh karena itu, kata-kata dari Topik *1*, seperti "*sugih*" (kaya) dan "*tiwas*" (miskin) akan sering muncul pada cerita *Anak Ririh*. Jika dilihat dari judul dokumen tersebut maka pesan yang disampaikan di dalamnya kemungkinan adalah bahwa anak yang rajin dapat menjadi kaya dan demikian pula sebaliknya.



Gambar 1. Representasi Dokumen

Luaran dari tahapan sebelumnya memberikan informasi tentang distribusi probabilitas topik dari masing-masing dokumen serta distribusi probabilitas kata untuk masing-masing topik. Disamping representasi dokumen yang telah dijelaskan sebelumnya, masing-masing dokumen juga dapat direpresentasikan dengan kata-kata yang memiliki probabilitas paling tinggi pada masing-masing topik. Sebagai contoh, dapat dilihat pada Tabel 2b untuk Dokumen *1* dengan judul *Anak Ririh*, dimana distribusi topik-topiknya didominasi oleh Topik *1* (0,44), Topik *2* (0,11), Topik *3* (0,22) dan Topik *10* (0,09). Dari Tabel 2a sendiri dapat kita lihat bahwa masing-masing Topik *1* sampai dengan Topik *10* memiliki distribusi kata-katanya masing-masing. Oleh karena itu, untuk tahapan selanjutnya yaitu tahap klasterisasi maka suatu dokumen akan direpresentasikan oleh gabungan kata-kata sejumlah *T* buah. *T* buah kata-kata untuk Dokumen *1* ini dapat diambil dari distribusi topiknya yaitu sebanyak 44%, 11%, 22%, 9% masing-masing dari Topik *1*, Topik *2*, Topik *3* dan Topik *10*. Sebelum mengambil kata-kata dari tiap topik sesuai dengan distribusi topiknya maka kata-kata pada tiap topik harus diurut secara menurun (*descending*) berdasarkan nilai probabilitasnya. Kata-kata teratas dari tiap topik

kemudian diambil sejumlah persentase dari masing-masing topik. Ilustrasi representasi dokumen *Anak Ririh* yang digunakan pada penelitian ini dapat dilihat pada Gambar 1 dimana jumlah kata $T = 1000$.

Penelitian-penelitian sebelumnya untuk klasterisasi dokumen menggunakan topik-topik yang ditemukan oleh LDA sebagai fitur dari masing-masing dokumen. Teknik *Multi-Grain Clustering Topic Model* (MGCTM) diusulkan oleh Xie, dkk. (Xie & Xing, 2016) dengan mengintegrasikan klasterisasi dokumen secara langsung pada model LDA yang telah dimodifikasi. Model yang dikemukakan terdiri atas dua komponen yang berhubungan. Komponen pertama adalah komponen campuran yang digunakan untuk menemukan topik yang tersembunyi dalam kumpulan dokumen. Komponen kedua adalah komponen pemodelan topik yang bertugas untuk menambang topik-topik yang bervariasi termasuk topik-topik lokal yang spesifik untuk tiap klaster dan topik-topik global yang terkandung pada semua klaster. Perbedaan mendasar antara penelitian Xie, dkk. dengan penelitian ini adalah LDA pada penelitian ini digunakan untuk mendapatkan distribusi topik masing-masing dokumen dan menentukan gabungan kata-kata pada masing-masing topik tanpa mengubah model LDA aslinya.

Pada penelitian lain, klasterisasi dokumen dilakukan dengan algoritma *K-Means* dan LDA (Guan, 2016). Untuk meningkatkan dampak klasterisasi dokumen menggunakan *K-Means*, titik-titik pusat klaster pada awalnya ditemukan dengan mencari topik-topik tersembunyi yang dihasilkan oleh LDA. Salah satu permasalahan mendasar pada algoritma *K-Means* adalah adanya penentuan titik pusat klaster (*centroid*) yang dilakukan secara acak (*random*). Melihat permasalahan pada algoritma *K-Means* tersebut maka pada penelitian ini klasterisasi dokumen dilakukan secara *agglomerative hierarchical clustering* (Tan, 2006). Klasterisasi secara hirarki ini dapat menghilangkan sifat acak yang terjadi pada algoritma *K-Means* karena dua klaster yang berbeda hanya akan digabung jika mereka memiliki "kedekatan".

$$\cos(\boldsymbol{\theta}) = \frac{\boldsymbol{D_i}.\boldsymbol{D_j}}{\|\boldsymbol{D_i}\|\|\boldsymbol{D_j}\|} = \frac{\sum_{k=1}^{n} \boldsymbol{D_{ik}} \boldsymbol{D_{jk}}}{\sqrt{\sum_{k=1}^{n} \boldsymbol{D_{ik}^2}} \sqrt{\sum_{k=1}^{n} \boldsymbol{D_{jk}^2}}} \quad (1)$$

Luaran dari tahapan sebelumnya adalah masing-masing dokumen yang direpresentasikan oleh kata-kata dengan probabilitas tertinggi dari masing-masing topik sesuai dengan distribusi topik pada dokumen tersebut yang didapatkan dari penerapan LDA. Tahap selanjutnya adalah klasterisasi hirarki yang dilakukan secara *agglomerative* dimana pada awalnya masing-masing dokumen dianggap sebagai klaster-klaster tersendiri. Selanjutnya akan dilakukan penggabungan dua buah klaster yang dianggap "dekat" sesuai dengan ukuran kedekatan yang digunakan. Sebelum melakukan penggabungan klaster berdasarkan ukuran kedekatan antar dua dokumen, maka pada penelitian ini dihitung terlebih dahulu nilai *cosine similarity* antara pasangan dokumen tersebut menggunakan persamaan (1). D_i dan D_j pada persamaan (1) di atas adalah dua buah dokumen yang dihitung nilai kesamaannya. D_{ik} dan D_{jk} merupakan fitur-fitur dari dokumen D_i dan D_j yang dalam hal ini adalah nilai probabilitas dari masing-masing kata pada dokumen tersebut. Ketika penghitungan nilai *cosine similarity* untuk semua pasangan dokumen telah selesai dilakukan maka akan didapatkan suatu matriks M seperti yang ditampilkan pada Tabel 3a.

Tabel 3. Nilai *Cosine Similarity*

(a) Matriks Awal

	D_1	D_2	D_3	…	D_{100}
D_1	-	0,78	0,21	…	0,43
D_2	0,78	-	0,48	…	0,19
D_3	0,21	0,48	-	…	0,22
…	…	…	…	-	…
D_{100}	0,43	0,19	0,22	…	-

(b) Matriks Setelah Dokumen 1 dan 2 Digabung Menggunakan Metode *Single Linkage*

	D_1D_2	D_3	…	D_{100}
D_1D_2	-	0,21	…	0,19
D_3	0,21	-	…	0,22
…	…	…	-	…
D_{100}	0,19	0,22	…	-

Label pada masing-masing kolom dan baris pada matriks di atas menunjukkan nomor dokumen (D_1, …, D_{100}). Masing-masing entri M_{ij} pada matriks M di atas merupakan nilai *cosine similarity* antara dokumen D_i dan D_j. Sebagai contoh, nilai *cosine similarity* antara dokumen 1 dan 2 sesuai dengan matriks M *pada* Tabel 3a adalah 0,78. Proses penggabungan dua klaster (dokumen) selanjutnya dapat dilakukan sesuai dengan ukuran kedekatan yang digunakan. Ukuran-ukuran kedekatan untuk *agglomerative hierarchical clustering* antar dua klaster yang dapat digunakan antara lain *single-linkage* (*MIN*), *double-linkage* (*MAX*), *average-linkage* (*Group-Average*) maupun metode *Ward* (Ward, 1963).

$$I_{ij} = SSE_{ij} = \frac{1}{2}\sum_{n=1}^{p}(M_{in} - M_{jn})^2 \quad (2)$$

Pada *MIN* maka untuk pertama kali dua klaster, dalam hal ini dokumen, (D_i, D_j) yang digabung adalah pasangan dokumen yang memiliki nilai yang kesamaan terbesar. Oleh karena pasangan dokumen 1 dan 2 memiliki nilai kesamaan tertinggi maka kedua dokumen tersebut adalah yang digabungkan pertama kali. Untuk itu maka nilai *cosine similarity* dokumen D_i dan D_j (M_i dan M_j) dihapus dari matriks di atas dan diganti dengan suatu klaster dokumen baru yaitu D_iD_j. Jarak D_iD_j ke klaster (dokumen) lain ($n \neq i, j$) selanjutnya dihitung sebagai $min\{M_{in}, M_{jn}\}$ untuk baris dan $min\{M_{ni}, M_{nj}\}$ untuk kolom. Tabel 3b menunjukkan nilai *cosine similarity* yang telah diperbaharui dengen menperhatikan klaster yang baru terbentuk (dokumen D_1D_2). Pembaharuan

matriks M ini akan terus dilakukan ketika terjadi penggabungan dua klaster yang berdekatan. Proses yang sama dilakukan juga pada penggabungan menggunakan metode *MAX*, *Group-Average* maupun *Ward* dengan perbedaan hanya pada penghitungan nilai D_iD_j ke klaster-klaster lain. Pada *MAX* dan *Group-Average*, jarak D_iD_j ke klaster lain pada matriks M dihitung sebagai $max\{M_{in}, M_{jn}\}$ dan $\frac{m_i.M_{in}+m_j.M_{jn}}{m_i+m_j}$ untuk baris serta $max\{M_{ni}, M_{nj}\}$ dan $\frac{m_i.M_{ni}+m_j.M_{nj}}{m_i+m_j}$ untuk kolom. Metode *Ward* menggunakan *sum of squarred error* (*SSE*) yang ditunjukkan pada persamaan (2) untuk menentukan klaster yang harus digabung. I_{ij} adalah kedekatan antara klaster D_i dan D_j, sedangkan n merupakan jumlah anggota pada klaster. Dengan demikian, metode *Ward* akan memilih untuk menggabungkan dua klaster yang akan meminimumkan galat (*error*) atau dalam hal ini menggabungkan dua dokumen yang memiliki nilai kesamaan (*cosine similarity*) yang paling tinggi. Penggabungan dua klaster yang berdekatan akan terus dilakukan sampai jumlah klaster yang diinginkan telah dipenuhi.

3. HASIL DAN PEMBAHASAN

A. Data dan Skenario Pengujian

Dokumen yang digunakan pada uji coba berjumlah 100 buah yang sebagian besar didapat dari situs daring (msatuabali[3], ngiringmabasabali[4], satua-bali[5]) dan digitalisasi cerita berbahasa Bali (*Satua Bali*) yang didapat dari Dinas Kebudayaan Provinsi Bali. Kosa kata yang digunakan berjumlah 8400 buah kata yang diambil dari situs BasaBali. Kumpulan dokumen tersebut kemudian melalui tahap pemrosesan awal yang telah dijelaskan sebelumnya. Masing-masing dokumen kemudian dikelompokkan oleh seorang ahli bahasa Bali yang digunakan sebagai *baseline* untuk dapat mengukur performa berupa akurasi dari klasterisasi yang dihasilkan. Dokumen-dokumen tersebut dikelompokkan oleh ahli bahasa Bali menjadi 3 yaitu dongeng rakyat (40 dokumen), fabel (30 dokumen) dan cerita biasa (30 dokumen).

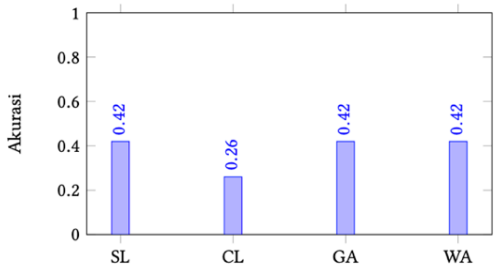
Pengujian pertama dilakukan untuk melihat akurasi dari hasil klasterisasi yang dilakukan. Pada tahap pencarian topik dengan LDA, masukan yang diberikan adalah $\alpha = 0{,}5$; $\beta = 0{,}5$; K (jumlah topik yang diinginkan) = 3. Jumlah K yang merupakan masukan untuk proses klasterisasi disesuaikan dengan jumlah kelompok yang dibuat oleh ahli bahasa Bali. Nilai 0,5 pada α dan β mengindikasikan asumsi yang digunakan bahwa semua cerita tidak mengandung topik yang dominan dan tiap kata memiliki kemungkinan yang sama untuk muncul pada suatu topik. Masing-masing dokumen kemudian direpresentasikan oleh 1000 buah kata (T = 1000) yang disesuaikan dengan sebaran topik pada masing-masing dokumen. Selanjutnya representasi dokumen tersebut digunakan untuk melakukan klasterisasi. Jumlah klaster C yang diinginkan adalah 3. Pengujian kedua kemudian dilakukan untuk melihat pengaruh pemilihan nilai T terhadap akurasi dari hasil klasterisasi. Pada pengujian kedua ini digunakan nilai T yang bervariasi yaitu (T = 3000, 5000 dan 8000).

Akurasi dari hasil klasterisasi pada pengujian pertama dan kedua dihitung menggunakan persamaan (3). T_{ji} akan bernilai 1 jika dokumen $D_j \in D$ dikelompokkan ke klaster $c_i \in C$ yang sesuai dengan *baseline* dan bernilai 0 jika sebaliknya. $|D|$ merupakan jumlah dokumen yang digunakan yaitu 100.

$$Akurasi = \frac{\sum_{c_i \epsilon C} \sum_{d_j \epsilon D} T_{ji}}{|D|} \quad (3)$$

B. Pembahasan

Klasterisasi secara *agglomerative* akan berhenti dilakukan ketika jumlah klaster yang diinginkan telah dipenuhi. Satu permasalahan yang dihadapi pada penelitian ini adalah label dari klaster yang dihasilkan tidak diketahui. Pelabelan klaster ini sangat penting untuk dilakukan agar akurasi hasil dapat dihitung. Oleh karena itu, pada penelitian ini penentuan label dari satu klaster dilakukan dengan membandingkan jumlah dokumen yang masuk ke masing-masing klaster pada *baseline*. Sebagai contoh, untuk klaster c_1 yang dihasilkan dari proses klasterisasi ini akan dihitung terlebih dahulu berapa jumlah dokumen pada c_1 yang masuk ke klaster dongeng rakyat, fabel dan cerita biasa. Jika dokumen klaster dongeng rakyat paling banyak ditemukan pada c_1 maka c_1 diasumsikan adalah klaster dongeng rakyat. Hal yang sama dilakukan juga untuk klaster-klaster lain $c_i \in C$. Dengan demikian klaster-klaster hasil klasterisasi dapat diberikan label dan nilai akurasinya dapat dihitung.

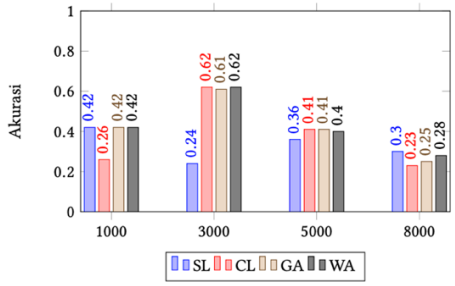


Gambar 2. Akurasi Hasil Klasterisasi dengan T=1000 (SL=*Single Linkage*, CL=*Complete Linkage*, GA=*Group Average*, WA=*Ward*)

Akurasi hasil klasterisasi untuk pengujian pertama dapat dilihat pada Gambar 2 dimana sumbu X dan Y masing-masing menyatakan ukuran kesamaan yang digunakan dan nilai akurasi yang didapat. Dari Gambar 2, penggabungan pasangan dokumen menggunakan ukuran kedekatan *Complete*

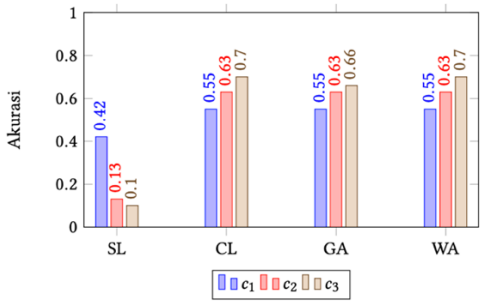
[3] https://msatuabali.blogspot.com/p/satua-bali-i-siap-selem-msatuabali.html
[4] https://ngiringmabasabali.wordpress.com/category/cerita-rakyat-bali/
[5] https://satua-bali.blogspot.com

Linkage (*CL*) menghasilkan nilai akurasi yang paling kecil (26%), sedangkan *Single Linkage* (*SL*), *Group Average* (*GA*) dan *Ward* (*WA*) memiliki persentase akurasi yang sama yaitu 42%. Dilihat dari proses klasterisasi dimana dua pasangan dokumen (D_1, D_2) digabung terlebih dahulu dan matriks kemudian diperbaharui dengan memilih nilai *cosine similarity* terbesar antara D_1D_2 yang baru terbentuk ke dokumen-dokumen yang lain. Karena nilai yang digunakan pada matriks adalah *cosine similarity* seharusnya dokumen D_1D_2 akan semakin "dekat" dengan dokumen memiliki kesamaan tertinggi. Namun, hasil dari uji coba menyatakan sebaliknya. Sebagai contoh, pada salah satu klaster yang terbentuk terdapat dokumen *I Lutung Teken I Kekua* (Si Monyet dan Si Kura-kura), *Lutung Teken Kambing* (Si Monyet dan Si Kambing), *Anak Ririh* (Anak Pintar), *Ni Bawang Teken Ni Kesuna* (Bawang Merah dan Bawang Putih). Setelah diteliti lebih jauh, klaster ini lebih banyak mengandung fabel. *Anak Ririh* dan *Ni Bawang Teken Ni Kesuna* bukanlah fabel melainkan cerita biasa sehingga terjadi kesalahan dalam klasterisasi. Di sisi lain, *I Siap Selem* (Si Ayam Hitam), *Kancil lan Lutung* (Kancil dan Monyet) yang merupakan fabel menjadi anggota pada klaster lain. Hal ini kemungkinan terjadi karena jumlah kata yang digunakan sebagai representasi dokumen adalah terlalu sedikit. Untuk itu pada pengujian tahap kedua dicoba dengan menggunakan nilai *T* yang lebih besar.



Gambar 3. Akurasi Hasil Klasterisasi untuk *T*=1000, 3000, 5000 dan 8000 (SL=*Single Linkage*, CL=*Complete Linkage*, GA=*Group Average*, WA=*Ward*)

Hasil pengujian tahap kedua dapat dilihat pada Gambar 3. Jumlah kata *T* yang diujikan divariasikan mulai dari 1000, 3000, 5000 dan 8000. Pada saat *T*=3000, *CL*, *GA* dan *WA* mencapai akurasi yang tertinggi yaitu melebihi 60%. Hal ini berbanding terbalik dengan *SL* yang mendapatkan akurasi terendahnya yaitu sebesar 24% pada saat *T*=3000. *CL*, *GA* dan *WA* memiliki performa yang relatif sama untuk $T > 1000$. Penambahan jumlah kata $T > 3000$ tidak dapat meningkatkan performa akurasi ketiganya. Hal ini kemungkinan terjadi karena ketika $T > 3000$ maka kata-kata yang digunakan sebagai respresentasi suatu dokumen akan semakin mirip antar satu klaster dengan klaster yang lainnya sehingga akurasi klaster yang dihasilkan menurun. Di lain sisi, jumlah kata yang terlalu sedikit *T*=1000, mengakibatkan satu dokumen dipenuhi dengan kata-kata yang sangat spesifik dari sebaran topiknya, yang pada akhirnya mengakibatkan ukuran kesamaan antar dokumen menggabungkan pasangan dokumen yang kurang tepat. Dari pengujian kedua ini dapat disimpulkan bahwa pemilihan jumlah kata *T* sangat berpengaruh terhadap hasil akurasi dari klasterisasi disamping ukuran kesamaan yang digunakan pada proses klasterisasi.



Gambar 4. Akurasi pada Masing-Masing Klaster (SL=*Single Linkage*, CL=*Complete Linkage*, GA=*Group Average*, WA=*Ward*)

Jika dilihat dari percobaan pertama dan kedua, ukuran kesamaan *SL* memiliki performa yang relatif sama dengan ukuran kesamaan yang lainnya. Anomali hanya terjadi pada saat $T = 3000$ dimana akurasi yang dicapai hanya 24%. Untuk itu perlu dilihat lebih lanjut kenapa hal ini bisa terjadi. Gambar 4 menunjukkan akurasi yang didapat oleh masing-masing ukuran kesamaan pada masing-masing klaster c_1, c_2 dan c_3. Dari ketiga klaster yang ada *SL* mendapatkan akurasi tertinggi pada c_1 yaitu 42%, namun dua klaster lain akurasinya hanya 13% dan 10%. Untuk ukuran kesamaan (*CL*, *GA*, *WA*) yang lain nilai akurasi masing-masing klaster adalah minimal 55%. Hal ini menunjukkan bahwa *SL* umumnya melakukan kesalahan penggabungan dokumen sehingga akurasi yang didapat kurang baik.

4. KESIMPULAN DAN SARAN

Dari uji coba yang dilakukan dapat ditarik suatu kesimpulan bahwa representasi dokumen berdasarkan topik-topik yang dihasilkan oleh LDA dapat digunakan untuk melakukan klasterisasi dokumen berbahasa Bali secara otomatis dengan tingkat akurasi 62% pada saat jumlah kata yang digunakan sebagai representasi dokumen adalah 3000. Disamping itu, ukuran kesamaan dan jumlah kata sebagai representasi dokumen juga angat mempengaruhi akurasi klasterisasi yang dihasilkan. Penelitian ini dapat dikembangkan dalam hal peningkatan akurasi dengan melihat pengaruh nilai α dan β pada saat pencarian topik dengan LDA. Dari sisi dokumen yang digunakan perlu diperhatikan pula pengaruh tingkatan bahasa (*sor singgih Basa Bali*) yang digunakan pada masing-masing dokumen.

5. UCAPAN TERIMA KASIH

Penelitian ini dibiayai oleh DIPA BLU Universitas Udayana Tahun Anggaran 2019 sesuai dengan Surat Perjanjian Penugasan Pelaksanaan Penelitian Unggulan Program Studi Nomor: 2048/UN14.2.8.II/LT/2019, tanggal 10 April 2019.

DAFTAR PUSTAKA

ABIMANYU, C. G., ER, N., & KARYAWATI, A. A. I. N. E. (2020). BALINESE AUTOMATIC TEXT SUMMARIZATION USING GENETIC ALGORITHM. JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer), 6(1), 13-20. https://doi.org/10.33480/jitk.v6i1.1344

BAKTI, VERY K., dan JATMIKO INDRIYATNO. "Klasterisasi Dokumen Tugas Akhir Menggunakan K-Means Clustering, sebagai Analisa Penerapan Sistem Temu Kembali." Kopertip, vol. 1, no. 1, 2017, pp. 31-34.

BLEI, D. M., NG, A. Y., dan JORDAN, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

CHIRU, COSTIN & REBEDEA, TRAIAN & CIOTEC, SILVIA. (2014). Comparison between LSA-LDA-lexical chains. WEBIST 2014 - Proceedings of the 10th International Conference on Web Information Systems and Technologies. 2.

GUAN, P. (2016). K-means document clustering based on latent dirichlet allocation.

HOFMANN, T. (1999). Probabilistic latent semantic analysis. In Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence (UAI'99). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 289–296.

HUDIN, M., FAUZI, M., & ADINUGROHO, S. Implementasi Metode Text Mining dan K-Means Clustering untuk Pengelompokan Dokumen Skripsi (Studi Kasus: Universitas Brawijaya). Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 2, no. 11, p. 5518-5524, juli 2018. ISSN 2548-964X. Tersedia pada: <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/3332>. Tanggal Akses: 21 agu. 2020

LEVENSHTEIN, V. "Binary Codes Capable of Correcting Deletions, Inser-tions and Reversals,"Soviet Physics Doklady, vol. 10, p. 707, 1966.

NATA, G. N. M. dan YUDIASTRA, P. P., "Stemming teks sor-singgih bahasabali,"E-Proceedings KNS&I STIKOM Bali, pp. 608–612, 2017.

PATIL, H. B. dan PATIL, A. S. "Mars: A rule-based stemmer for morphologically rich language marathi," in 2017 International Conference on Computer, Communications and Electronics (Comptelix), pp. 580–584, IEEE, 2017.

PRATIWI, NUR & WIDODO, WIDODO. (2017). Klasifikasi Dokumen Karya Akhir Mahasiswa Menggunakan Naive Bayes Classifier (NBC) Berdasarkan Abstrak Karya Akhir di Jurusan Teknik Elektro Universitas Negeri Jakarta. 1. 31-38. 10.21009/pinter.1.1.5.

PRIHATINI, Putu Manik; SURYAWAN, I Ketut; MANDIA, I Nyoman. METODE LATENT DIRICHLET ALLOCATION UNTUK EKSTRAKSI TOPIK DOKUMEN. Logic : Jurnal Rancang Bangun dan Teknologi, [S.l.], v. 17, n. 3, p. 153-157, nov. 2017. ISSN 2580-5649.

PURNAJIWA ARIMBAWA, I. G. A.; SANJAYA ER, N. A. Lemmatization in Balinese Language. JELIKU - Jurnal Elektronik Ilmu Komputer Udayana, [S.l.], v. 8, n. 3, p. 235-242, jan. 2020. ISSN 2301-5373. Available at: <https://ojs.unud.ac.id/index.php/JLK/article/view/51892>. Diakses tanggal: 5 Mei 2020.

PUTRI, REKYAN & HERLAMBANG, ROMARIO & WIHANDIKA, RANDY. (2017). Implementasi Metode K-Nearest Neighbour Dengan Pembobotan TF.IDF.ICF Untuk Kategorisasi Ide Kreatif Pada Perusahaan. Jurnal Teknologi Informasi dan Ilmu Komputer. 4. 97. 10.25126/jtiik.201742296.

SUBALI, M. A. P. dan FATICHAH, C., "Kombinasi metode rule-based dan n-gram stemming untuk mengenali stemmer bahasa bali", Jurnal TeknologiInformasi dan Ilmu Komputer, vol. 6, no. 2, pp. 219–228, 2019.

SUHARNO, CLAUDIO & FAUZI, MUHAMMAD & PERDANA, RIZAL. (2017). Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors Dan Chi-square. systemic information system and informatics journal. 3. 25-32. 10.29080/systemic.v3i1.191.

SUHARTONO, DERWIN. (2015). Probabilistic Latent Semantic Analysis (PLSA) untuk Klasifikasi Dokumen Teks Berbahasa Indonesia.

TAN, P. N., STEINBACH, M., dan KUMAR, V. (2006). *Introduction to Data Mining*. Pearson Education.

WARD, J.H.: Hierarchical Grouping to Optimize an Objective Function. Journal of the American Statistical Association. 58, 301, 236--244 (1963).

XIE, P. DAN XING, E.P. (2013). Integrating document clustering and topic modeling. CoRR, abs/1309.6874.